Contents

- Cloud Adoption Framework for Azure
- About the Framework
- What's new
- Adoption journeys
  - Get started
    - Overview
    - Foundational alignment
      - Document foundational alignment decisions
      - How does Azure work?
      - Azure fundamental concepts
      - Portfolio hierarchy
      - Azure hierarchy support
    - Accelerate adoption
      - Accelerate migration
      - Build new products and services
      - Unblock environment design and configuration
  - Improve controls
    - Enable customer success
    - Deliver operational excellence
    - Manage cloud costs
    - Secure the enterprise environment
    - Improve reliability
    - Ensure consistent performance
  - Establish teams
    - Align your organization
    - Build a cloud strategy team
    - Build a cloud adoption team
    - Build a cloud governance team
    - Build a cloud operations team

- Adoption scenarios
  - Hybrid and multicloud
  - Modern application platform
  - SAP
  - Virtual desktops
- Industry scenarios
  - Retail
- Antipatterns to avoid

Strategy

- Overview
- Motivations
- Business outcomes
  - Overview
  - Data innovation
  - Data democratization
  - Fiscal outcomes
  - Agility outcomes
  - Global reach outcomes
  - Customer engagement outcomes
  - Performance outcomes
  - Sustainability outcomes
  - Business outcome template
  - Align efforts to learning metrics
  - Objectives and key results
- Financial considerations
  - Overview
  - Common finance vocabulary terms
  - Understand cloud accounting
- Technical considerations
  - Overview
  - Achieve more with your investment
  - Align your partner strategy

- Build a digital transformation timeline
- Create a business case
- Share your strategy
- Additional considerations
  - Balance competing priorities
  - Balance the portfolio
  - Global markets
  - Define security strategy
  - Define monitoring strategy
  - Responsible AI
  - Accelerate cloud adoption with GitHub
- Skills relevant to strategy
- Strategy antipatterns

Plan

- Overview
- Digital estate
  - The five Rs of rationalization
  - What is a digital estate?
  - Digital estate planning
  - Gather inventory data
  - Rationalize the digital estate
  - Align cost models to forecast costs
  - Measure business outcomes with AppDynamics
- Initial organization alignment
- Skills readiness plan
  - Build a readiness plan
  - Map roles and skills
- Cloud adoption plan
  - Overview
  - Prerequisites
  - Deploy the template to Azure DevOps
  - Define and prioritize workloads

- Align assets to workloads
- Review rationalization decisions
- Establish iterations and release plans
- Estimate timelines
- Best practices
- Digital estate assessment in Azure
- Plan a data warehouse migration
- Strategic Migration Assessment and Readiness Tool (SMART)
- Plan antipatterns
- Ready
- Overview
- Azure setup guide
- Setup guide overview
- Organize resources
- Manage access
- Manage costs and billing
- Plan governance, security, and compliance
- Establish monitoring and reporting
- Stay current with Azure
- Operating model
- Overview
- Define your operating model
- Compare common cloud operating models
- Common operating model terms
- Azure landing zones
- What is a landing zone?
- Journey to the target architecture
- Design areas
- Choose a landing zone option
- Implementation options
- Start with enterprise scale
- Overview

- Architecture
- Design principles
- Design guidelines
- Critical design areas
  - Enterprise enrollment and Azure AD tenants
  - Identity and access management
  - Management group and subscription organization
  - Network topology and connectivity
    - Overview
    - Plan for IP addressing
    - DNS for on-premises and Azure resources
    - Private Link and DNS integration at scale
    - Define an Azure network topology
    - Virtual WAN network topology (Microsoft-managed)
    - Traditional Azure networking topology
    - Connectivity to Azure
    - Connectivity to Azure PaaS services
    - Plan for inbound and outbound internet connectivity
    - Plan for application delivery
    - Plan for landing zone network segmentation
    - Define network encryption requirements
    - Plan for traffic inspection
    - Connectivity to other cloud providers
    - Connectivity to Oracle Cloud Infrastructure
  - Management and monitoring
  - Business continuity and disaster recovery
  - Security, governance, and compliance
  - Platform automation and DevOps
- Implement enterprise-scale landing zones
- Testing approach for enterprise-scale
- Terraform module for CAF enterprise-scale
- Transition to enterprise-scale (brownfield)

- Scenario-specific enterprise-scale landing zones
- Start small and expand
  - CAF Migration landing zone blueprint
  - CAF Foundation blueprint
  - Terraform landing zones
  - Expand your landing zone
  - Basic landing zone considerations
    - Overview
    - Review compute options
    - Review networking options
    - Review storage options
    - Review data options
    - Azure role-based access control
    - Create hybrid cloud consistency
  - Improve landing zone operations
  - Improve landing zone governance
  - Improve landing zone security
    - Overview
    - Onboard a subscription to Azure Security Center
    - Onboard Azure Sentinel
    - Implementing a secure hybrid network architecture in Azure
    - Azure identity management and access control security best practices
    - Azure network security best practices
    - Azure operational security best practices
  - Partner landing zones
  - Infrastructure-as-code theory
    - Refactor landing zones
    - Test-driven development (TDD) for landing zones
    - Landing zone TDD in Azure
- Best practices
  - Overview
  - Resource organization

- Create your initial subscriptions
- Scale with multiple subscriptions
- Organize your subscriptions
- Naming and tagging
- Define your naming convention
- Abbreviations for resource types
- Define your tagging strategy
- Move resources

Networking

- Network boundary security
- Plan virtual networks
- Best practices for network security
- Perimeter networks
- Hub and spoke network topology

Identity and access controls

- Identity management best practices
- Secure privileged access
- Choose an authentication method

Storage

- Storage security guide

Databases

- Database security best practices
- Choose a deployment option in Azure SQL Database

AI + Machine Learning

- Organize and set up Azure Machine Learning environments
- Budget, cost, and quota management with Azure Machine Learning
- Machine learning DevOps (MLOps)

Cost management

- Track costs
- Optimize your cloud investment
- Create and manage budgets
- Export cost data

- Optimize costs from recommendations
- Monitor usage and spending
- Skills relevant to ready and landing zones
- Ready antipatterns

Adopt

- Migrate
  - Overview
  - Azure migration guide
    - Migration guide overview
    - Assess workloads
    - Deploy workloads
    - Release workloads
    - Migration-focused cost-control mechanisms
    - Get assistance
  - Migration scenarios
    - Overview
    - Contoso overview
    - Tailwind Traders case study
      - Assessment and migration planning with Azure Migrate for servers
      - Migration execution with Azure Migrate for servers
    - Windows Server workloads
      - Rehost a Windows application on Azure VMs
    - SQL Server workloads
      - Migrate SQL Server databases to Azure
      - Rehost an application on Azure VMs and Azure SQL Managed Instance
      - Rehost an application on Azure VMs and SQL Server Always On availability groups
    - Storage migration
      - Storage migration overview
      - Tools for unstructured storage migration
    - Linux and open-source databases
      - Migrate open-source databases to Azure
      - Migrate MySQL to Azure

- Migrate PostgreSQL to Azure
- Migrate MariaDB to Azure
- Rehost a Linux application on Azure VMs
- Rehost a Linux application on Azure VMs and Azure Database for MySQL
- Open-source platforms
  - Moodle migration
    - Overview
    - Prepare for Moodle migration
    - Architecture and template
    - Moodle migration resources
    - Migration steps
    - Set up Moodle worker nodes
    - Follow up Moodle migration
- Dev/test workloads
  - Migrate dev/test environments to Azure IaaS
  - Migrate to Azure DevTest Labs
- ASP.NET and PHP web apps
  - Refactor a Windows application onto App Service and Azure SQL Database
  - Refactor a Windows application onto App Service and SQL Managed Instance
  - Refactor a Linux application onto App Service and MySQL
  - Rebuild an application in Azure
  - Refactor TFS to Azure DevOps Services
- Java applications
  - Migration overview
  - Spring Boot to Azure App Service
  - Spring Boot to AKS
  - Spring Boot to Azure Spring Cloud
  - Spring Cloud to Azure Spring Cloud
  - Tomcat to Azure App Service
  - Tomcat to containers on AKS
  - Tomcat to Azure Spring Cloud
  - WebLogic to Azure

- WebLogic to Azure Virtual Machines
- WebLogic with Azure AD via LDAP
- WebLogic with Application Gateway
- WildFly to WildFly on AKS
- WebLogic to WildFly on AKS
- WebSphere to WildFly on AKS
- JBoss EAP to WildFly on AKS

SAP

- SAP migration guide
- Migrate SAP applications to Azure
- Migration methodologies for SAP on Azure

Specialized workloads

- Move on-premises VMware infrastructure to Azure
- Azure NetApp Files
- Oracle in Azure
- Cray in Azure
- VDI

VMware hosts

- Considerations when rehosting VMware
- Prerequisites
  - Secure your environment
  - Private cloud management
  - Private cloud networking
  - VMware platform
  - Azure core integration
- Rehost and disaster recovery options
  - Rehost workload VMs to Private Cloud vCenter
  - Rehost data using Azure Data Box
  - Back up workload VMs
  - Set up CloudSimple Private Cloud as disaster recovery site using Zerto
  - Set up CloudSimple Private Cloud as disaster recovery site using VMware SRM

Data platforms

- Overview

- Azure Database Migration Guide
  - Refactor SQL Server to Azure SQL Database
  - Refactor SQL Server to Azure SQL Managed Instance
  - Refactor SQL Server to SQL Server on Azure VMs
  - Refactor SQL Server to Azure SQL Data Warehouse
  - MySQL
  - PostgreSQL
  - MariaDB
  - MongoDB
  - Cassandra
  - Oracle
  - DB2
  - SAP ASE
  - Access
- Azure Database Migration Service tutorials
  - Azure Database Migration Service overview
  - Migrate SQL Server to Azure SQL Database offline
  - Migrate SQL Server to Azure SQL Database online
  - Migrate SQL Server to Azure SQL Managed Instance offline
  - Migrate SQL Server to Azure SQL Managed Instance online
  - Migrate Amazon RDS SQL Server to Azure SQL Database or Azure SQL Managed Instance online
  - Migrate MySQL to Azure Database for MySQL online
  - Migrate Amazon RDS MySQL to Azure Database for MySQL online
  - Migrate PostgreSQL to Azure Database for PostgreSQL online
  - Migrate Amazon RDS PostgreSQL to Azure Database for PostgreSQL online
  - Migrate MongoDB to Azure Cosmos DB's API for MongoDB offline
  - Migrate MongoDB to Azure Cosmos DB's API for MongoDB online
  - Migrate Oracle to Azure Database for PostgreSQL online
- Azure Stack
  - Overview
  - Planning

- Azure landing zone review
- Assess
- Migrate and deploy
- Govern
- Manage
- Analytics solutions
  - Overview
  - Teradata
  - Netezza
  - Exadata
- Multitenant migration
  - Azure Lighthouse for scale migration
- Mainframes
  - Overview
  - Myths and facts
  - Switch from mainframes to Azure
  - Mainframe application migration
- Migrate secure workloads
  - Securing and managing workloads after migration
  - Azure database security best practices
  - Azure data security and encryption best practices
  - Azure PaaS best practices
  - Azure Service Fabric security best practices
  - Best practices for Azure VM security
  - IoT security best practices
  - Securing PaaS databases in Azure
  - Securing PaaS web and mobile applications using Azure App Service
  - Securing PaaS web and mobile applications using Azure Storage
  - Security best practices for IaaS workloads in Azure
- Best practices
  - Overview
  - Multiple datacenters

- Multiple regions
- Data requirements exceed network capacity
- Set up networking for migrated workloads
- Deploy a migration infrastructure
- Cost optimize migrated workloads
- Scale a migration
- Schema migration and DDL commands
- Database restore points for Azure Synapse
- Governance or compliance

Migrate antipatterns

Process improvements

- Overview
- Prerequisites
  - Overview
  - Decisions that affect migration
  - Environment planning checklist
  - Align roles and responsibilities
  - Agile change management
  - Migration backlog review
- Assess workloads
  - Validate assessment assumptions before migration
  - Classify workloads
  - Keep priorities aligned
  - Evaluate workload readiness
  - Architect workloads
  - Update and refine initial cloud estimates
  - Understand partnership options
  - Manage change
  - Approve architecture changes
- Deploy workloads
  - Overview
  - Promotion models

- Remediate assets
- Replicate assets
- Replicate options
- Stage workloads
- Release workloads
- Overview
- Business change plan
- Business testing
- Benchmark and resize assets
- Prepare for promotion
- Promote to production
- Decommission retired assets
- Conduct retrospectives
- Skills relevant to migrate

Innovate

- Overview
- Azure innovation guide
- Innovation guide overview
- Prepare for customer feedback
- Democratize data
- Engage customers through applications
- Empower adoption
- Interact through devices
- Innovate with AI
- Innovation scenarios
- Kubernetes
- Innovation with Kubernetes
- Application development and deployment
- Cluster design and operations
- Cluster and application security
- AI
- Innovate with AI

- Machine learning
- AI applications and agents
- Knowledge mining

Best practices

- Overview and Azure toolchain
- Democratize data
  - Overview
  - Share data with experts
    - Quickly generate data insights
    - Sharing data with coworkers and partners
    - Embed reports in a website or portal
    - Create new workspaces in Power BI
  - Govern data
    - Classify data
    - Secure data
    - Annotate data with Azure Data Catalog
    - Document data sources with Azure Data Catalog
  - Centralize data
    - Create and query an Azure Synapse Analytics SQL pool
    - Best practices for loading data for data warehousing
    - Visualize warehouse data with Power BI
    - Reference architecture for enterprise BI with Azure Synapse Analytics
    - Manage enterprise big data with Azure Data Lake Storage
    - What is a data lake?
  - Collect data
    - Migrate on-premises data to Azure from SQL, Oracle, or NoSQL platforms
    - Integrate cloud data sources with a SQL Analytics data warehouse
    - Load on-premises data into Azure Synapse Analytics
    - Integrate data - Azure Data Factory to OLAP
    - Use Azure Stream Analytics with Azure Synapse Analytics
    - Reference architecture for ingestion and analysis of new feeds
    - Load data into Azure Synapse Analytics SQL pool

- Engage via applications
  - Overview
  - Citizen developers
    - Power Apps overview
    - Creating applications in Power Apps
    - Create your first workflow with Power Automate
    - Using AI Builder
    - Compliance and data privacy for citizen developer solutions
    - Data loss prevention policies for citizen developer solutions
  - Intelligent experiences
    - Modern web apps
    - Infusing intelligence
    - Chatbots
  - Cloud-native applications
    - Microservices architecture
    - Containers
    - Spring Boot microservices
    - Event-driven applications
- Empower adoption
  - Overview
  - Shared solution
    - Get started with a shared repository via GitHub
    - Get started with a shared backlog
    - Synchronize Power Apps with Azure DevOps
  - Feedback loops
    - Manage feedback with Azure DevOps
  - Continuous integration
    - Continuous integration with Azure Pipelines and GitHub
    - MLOps with Azure Machine Learning
  - Reliable testing
    - Manage and track test plans
  - Solution deployment

- Continuous deployment with Azure Pipelines and GitHub
- Integrated metrics
  - Monitor ASP.NET applications
  - Monitor ASP.NET Core web applications
  - Monitor Node.js applications
  - Monitor mobile applications
  - Monitor web applications
  - Monitor VMs hosting traditional applications
- Interact with devices
  - Overview
  - Mobile experience
    - Extend a legacy claims-processing application with a web and mobile experience
    - Optimize reports to share data on a mobile app
    - Extend Power Apps canvas application to a mobile experience
    - Extend Power Automate to add a mobile experience
    - Secure mobile experiences
  - Mixed reality
    - Develop mixed reality experiences with Unity
    - Quickstarts to add Azure Spatial Anchors to a mixed reality solution
  - Integrated reality and IoT
    - Visualize sensor data with Azure IoT in Power BI
    - Visualize sensor data with Azure IoT Hub in a web solution
    - Securing an IoT solution
    - Get started with Azure Sphere
    - Create a deployment with Azure Sphere
    - Get started with Azure Kinect DK
    - Build your first Azure Kinect DK application
  - Adjusted reality
    - Azure Digital Twins + HoloLens: Adjusting virtual reality
    - Get started with Azure Digital Twins
    - Monitor a building with Azure Digital Twins
    - Azure IoT for cloud-to-device communications guide

- Azure IoT configuration for cloud-to-device communications
- Innovate with AI
  - Overview
  - Machine learning
    - What is machine learning?
    - How to approach machine learning operations
    - The machine learning operations process
    - Machine learning security
    - Machine learning inference and deployment
    - Determine compute instances for your model
    - Set up machine learning workspaces
    - Responsible and trusted AI
  - AI applications and agents
    - What are AI applications?
    - What are AI agents
  - Knowledge mining
    - What is Azure Cognitive Search?
- Process improvements
  - Overview
  - Business value consensus
  - Customer adoption
    - Feedback loops
    - Build with customer empathy
    - Measure for customer impact
    - Learn with customers
    - Customer challenges and blockers
  - Digital invention
    - Develop digital inventions
    - Democratize data
    - Application development for innovative applications
    - Empower adoption
    - Interact with devices

- Predict and influence
- Govern
  - Overview
  - Methodology
  - Benchmark
  - Initial governance foundation
  - Governance foundation improvements
  - Governance guides
    - Overview
    - Standard enterprise governance guide
      - Overview
      - Narrative
      - Initial corporate policy
      - Prescriptive guidance
      - Improve the Security Baseline discipline
      - Improve the Resource Consistency discipline
      - Improve the Cost Management discipline
      - Multicloud scenarios
    - Governance guide for complex enterprises
      - Overview
      - Narrative
      - Initial corporate policy
      - Prescriptive guidance
      - Improve the Identity Baseline discipline
      - Improve the Security Baseline discipline
      - Improve the Resource Consistency discipline
      - Improve the Cost Management discipline
      - Multicloud scenarios
      - Multiple layers of governance
  - Governance considerations
    - Evaluate corporate policy
    - Cloud-ready corporate policy and compliance

- Make corporate policy cloud-ready
- Understand business risks
- Evaluate risk tolerance
- Define corporate policy
- Align design with policy
- Establish policy adherence processes
- Implement cloud service controls
- Regulatory compliance
- Cloud security readiness
- Cloud policy review
- Data classification

Disciplines of cloud governance

Implement disciplines of cloud governance

- Cost management
  - Overview of cost management
  - Download the template
  - Understand business risks
  - Risk tolerance metrics and indicators
  - Sample Cost Management policies
  - Policy compliance processes
  - Improve cost management
  - Best practices
  - Azure tools for cost management
- Security baseline
  - Overview of the security baseline
  - Download the template
  - Understand business risks
  - Risk tolerance metrics and indicators
  - Sample Security Baseline policies
  - Policy compliance processes
  - Improve the security baseline
  - Cloud-native security baseline

- Additional Azure security guidance
- Azure tools for security baseline
- Identity baseline
  - Overview of the identity baseline
  - Download the template
  - Understand business risks
  - Risk tolerance metrics and indicators
  - Sample Identity Baseline policies
  - Policy compliance processes
  - Improve the identity baseline
  - Azure tools for identity baseline
- Resource consistency
  - Overview of resource consistency
  - Download the template
  - Understand business risks
  - Risk tolerance metrics and indicators
  - Sample Resource Consistency policies
  - Policy compliance processes
  - Improve resource consistency
  - Azure tools for resource consistency
  - Resource access management
  - Governance design for a simple workload
  - Governance design for multiple teams
- Deployment acceleration
  - Overview of deployment acceleration
  - Download the template
  - Understand business risks
  - Risk tolerance metrics and indicators
  - Sample Deployment Acceleration policies
  - Policy compliance processes
  - Improve deployment acceleration
  - Azure tools for deployment acceleration

- Govern antipatterns
- Manage
  - Overview
  - Azure Management Guide
    - Before you start
    - Inventory and visibility
    - Operational compliance
    - Protect and recover
    - Enhanced baseline
    - Platform specialization
    - Workload specialization
  - Best practices
    - Overview
    - Azure server management services
      - Introduction to Azure server management services
      - Getting ready for cloud operations
      - Get started with cloud operations
        - Overview
        - Configure the service for a single VM
        - Configure the service for an entire subscription
        - Configure at scale with automation
        - Set up basic alerts
      - Ongoing cloud operations
        - Overview
        - Enable guest configuration policy
        - Critical changes (tracking and alerting)
        - Update schedules
        - Common policies in Azure
      - Review of tools and services
    - Monitoring
      - Overview
      - Monitoring cloud models

- Observability
- Data collection
- Alerting
- Monitoring platforms overview
- Skills relevant to monitoring
- Centralize management operations
- Establish an operational fitness review
- Improving platform or workload reliability
- Reliability checklist for Azure services
- Failure mode analysis
- Recover from a region wide service disruption
- Recover from data corruption or accidental deletion
- Management considerations
- Overview
- Business alignment
- Define criticality
- Understand business impact
- Establish business commitments
- Management disciplines
- Inventory and visibility
- Operational compliance
- Protect and recovery
- Platform operations
- Workload operations
- Advanced management and system design
- Manage antipatterns

Secure

- Overview
- Methodology
- Risk insights
- Security integration
- Business resilience

- Access control
- Security operations
- Asset protection
- Security governance
- Innovation security
- DevSecOps controls
- Best practices
- Securing DevOps
- Secure DevOps governance
- Azure Virtual Desktop
- Managing secrets
- Network security
- Toolchain
- Security considerations
- Security strategy
- Roles and responsibilities
- Getting started
- Azure security top 10
- Microsoft cybersecurity reference architectures
- Security documentation
- Azure security benchmarks

Organize

- Managing organization alignment
- Required cloud functions
- Cloud strategy functions
- Cloud adoption functions
- Cloud governance functions
- Central IT functions
- Cloud operations functions
- Cloud center of excellence functions
- Cloud platform functions
- Cloud automation functions

- Cloud data functions
- Cloud security functions
  - Cloud security team functions
  - Policy and standards
  - Security operations center (SOC)
  - Security architecture
  - Security compliance management
  - People security
  - Application security and DevSecOps
  - Data security
  - Infrastructure and endpoint security
  - Identity and keys
  - Threat intelligence
  - Posture management
  - Incident preparation
- Mature teams structure
- Align the RACI matrix
- Building technical skills
- Creating a cost-conscious organization
- Organize antipatterns
  - Organizational antipatterns
  - IT fiefdoms and IT silos

Resources

- Tools and templates
- Decision guides
  - Overview
  - Subscriptions
  - Identity
  - Policy enforcement
  - Resource consistency
  - Resource tagging
  - Encryption

- Software-defined networks
  - Overview
  - PaaS-only
  - Cloud-native
  - Cloud DMZ
  - Hybrid
  - Hub and spoke model
- Logging and reporting
- Migration tools

Additional resources

- Azure Architecture Center for solution architecture
- Microsoft Azure Well-Architected Framework for workload architecture
- Docs for product implementation
- Learn to develop skills
- Assessments to personalize guidance

Archived resources

- Cloud Operating Model
- Azure enterprise scaffold
- Virtual Datacenter (VDC)

What is the Microsoft Cloud Adoption Framework for Azure?

8/25/2021 • 3 minutes to read • Edit Online

The Microsoft Cloud Adoption Framework for Azure is proven guidance that's designed to help you create and implement the business and technology strategies necessary for your organization to succeed in the cloud. It provides best practices, documentation, and tools that cloud architects, IT professionals, and business decision makers need to successfully achieve short-term and long-term objectives.

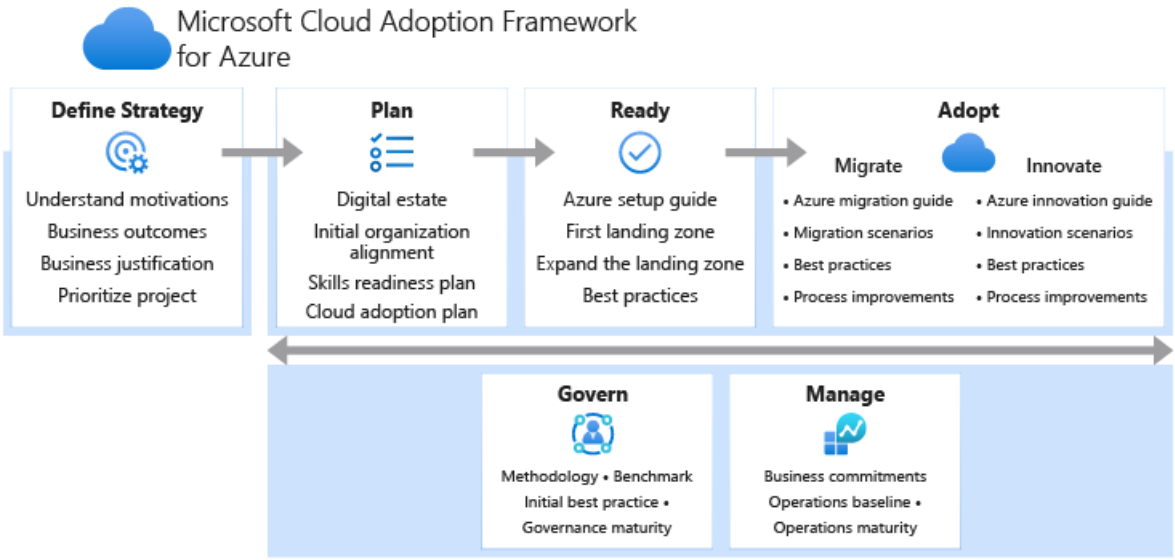
By using the Microsoft Cloud Adoption Framework for Azure best practices, organizations can better align their business and technical strategies to ensure success. Watch the following video to learn more.

The Cloud Adoption Framework brings together cloud adoption best practices from Microsoft employees, partners, and customers. It provides a set of tools, guidance, and narratives that help shape technology, business, and people strategies for driving desired business outcomes during your cloud adoption effort. Review the guidance for each methodology below, providing you with easy access to the right guidance at the right time.

	Strategy: Define business justification and expected outcomes of adoption.		Plan: Align actionable adoption plans to business outcomes.
	Ready: Prepare the cloud environment for the planned changes.		Migrate: Migrate and modernize existing workloads.
	Innovate: Develop new cloud-native or hybrid solutions.		Govern: Govern the environment and workloads.
	Manage: Operations management for cloud and hybrid solutions.		Organize: Align the teams and roles supporting your organization's cloud adoption efforts.

Understand the lifecycle

Each methodology above is part of a broad cloud adoption lifecycle. The Cloud Adoption Framework is a full lifecycle framework, supporting customers throughout each phase of adoption by providing methodologies as specific approaches to overcoming common blockers, as shown here.



Intent

The cloud fundamentally changes how enterprises procure, use, and secure technology resources. Traditionally, enterprises assumed ownership of and responsibility for all aspects of technology, from infrastructure to software. By moving to the cloud, enterprises can provision and consume resources only when they're needed. Although the cloud offers tremendous flexibility in design choices, enterprises need a proven and consistent methodology for adopting cloud technologies. The Microsoft Cloud Adoption Framework for Azure meets that need, helping guide decisions throughout cloud adoption.

But cloud adoption is only a means to an end. Successful cloud adoption starts well before a cloud platform vendor is selected. It begins when business and IT decision makers realize that the cloud can accelerate a specific business transformation objective. The Cloud Adoption Framework helps align strategies for business, culture, and technical change to achieve their desired business outcomes.

The Cloud Adoption Framework provides technical guidance for Microsoft Azure. Because enterprise customers might still be in the process of choosing a cloud vendor or might have an intentional multicloud strategy, the framework provides cloud-agnostic guidance for strategic decisions whenever possible.

Intended audience

This guidance affects the business, technology, and culture of enterprises. The affected roles include line-of-business leaders, business decision makers, IT decision makers, finance, enterprise administrators, IT operations, IT security and compliance, IT governance, workload development owners, and workload operations owners. Each role uses its own vocabulary, and each has different objectives and key performance indicators. A single set of content can't address all audiences effectively.

Enter the *cloud architect*. The cloud architect serves as the thought leader and facilitator to bring these audiences together. We've designed this collection of guides to help cloud architects facilitate the right conversations with the right audiences and drive decision-making. Business transformation that's empowered by the cloud depends on the cloud architect role to help guide decisions throughout the business and IT.

Each section of the Cloud Adoption Framework represents a different specialization or variant of the cloud architect role. These sections also create opportunities to share cloud architecture responsibilities across a team of cloud architects. For example, the governance section is designed for cloud architects who have a passion for mitigating technical risks. Some cloud providers refer to these specialists as cloud custodians; we prefer the term *cloud guardian* or, collectively, the *cloud governance team*.

How to use the Microsoft Cloud Adoption Framework for Azure

If your enterprise is new to Azure, begin by understanding and documenting foundational alignment decisions. When your enterprise's digital transformation involves the cloud, understanding these fundamental concepts will help you during each step of the process.

Get started

What's new in the Microsoft Cloud Adoption Framework for Azure

8/2/2021 • 20 minutes to read • Edit Online

Here's a list of recent changes made to the Cloud Adoption Framework.

This framework is built collaboratively with customers, partners, and internal Microsoft teams. New and updated content is released when it becomes available. These releases allow you to test, validate, and refine the guidance along with us. We encourage you to partner with us to build the Cloud Adoption Framework.

July 2021

This release of the Cloud Adoption Framework includes updates to the Azure landing zone content in the Ready methodology.

- Expanded the What is a landing zone article to include a conceptual architecture and Azure landing zone accelerator, aimed at helping customers plan their cloud environments and get to best practice implementation quickly and easily
- Added new content for the Journey to the target architecture aimed at helping organizations determine which guidance is best suited to their current position on their cloud journey
- Updated the Azure landing zone design areas to help distinguish between considerations for implementation and considerations for operating a landing zone

June 2021

We've added a new retail industry scenario in this release, designed to guide a retail company's cloud adoption journey from beginning to end.

ARTICLE	DESCRIPTION
Retail industry scenario	Review a summary of the retail scenario components, common challenges, and supporting guidance.
Cloud adoption strategy for the retail industry	Learn about how the priorities of retailers are shifting, and the strategic impact of cloud adoption for the retail industry.
Cloud adoption plan for the retail industry	Use this guidance to develop a refined cloud adoption plan for the retail industry.
Innovation in the retail industry	Learn how to innovate even faster and improve your retail outcomes by using the steps outlined in the Innovate methodology.
Governance in the retail industry	Use the Cloud Adoption Framework methodology to systematically and incrementally improve governance of your cloud portfolio across the five governance disciplines. This article refines the methodology for governance needs specific to the retail industry.

ARTICLE	DESCRIPTION
Management in the retail industry	Learn how to use the Manage methodology in the Cloud Adoption Framework for Azure to manage the cloud for the retail industry.

May 2021

Cloud Adoption Framework Secure methodology

This release of the Cloud Adoption Framework introduces the new Secure methodology, which will centralize the theme of security best practices and provide end-to-end guidance for customers looking to adopt Microsoft guidance for their cloud environments. Security concerns have surged to the forefront of many top priorities lists recently, building on feedback requesting a more security-centric view, the Secure methodology will assist in defining the security aspect of the cloud journey.

Azure Virtual Desktop adoption journey

As part of the ongoing series of adoption journeys, this release includes expanded guidance for integrating Azure Virtual Desktop (formerly Windows Virtual Desktop) into your cloud adoption journey. This rich guidance shows how each methodology in the Cloud Adoption Framework helps provide structure and best practices to your implementation approach.

Cloud economics updates

This release continues the refresh of the Strategy methodology by integrating Microsoft's cloud economics program.

We've added new content that includes the financial and technical considerations to keep in mind as you develop your business case.

- Financial considerations
- Technical considerations
- Build a digital transformation timeline
- Create a business case
- Share your strategy

Modern application platform

We expanded and revised the modern application platform content in this release. Commonly, customers need to provide their developers with a consistent application platform. This article series outlines how you can use orchestrated containers to manage pools of workloads in one or more centralized clusters, thus providing that consistent application platform. The articles in this series for modern application platform combine technical and non-technical considerations required to prepare for Kubernetes and container integration into centralized operations and your broader cloud strategy.

April 2021

Cloud service controls

To help you implement cloud governance at a pace that sustains a high level of innovation, and implements the right level of control for your environment, we've added a new article, Implement cloud service controls. It focuses on a proven process to help governance teams implement granular controls across the Azure platform, while accelerating the development capabilities for cloud adoption teams. The process can be implemented by enterprises of any size, at any stage of their cloud journey. This guidance outlines the different stages of the process, from taking the enterprise requirements for cloud control for security, operations, and cost management, through the stages of development and release of controls.

Azure Storage migration overview

To help you plan your Azure Storage migration, we've provided guidance in a new article, Azure Storage migration overview, for two migration scenarios:

- Migration of unstructured data, such as files and objects.
- Migration of block-based devices, such as disks and storage area networks.

March 2021

This release is the largest update of the framework yet, adding a number of new broad-reaching collections of guidance that span across the entire framework.

Adoption journeys

Most notable in this release is the addition of adoption journeys, which provide a short, consumable overlay or lenses that rest on top of the deeper framework to accelerate engagement. These shorter guides demonstrate how to apply the guidance in the Cloud Adoption Framework, the Microsoft Azure Well-Architected Framework, the Azure Architecture Center, Microsoft Learn, and other Microsoft documentation to the adoption of specific technology platforms. The following table provides links to the overview page for each of the new journeys:

JOURNEY	DESCRIPTION
Hybrid and multicloud	Lifecycle guide to integrate hybrid, multicloud, and unified operations into your cloud adoption journey.
Modern containers	Container modernization enables rapid innovation and workload portability. Learn how to integrate containers into your cloud adoption journey.
SAP on Azure	As part of our One Migrate (migration scenarios) commitment, this journey bridges the gaps between the SAP migration process and other mainstream migration processes to deliver on full-scale adoption of SAP on Azure.

Cloud economics

Building on feedback and lessons learned, this is our first step to refresh the Strategy methodology by integrating Microsoft's cloud economics program.

We've updated the introduction to each category of business outcomes with references to case studies on cloud economics, demonstrating how organizations achieve the related business objective. The refreshed introductions with illustrative case studies include:

- Fiscal outcomes
- Agility outcomes
- Global reach outcomes
- Customer engagement outcomes
- Performance outcomes
- Sustainability goals

Enterprise-scale updates

The critical design area of network topology and connectivity includes new articles that simplify rationalizing individual components of your network design. Those design aspects now include guidance on connecting to multicloud providers like Oracle Cloud Infrastructure. We've also released the new enterprise-scale Terraform module to demonstrate Microsoft's continued investment in open-source approaches to Azure landing zone configuration. Lastly, we've updated guidance on how enterprises can optimize management groups and

organize subscriptions in Azure to meet cloud governance requirements.

Antipatterns

Companies often miss important steps in their cloud adoption journey. The new cloud adoption antipatterns guidance highlights common customer pain points, what skipped step led to that pain, and the fastest path to recovery. The antipatterns are distributed throughout each methodology, but a top 10 list is available in the getting started section of the framework.

AI + Machine Learning

When planning an Azure Machine Learning deployment for an enterprise environment, there are common decision points that affect how you create the workspaces. We've published a new article, Organize and set up Azure Machine Learning environments, that highlights these decision areas and the benefits of each:

- **Team structure:** How your Machine Learning teams are organized and collaborate on a project's given use case and data segregation, or cost management requirements.
- **Environments:** The environments used as part of your development and release workflow to segregate development from production.
- **Region:** The location of your data and the audience you need to serve your Machine Learning solution to.

We added a reference implementation example to help illustrate the deployment of Azure Machine Learning in a larger setting. It outlines how the organization 'Contoso' set up Azure Machine Learning given their organizational constraints, reporting, and budgeting requirements.

January 2021

To help you accelerate adoption and innovation, we've added new information about using GitHub and updated best practices for machine learning. We've published a new article and video to help you choose the best landing zone.

ARTICLE	DESCRIPTION
How GitHub accelerates cloud adoption	This article outlines the benefits of using GitHub to accelerate cloud adoption by taking advantage of open-source assets, collaborative development environments, automation and security features.
Best practices for machine learning	We've updated and expanded the best practice guidance for machine learning. Included in the best practices are: • How to approach machine learning operations and the MLOps process • Machine learning security • Machine learning inference and deployment • Determine compute instances for your model • Set up machine learning workspaces • Responsible and trusted AI
Choose a landing zone option	Microsoft offers two implementation options for landing zones: *Start small and expand* and *Enterprise-scale*. Use this new article to review both options and choose the right approach for your organization.

December 2020

We've revised the naming and tagging strategy guidance, as well as adding guidance for Moodle migration scenarios.

ARTICLE	DESCRIPTION
Develop your naming and tagging strategy for Azure resources	We've refined the guidance for defining your naming and tagging strategy. In addition to the overview, we've separated this guidance into multiple articles that cover these topics: • Define your naming convention • Recommended abbreviations for various Azure resource types • Define your tagging strategy
Migrate a Moodle deployment to Azure	Learn how to migrate a deployment of the Moodle open-source learning management system from an on-premises environment to Azure. Steps are provided for using either the Azure portal or the Azure CLI for your deployment.

October 2020

This month's updates include incremental improvements throughout the Cloud Adoption Framework and supporting web assets.

Our biggest investments have focused on building out Microsoft Learn modules to accelerate application of the Cloud Adoption Framework. This month, we released the modules listed below. Note that the Getting Started module provides our first guidance aligned with an industry vertical, by introducing a retail customer (Tailwind Traders) who we will follow through all of the core methodology modules to follow.

MODULE	DESCRIPTION
Overview module	Entry-level introduction to the framework.
Getting Started module	Introduction to the getting started guides to accelerate application of the proper methodologies for overcoming specific blockers.
Azure landing zones	Before building out your cloud environment, understand your operating requirements and choose the most appropriate Azure landing zone product to get you started.
Create an enterprise-scale architecture	Create landing zones at scale following a set of enterprise-scale design principles, reference architectures, and reference implementations. Four modules to create a single learning path to success.

We've also expanded the business outcomes to share a number of common business motivations and approaches that continue to emerge in the post-COVID marketplace.

ARTICLE	DESCRIPTION
Examples of sustainability outcomes	Learn how cloud computing can help you reduce carbon emissions, use resources more efficiently, and shrink your environmental footprint.
Measure business outcomes with objectives and key results (OKRs)	Learn how to use OKRs to measure business outcomes.

ARTICLE	DESCRIPTION
Measure business outcomes with AppDynamics	Understanding an application's performance and user experience is key to measuring business outcomes. See how AppDynamics can provide business insights for most use cases.
Cost management update: Spot VMs	Use of Spot VMs in nonproduction environments is a rapidly emerging practice to further reduce costs in those environments.existing environments. "I already have a working environment. How can I apply the design principles of enterprise-scale?" The new article on transitioning to enterprise-scale can help.

ARTICLE	DESCRIPTION
Transition existing Azure environments to enterprise-scale	This article helps organizations to navigate the right path based on an existing Azure environment transitioning into enterprise-scale.
Cloud Adoption Framework enterprise-scale landing zone architecture	This article was updated to include a high-level diagram for an enterprise-scale landing zone architecture based on the hub and spoke network topology, and updates to describe and cross-reference the critical design areas for an enterprise-scale landing zone architecture.

August 25, 2020

This release provides better definition and decision criteria regarding landing zone implementations.

Operating model

One of the most important considerations in landing zone design and implementation is your operating model. How you want to operate in the cloud will have a direct impact on the architecture and controls to implement. The following articles will help align your target operating model with a few models that are common in the cloud. Then map those to the most appropriate implementation to get started.

ARTICLE	DESCRIPTION
Compare common operating models	This article is the primary guide for comparing operating models and choosing a course of action.
Understand cloud operating models	Primer for making import decisions regarding your operating model.
Define your operating model with CAF	The Cloud Adoption Framework is an incremental guide to building out your environment and adopting the cloud within your chosen operating model. This article creates a frame of reference to understand how the various methodologies support the development of your operating model.
Terms	Terms that are likely to come up when discussing your operating model with counterparts. These terms are not as commonly used by architects or technical specialists, but will prove important in those conversations.

Azure landing zones: Additional implementation options

The concept and implementations options behind Azure landing zones was built alongside the leading Microsoft partners. This release recognizes the existing intellectual property (IP) that those partners use to accelerate cloud adoption.

ARTICLE	DESCRIPTION
Partner landing zones	Review and compare Azure landing zone offers from your partner.
Implementation options	Updated to add partner landing zone options to the existing Azure landing zone implementation options.
Enterprise-scale reference implementations	Updated to add a hub-spoke reference implementation to enterprise-scale reference implementations.

> **NOTE**
> The new partner landing zone articles don't specify how a partner should define or implement a landing zone. Instead, it's designed to add structure to a complex conversation, so you can better understand the partner offer. This list of questions and minimum evaluation criteria can also be used to compare offers from potential partners. It's also being used by some partners to more clearly communicate the value of their Azure landing zone implementation options.

July 17, 2020

This release adds a number of new scenarios to make cloud adoption more actionable.

Migration scenarios:

The new migration scenarios overview page builds on the Migrate methodology to demonstrate how Azure delivers on the One Migrate promise. It provides approaches to migrating multiple first and third-party scenarios to Azure. This includes new migration scenarios:

ARTICLE	DESCRIPTION
Azure Virtual Desktop	This scenario enables productivity boosts and accelerates the migration of various workloads to support the end-user experience.
Azure Stack	Learn about deploying Azure in your datacenter using Azure Stack Hub.

Analytics in the Cloud Adoption Framework:

Analytics solutions are now included in the Microsoft Cloud Adoption Framework. These new topics highlight best practices for enabling analytics solutions during your cloud adoption journey.

ARTICLE	DESCRIPTION
Analytics solution for Teradata, Netezza, Exadata	Learn about migrating legacy on-premises environments including Teradata, Netezza, and Exadata to modern analytics solutions.
High availability for Azure Synapse	Learn about one of the key benefits of a modern cloud-based infrastructure, built-in high availability and disaster recovery.

ARTICLE	DESCRIPTION
Schema migration data definition languages (DDL)	Learn about the database objects and associated processes when preparing to migrate existing data.

AI in the Cloud Adoption Framework:

AI solutions and best practices are now integrated into the Microsoft Cloud Adoption Framework. These AI solutions can help accelerate innovation with predictions about customer's needs, automate business processes, discover information, find new ways to engage with customers, and deliver better experiences during your cloud adoption journey.

ARTICLE	DESCRIPTION
Responsible AI	Learn about the AI principles you should consider when implementing AI solutions and learn how to establish a responsible AI strategy.
Azure innovation guide: innovate with AI	Learn about how you can innovate with AI and find the best solution based on your implementation needs.
AI in the Cloud Adoption Framework	Review a prescriptive framework that includes the tools, programs, and content (best practices, configuration templates, and architecture guidance) to simplify adoption of AI and cloud-native practices at scale.
MLOps with Azure Machine Learning	Learn about machine learning operations (MLOps) best practices.
Innovate with AI	Learn about AI solutions (machine learning, AI applications and agents, knowledge mining) and best practices that can accelerate digital invention.

June 15, 2020

Proper configuration of the cloud environment is often the first and most common technical blocker during cloud adoption. This release focuses heavily on guidance that accelerates deployment of cloud environments. To overcome this common blocker, the Cloud Adoption Framework introduces **Azure landing zones**.

ARTICLE	DESCRIPTION
Azure landing zones	Azure landing zones create a common set of design areas and implementation options to accelerate environment creation aligned to the cloud adoption plan and cloud operating model. This new article defines Azure landing zones more clearly.
Azure landing zones: design areas	All Azure landing zones share a common set of design areas. Before deploying any of the Azure landing zones, customers should consider these design areas to make critical decisions.
Azure landing zones: implementation options	Choose the best Azure landing zone implementation option, depending on your cloud adoption plan and cloud operating model.

The existing CAF blueprint definitions and CAF Terraform modules provide a starting point for Azure landing

zone implementation. However, some customers need a richer implementation option that can meet the demands of enterprise-scale cloud adoption plans. This release adds **CAF enterprise-scale** to the Azure landing zone implementation options to fill that need. The following lists a few of the articles to get you started with the CAF enterprise-scale architecture and reference implementations.

ARTICLE	DESCRIPTION
Enterprise-scale overview	Overview to enterprise-scale
Implement CAF enterprise-scale landing zones	Rapid implementation options and GitHub examples
Enterprise-scale architecture	Understand the architecture behind enterprise-scale
Enterprise-scale design principles	Understand the architectural design principles that guide decisions during implementation to evaluate whether this approach fits your cloud operating model
Enterprise-scale design guideline	Evaluate the enterprise-scale guidelines for fulfilling the common design areas of Azure landing zones
Implementation details	Review the activities required for an enterprise-scale implementation before deployment

Partners are an important aspect of successful cloud adoption. Throughout the Cloud Adoption Framework guidance, we have added references to show the important role that partners play and how customers can better engage partners. For a list of validated CAF partners, see the CAF-aligned partner offers, Azure expert managed service providers (MSPs), or advanced specialist partners.

May 15, 2020

Based on feedback, we've created new content to get you started using the Cloud Adoption Framework. The new getting started guides help you navigate the framework based on what you want to accomplish. We've also create a new landing page to make it easier to find the guidance, tools, learn modules and programs that support a successful cloud adoption journey.

ARTICLE	DESCRIPTION
Cloud Adoption Framework for Azure	The Cloud Adoption Framework landing page has been redesigned to make it easier to find the guidance, tools, learn modules and programs that support a successful cloud adoption journey.
Get started with the Cloud Adoption Framework	Choose a getting started guide that's aligned with your cloud adoption goals. These common scenarios provide a roadmap through the Microsoft Cloud Adoption Framework for Azure.
Understand and document foundational alignment decisions	Learn about the initial decisions that every team involved in cloud adoption should understand.
Understand and align the portfolio hierarchy	Learn how a portfolio hierarchy shows how your workloads and supporting services all fit together.
How do Azure products support the portfolio hierarchy?	Learn about the Azure tools and solutions that support your portfolio hierarchy.

ARTICLE	DESCRIPTION
Manage organizational alignment	Establish well-staffed organizational structures that an effective operating model for the cloud.

April 14, 2020

We've brought all the cloud adoption tools and templates together in one place to make them easier to find.

ARTICLE	DESCRIPTION
Tools and templates	Find the tools, templates, and assessments that can help you accelerate your cloud adoption journey.

April 4, 2020

Continued iteration of refinement to the Migrate methodology and the Ready methodology, to more tightly align them with feedback from Microsoft customers, partners, and internal programs.

Migrate methodology updates:

ARTICLE	DESCRIPTION
Migrate methodology	These changes streamline the phases of the migration effort (assess workloads, deploy workloads, and release workloads). The changes also remove the details regarding the migration backlog. Removing those details and referencing Plan, Ready, and Adopt methodologies instead creates flexibility for various different cloud adoption programs to better align with the methodology.

Ready methodology updates:

ARTICLE	DESCRIPTION
Refactor landing zones	**New article:** Drawing from Ready methodology workshops, this article demonstrates the theory of starting with an initial template, using decision trees and refactoring to expand the landing zone, and moving toward a future state of enterprise readiness.
Expand your landing zone	**New article:** Builds on the parallel iterations section of the refactoring article to show how various types of landing zone expansions would embed shared principles into the supporting platform. The original content for this overview has been moved to the basic landing zone considerations node in the table of contents.
Test-driven development (TDD) for landing zones	**New article:** The refactoring approach is much improved through the adoption of a test-driven development cycle to guide landing zone development and refactoring.
Landing zone TDD in Azure	**New article:** Azure governance tools provide a rich platform for TDD cycles or red/green tests.

ARTICLE	DESCRIPTION
Improve landing zone security	**New article:** Overview of the best practices in this section, related back to the TDD cycle.
Improve landing zone operations	**New article:** List of best practices in the Manage methodology, with a transition to that modular approach to improving operations, reliability, and performance.
Improve landing zone governance	**New article:** List of best practices related to Govern methodology, with a transition to that modular approach to improving governance, cost management, and scale.
Start with enterprise-scale	**New article:** Demonstrate an approach that shows the differences in the process, when a customer starts with CAF enterprise-scale landing zone templates. This article helps customers understand qualifiers that would support this decision.

March 27, 2020

We've added guidance about the initial subscriptions you should create when you adopt Azure.

Subscription guidance updates:

ARTICLE	DESCRIPTION
Create your initial Azure subscriptions	**New article:** Create your initial production and nonproduction subscriptions, and decide whether to create sandbox subscriptions, as well as a subscription to contain shared services.
Create additional subscriptions to scale your Azure environment	Learn about reasons to create additional subscriptions, moving resources between subscriptions, and tips for creating new subscriptions.
Organize and manage multiple Azure subscriptions	Create a management group hierarchy to help organize, manage, and govern your Azure subscriptions.

March 20, 2020

We've added prescriptive guidance that includes the tools, programs, and content categorized by persona to drive successful deployment of applications on Kubernetes, from proof of concept to production, followed by scaling and optimization.

Kubernetes:

ARTICLE	DESCRIPTION
Application development and deployment	**New article:** Provides checklists, resources, and best practices for planning application development, configuring CI/CD pipelines, and implementing site reliability engineering for Kubernetes.

ARTICLE	DESCRIPTION
Cluster design and operations	**New article:** Provides checklists, resources, and best practices for cluster configuration, network design, future-proof scalability, business continuity, and disaster recovery for Kubernetes.
Cluster and application security	**New article:** Provides checklists, resources, and best practices for Kubernetes security planning, production, and scaling.

March 2, 2020

In response to feedback about continuity in the migration approach through multiple sections of the Cloud Adoption Framework, including Strategy, Plan, Ready, and Migrate, we've made the following updates. These updates are designed to make it easier for you to understand planning and adoption refinements as you continue a migration journey.

Strategy methodology updates:

ARTICLE	DESCRIPTION
Balance the portfolio	Moved this article to appear earlier in the Strategy methodology. This gives you visibility into the thought process earlier in the lifecycle.
Balancing competing priorities	**New article:** Outlines the balance of priorities across methodologies to help inform your strategy.

Plan methodology updates:

ARTICLE	DESCRIPTION
Assessment best practice	Moved this article to the new "best practices" section of the Plan methodology. This gives you visibility into the practice of assessing local environments earlier in the lifecycle.

Ready methodology updates:

ARTICLE	DESCRIPTION
What is a landing zone?	**New article:** Defines the term landing zone.
First landing zone	**New article:** Expands on the comparison of various landing zones.
CAF Migration landing zone	Separated the blueprint definition from the selection of the first landing zone.
CAF Terraform modules	Moved to the new "landing zone" section of the Ready methodology, to elevate Terraform in the landing zone conversation.

Migrate methodology updates:

ARTICLE	DESCRIPTION
Overview	Updated with a clearer description of the guide and fewer steps.
Assess	Added a "challenging assumptions" section to demonstrate how this level of assessment works with the incremental assessment approach mentioned in the Plan methodology.
Classification during assess processes	**New article:** Outlines the importance of classifying every asset and workload prior to migration.
Migrate	Added a reference to UnifyCloud in the third-party tool options, in response to feedback at tier 1 conferences.
Test, optimize, and promote	Aligned the title of this article with other process improvement suggestions.
Assess overview	Updated to illustrate that the assessment in this phase focuses on assessing the technical fit of a specific workload and related assets.
Planning checklist	Updated to clarify the importance of operations alignment during planning for migration efforts to ensure a well-managed workload following migration.

Get started with the Cloud Adoption Framework

7/29/2021 • 3 minutes to read • Edit Online

The Cloud Adoption Framework can help you get started with several different getting started guides. This article groups the guides to help you find the one that best aligns with your current challenges.

Each of the following links takes you to questions that are typically asked when an organization is trying to accomplish a certain goal during their cloud adoption journey.

- Choose the cloud adoption scenario that best supports your strategy
- Examine antipatterns across methodologies and their solutions
- Align foundational concepts to onboard a person, project, or team
- Adopt the cloud to deliver business and technical outcomes sooner
- Improve controls to ensure proper operations of the cloud
- Establish teams to support adoption and operations

Cloud adoption scenarios

Your organization's cloud adoption effort should support long-term strategic goals for your cloud journey. Depending on whether you're considering a comprehensive hybrid and multicloud effort, preparing for Kubernetes and container integration into your cloud strategy, or designing a cloud adoption journey for a retail business, we have updated guidance for these scenarios and more.

- Hybrid and multicloud adoption scenario
- Modern application platform scenario
- SAP adoption scenario
- Desktop virtualization
- Cloud adoption for the retail industry

Cloud adoption antipatterns

You might encounter missteps with design, planning, or implementation when migrating to the cloud. We have updated detailed guidance on antipatterns, that can block innovation and prevent businesses from adopting and realizing goals.

Align foundation

Your company's cloud adoption journey is guided with foundational decisions that affect the outcomes of your cloud adoption journey. The following information can help you make core decisions, and record them as a reference to be used during the cloud adoption lifecycle.

- Get started aligning foundation decisions
- How does Azure work
- Fundamental concepts
- Portfolio hierarchy
- Azure hierarchy support

Accelerate adoption

Cloud adoption requires technical change, but to digitally transform with the cloud, it requires more than just IT. Use these guides to start aligning various teams to accelerate migration and innovation efforts.

GUIDE	DESCRIPTION
We want to migrate existing workloads to the cloud.	This guide is a great starting point if your primary focus is migrating on-premises workloads to the cloud.
We want to build new products and services in the cloud.	This guide can help you prepare to deploy innovative solutions to the cloud.
We're blocked by environment design and configuration.	This guide provides a quick approach to designing and configuring your environment.

Improve controls

As your cloud adoption journey progresses, a solid operating model can help ensure that wise decisions are made. You'll also want to consider organizational change. These guides help you align people and improve operations to develop your cloud operating model.

GUIDE	DESCRIPTION
How do we deliver operational excellence during cloud transformation?	The steps in this guide can help the strategy team lead the organizational change management required to consistently ensure operational excellence.
How do we manage enterprise costs?	This guide can help you start optimizing enterprise costs and manage cost across the environment.
How do we consistently secure the enterprise cloud environment?	This guide can help ensure that the security requirements are applied across the enterprise to minimize risk of breach, and to accelerate recovery when a breach occurs.
How do we apply the right controls to improve reliability?	This guide helps minimize disruptions related to inconsistencies in configuration, resource organization, security baselines, or resource protection policies.
How do we ensure performance across the enterprise?	This guide can help you establish processes for maintaining performance across the enterprise.

Establish teams

Depending on your adoption strategy and operating model, you might need to establish a few teams. This section helps you get those new teams started.

GUIDE	DESCRIPTION
How do we align our organization?	This guide can help you establish an appropriately staffed organizational structure.
Do I need a cloud strategy team?	This team ensures that cloud adoption efforts progress in alignment with business outcomes.
What does a cloud adoption team do?	This team implements technical solutions outlined in the plan, and in accordance with governance requirements.

GUIDE	DESCRIPTION
How do I build a cloud governance team?	This team ensure that risks and risk tolerance are properly evaluated and managed.
How does a cloud operations team work?	This team focuses on monitoring, repairing, and the remediation of issues related to traditional IT operations and assets.

Get started: Understand and document foundational alignment decisions

7/21/2021 • 6 minutes to read • Edit Online

The cloud adoption journey can unlock many business, technical, and organizational benefits. Whatever you want to accomplish, if your journey involves the cloud, there are a few initial decisions that every team involved should understand.

> **NOTE**
>
> Selecting any of the following links might lead you to bounce around the table of contents for the Microsoft Cloud Adoption Framework for Azure, looking for fundamental concepts that you'll use later to help the team implement the associated guidance. Bookmark this page to come back to this checklist often.

Before you begin

As you work through this guide, record our foundational decisions using the initial decision template. The template can help you quickly onboard team members who participate in the cloud adoption lifecycle by clarifying how your cloud environment is configured and why.

If you already have an environment running in Azure, the Azure governance visualizer can help you accelerate your documentation. Gain insight into policies, Azure role-based access control (Azure RBAC), Azure Blueprints, subscriptions, and more. From the collected data the tool provides visibility on your hierarchy map, creates a tenant summary, and builds granular scope insights about management groups and subscriptions.

Step 1: Understand how Azure works

If you've chosen Azure as a cloud provider to support your cloud adoption journey, it's important to understand how Azure works.

Involved teams, deliverables, and supporting guidance:

Everyone involved in the cloud adoption lifecycle should have a basic understanding of how Azure works.

Step 2: Understand initial Azure concepts

Azure is built on a set of foundational concepts that are required for a deep conversation about the technical strategy for Azure implementations.

Involved teams, deliverables, and supporting guidance:

Everyone involved in Azure implementation of the technology strategy should understand the terms and definitions in the foundational concepts.

Step 3: Review the portfolio

Whatever cloud provider you choose, all cloud hosting and environmental decisions start with an understanding of the portfolio of workloads. The Cloud Adoption Framework includes a few tools for understanding and evaluating the portfolio.

Deliverables:

- Record the location, status, and accountable person for the portfolio documentation in the initial decision template.

Guidance to support deliverable completion:

- Fundamental concepts help you understand key Azure topics before you begin your cloud adoption.
- The operations management workbook and business alignment approach help you understand the workloads and assets that have been transitioned to a cloud operations team.
- The cloud adoption plan provides a backlog of the workloads and assets that are slated for adoption in the cloud.
- Digital estate analysis is an approach to documenting existing workloads and assets that are slated for adoption in the cloud. In Azure, the digital estate is best represented in a tool called Azure Migrate.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• The cloud strategy team is accountable for defining a way to view the portfolio.	• Multiple teams will use the following guidance to create those views. Everyone involved in cloud adoption should know where to find the portfolio view to support decisions later in the process.

Step 4: Define portfolio-hierarchy depth to align the portfolio

Hosting assets and workloads in the cloud can be simple, consisting of a single workload and its supporting assets. For other customers, the cloud adoption strategy might include thousands of workloads and many more supporting assets. The portfolio hierarchy gives common names for each level to help create a common language for organization, regardless of the cloud provider.

Deliverables:

- Record the relevant hierarchy needs in the initial decision template.

Guidance to support deliverable completion:

- Understand the levels of the portfolio hierarchy to align fundamental terms.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• The cloud governance team is accountable for defining, enforcing, and automating the portfolio hierarchy to shape corporate policy in the cloud.	• Everyone involved in the technical strategy for cloud adoption should be familiar with the portfolio hierarchy and the levels of the hierarchy in use today.

Step 5: Establish a naming and tagging standard across the portfolio

All existing workloads and assets should be properly named and tagged in accordance with a naming and tagging standard. Those standards should be documented and available as a reference for all team members. When possible, the standards should also be automatically enforced to ensure minimum tagging requirements.

Deliverables:

- Record the location, status, and accountable party for the naming and tagging conventions workbook in the initial decision template.

Guidance to support deliverable completion:

- Create a naming and tagging standard.
- Populate the naming and tagging conventions tracking template to track decisions.
- Review and update existing tags in Azure.
- Enforce tagging policies in Azure.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• The cloud governance team is accountable for defining, enforcing, and automating the naming and tagging standards to ensure consistency across the portfolio.	• Everyone involved in the technical strategy for cloud adoption should be familiar with the naming and tagging standards before deployment to the cloud.

Step 6: Create a resource organization design to implement the portfolio hierarchy

To ensure consistent alignment with the portfolio hierarchy decisions, it's important to create a design for resource organization. Such a design aligns organizational tools from the cloud provider with the portfolio hierarchy required to support your cloud adoption plan. This design will guide implementation by clarifying which assets can be deployed into specific boundaries within the cloud environments.

Deliverables:

- Map Azure products to the aligned level of the portfolio hierarchy in the initial decision template.

Guidance to support deliverable completion:

- Understand how Azure products support the portfolio hierarchy.
- Review existing subscriptions for alignment to the chosen portfolio hierarchy.

Build a subscription strategy:

- Start with Two subscriptions by design. Add basic subscription designs to account for common enterprise needs, like shared services or sandbox subscriptions.
- Manage multiple subscriptions as additional subscriptions are required to support the cloud adoption plan.
- Establish clear boundaries based on the portfolio hierarchy.
- When required, move resource groups and assets between subscriptions to adhere to the organization strategy.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• The cloud governance team is accountable for defining, implementing, and automating the resource organization design across the portfolio.	• Everyone involved in the technical strategy for cloud adoption should be familiar with the resource organization design before deployment to the cloud.

Step 7: Map capabilities, teams, and RACI to fundamental concepts

Complexity of the portfolio hierarchy will help inform organizational structures and methodologies to guide the day-to-day activities of various teams.

Deliverables:

- Complete the getting started guides for organizational alignment based on these concepts.

Guidance to support deliverable completion:

- Use the prior steps as a guide to evaluate the portfolio hierarchy accountability guidance. Determine which capabilities might need to be delivered by dedicated organizations or virtual teams.
- Use Get started: Align your organization to apply the portfolio hierarchy accountability guidance to the RACI (responsible, accountable, consulted, and informed) diagram.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• The cloud strategy team is accountable for aligning virtual or dedicated organizational structures to ensure success of the cloud adoption lifecycle.	• Everyone involved in the cloud adoption lifecycle should be familiar with the alignment of people and levels of accountability.

What's next

Build on this set of fundamental concepts through the series of getting started guides in this section of the Cloud Adoption Framework.

Apply fundamental concepts to other getting started guides

How does Azure work?

7/15/2021 • 2 minutes to read • Edit Online

Azure is Microsoft's public cloud platform. Azure offers a large collection of services including platform as a service (PaaS), infrastructure as a service (IaaS), and managed database service capabilities. But what exactly is Azure, and how does it work?

Azure, like other cloud platforms, relies on a technology known as *virtualization*. Most computer hardware can be emulated in software, because most computer hardware is simply a set of instructions permanently or semi-permanently encoded in silicon. Using an emulation layer that maps software instructions to hardware instructions, virtualized hardware can execute in software as if it were the actual hardware itself.

Essentially, the cloud is a set of physical servers in one or more datacenters that execute virtualized hardware on behalf of customers. So how does the cloud create, start, stop, and delete millions of instances of virtualized hardware for millions of customers simultaneously?

To understand this, let's look at the architecture of the hardware in the datacenter. Inside each datacenter is a collection of servers sitting in server racks. Each server rack contains many server *blades* as well as a network switch providing network connectivity and a power distribution unit (PDU) providing power. Racks are sometimes grouped together in larger units known as *clusters*.

Within each rack or cluster, most of the servers are designated to run these virtualized hardware instances on behalf of the user. But some of the servers run cloud management software known as a fabric controller. The *fabric controller* is a distributed application with many responsibilities. It allocates services, monitors the health of the server and the services running on it, and heals servers when they fail.

Each instance of the fabric controller is connected to another set of servers running cloud orchestration software, typically known as a *front end*. The front end hosts the web services, RESTful APIs, and internal Azure databases used for all functions the cloud performs.

For example, the front end hosts the services that handle customer requests to allocate Azure resources such as virtual machines, and services like Azure Cosmos DB. First, the front end validates the user and verifies the user is authorized to allocate the requested resources. If so, the front end checks a database to locate a server rack with sufficient capacity and then instructs the fabric controller on that rack to allocate the resource.

So fundamentally, Azure is a huge collection of servers and networking hardware running a complex set of distributed applications to orchestrate the configuration and operation of the virtualized hardware and software on those servers. It is this orchestration that makes Azure so powerful, because users are no longer responsible for maintaining and upgrading hardware because Azure does all this behind the scenes.

Next steps

Learn more about cloud adoption with the Microsoft Cloud Adoption Framework for Azure.

Microsoft Cloud Adoption Framework for Azure

Azure fundamental concepts

5/18/2021 • 5 minutes to read • Edit Online

Learn fundamental concepts and terms that are used in Azure, and how the concepts relate to one another.

Azure terminology

It's helpful to know the following definitions as you begin your Azure cloud adoption efforts:

- **Resource:** An entity that's managed by Azure. Examples include Azure Virtual Machines, virtual networks, and storage accounts.
- **Subscription:** A logical container for your resources. Each Azure resource is associated with only one subscription. Creating a subscription is the first step in adopting Azure.
- **Azure account:** The email address that you provide when you create an Azure subscription is the Azure account for the subscription. The party that's associated with the email account is responsible for the monthly costs that are incurred by the resources in the subscription. When you create an Azure account, you provide contact information and billing details, like a credit card. You can use the same Azure account (email address) for multiple subscriptions. Each subscription is associated with only one Azure account.
- **Account administrator:** The party associated with the email address that's used to create an Azure subscription. The account administrator is responsible for paying for all costs that are incurred by the subscription's resources.
- **Azure Active Directory (Azure AD):** The Microsoft cloud-based identity and access management service. Azure AD allows your employees to sign in and access resources.
- **Azure AD tenant:** A dedicated and trusted instance of Azure AD. An Azure AD tenant is automatically created when your organization first signs up for a Microsoft cloud service subscription like Microsoft Azure, Intune, or Microsoft 365. An Azure tenant represents a single organization.
- **Azure AD directory:** Each Azure AD tenant has a single, dedicated, and trusted directory. The directory includes the tenant's users, groups, and applications. The directory is used to perform identity and access management functions for tenant resources. A directory can be associated with multiple subscriptions, but each subscription is associated with only one directory.
- **Resource groups:** Logical containers that you use to group related resources in a subscription. Each resource can exist in only one resource group. Resource groups allow for more granular grouping within a subscription, and are commonly used to represent a collection of assets required to support a workload, application, or specific function within a subscription.
- **Management groups:** Logical containers that you use for one or more subscriptions. You can define a hierarchy of management groups, subscriptions, resource groups, and resources to efficiently manage access, policies, and compliance through inheritance.
- **Region:** A set of Azure datacenters that are deployed inside a latency-defined perimeter. The datacenters are connected through a dedicated, regional, low-latency network. Most Azure resources run in a specific Azure region.

Azure subscription purposes

An Azure subscription serves several purposes. An Azure subscription is:

- **A legal agreement.** Each subscription is associated with an Azure offer, such as a free trial or pay-as-you-go. Each offer has a specific rate plan, benefits, and associated terms and conditions. You choose an Azure offer when you create a subscription.

- **A payment agreement.** When you create a subscription, you provide payment information for that subscription, such as a credit card number. Each month, the costs incurred by the resources deployed to that subscription are calculated and billed via that payment method.
- **A boundary of scale.** Scale limits are defined for a subscription. The subscription's resources can't exceed the set scale limits. For example, there's a limit on the number of virtual machines that you can create in a single subscription.
- **An administrative boundary.** A subscription can act as a boundary for administration, security, and policy. Azure also provides other mechanisms to meet these needs, such as management groups, resource groups, and Azure role-based access control.

Azure subscription considerations

When you create an Azure subscription, you make several key choices about the subscription:

- **Who is responsible for paying for the subscription?** The party associated with the email address that you provide when you create a subscription by default is the subscription's account administrator. The party associated with this email address is responsible for paying for all costs that are incurred by the subscription's resources.
- **Which Azure offer am I interested in?** Each subscription is associated with a specific Azure offer. You can choose the Azure offer that best meets your needs. For example, if you intend to use a subscription to run nonproduction workloads, you might choose the Pay-As-You-Go Dev/Test offer or the Enterprise Dev/Test offer.

> **NOTE**
>
> When you sign up for Azure, you might see the phrase *create an Azure account*. You create an Azure account when you create an Azure subscription and associate the subscription with an email account.

Azure administrative roles

Azure defines three types of roles for administering subscriptions, identities, and resources:

- Classic subscription administrator roles
- Azure roles
- Azure Active Directory (Azure AD) roles

The account administrator role for an Azure subscription is assigned to the email account that's used to create the Azure subscription. The account administrator is the billing owner of the subscription. The account administrator can manage subscription administrators via the Azure portal.

By default, the Service Administrator role for a subscription also is assigned to the email account that's used to create the Azure subscription. The Service Administrator has permissions to the subscription equivalent to the Azure RBAC-based Owner role. The Service Administrator also has full access to the Azure portal. The account administrator can change the Service Administrator to a different email account.

When you create an Azure subscription, you can associate it with an existing Azure AD tenant. Otherwise, a new Azure AD tenant with an associated directory is created. The role of global administrator in the Azure AD directory is assigned to the email account that's used to create the Azure AD subscription.

An email account can be associated with multiple Azure subscriptions. The account administrator can transfer a subscription to another account.

For a detailed description of the roles defined in Azure, see Classic subscription administrator roles, Azure roles, and Azure AD roles.

Subscriptions and regions

Every Azure resource is logically associated with only one subscription. When you create a resource, you choose which Azure subscription to deploy that resource to. You can move a resource to another subscription later.

While a subscription isn't tied to a specific Azure region, each Azure resource is deployed to only one region. You can have resources in multiple regions that are associated with the same subscription.

> **NOTE**
> Most Azure resources are deployed to a specific region. Certain resource types are considered global resources, such as policies that you set by using the Azure Policy services.

Related resources

The following resources provide detailed information about the concepts discussed in this article:

- How does Azure work?
- Resource access management in Azure
- Azure Resource Manager overview
- Azure role-based access control (Azure RBAC)
- What is Azure Active Directory?
- Associate or add an Azure subscription to your Azure Active Directory tenant
- Topologies for Azure AD Connect
- Subscriptions, licenses, accounts, and tenants for Microsoft's cloud offerings

Next steps

Now that you understand fundamental Azure concepts, learn how to scale with multiple Azure subscriptions.

Scale with multiple Azure subscriptions

Understand and align the portfolio hierarchy

5/18/2021 • 11 minutes to read • Edit Online

Business needs are often supported, improved, or accelerated through information technology. A collection of technologies that delivers defined business value is called a *workload*. That collection might include applications, servers or virtual machines, data, devices, and other similarly grouped assets.

Typically, a business stakeholder and technical leader share accountability for the ongoing support of each workload. In some phases of the workload lifecycle, those roles are clearly stated. In more operational phases of a workload's lifecycle, those roles might be transitioned to a shared operations management team or cloud operations team. As the number of workloads increases, the roles (stated or implied) become more complex and more matrixed.

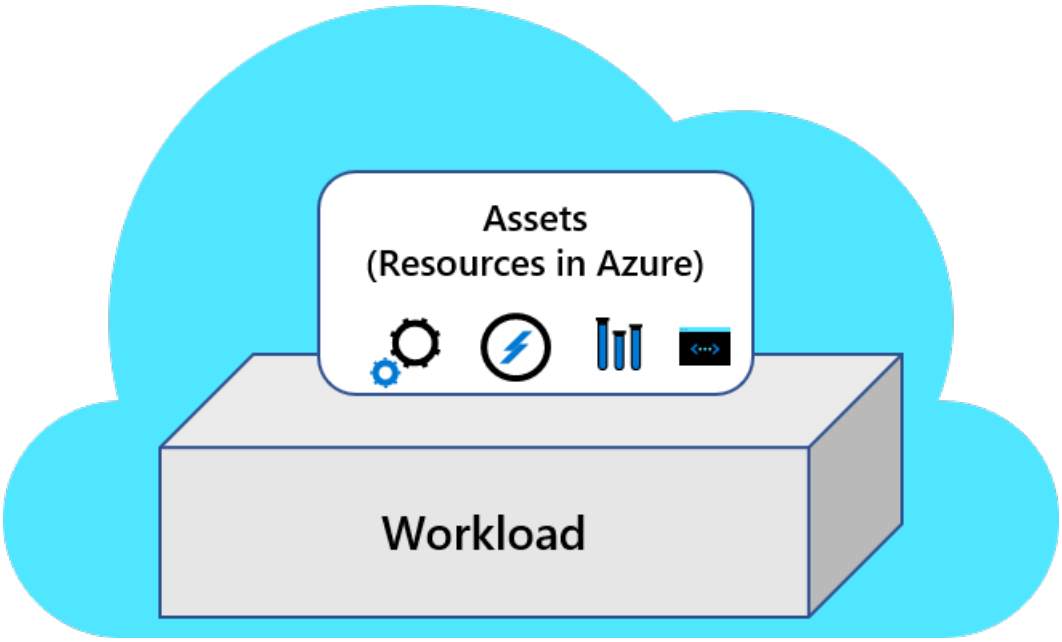
Most businesses rely on multiple workloads to deliver vital business functions. The collection of workloads, assets, and supporting factors (projects, people, processes, and investments) is called a *portfolio*. The matrix of business, development, and operations staff requires a portfolio hierarchy to show how the workloads and supporting services all fit together.

This article provides clear definitions for the levels of the portfolio hierarchy. The article aligns various teams with the appropriate accountability in each layer, along with the source of the best guidance for that team to deliver on the expectations for that level. Throughout this article, each level of the hierarchy is also called a *scope*.

Portfolio hierarchy

Workloads

Workloads and their supporting assets are at the core of any portfolio. The additional scopes or layers below define how those workloads are viewed and to what extent they're affected by the matrix of potential supporting teams.



Although the terms can vary, all IT solutions include assets and workloads:

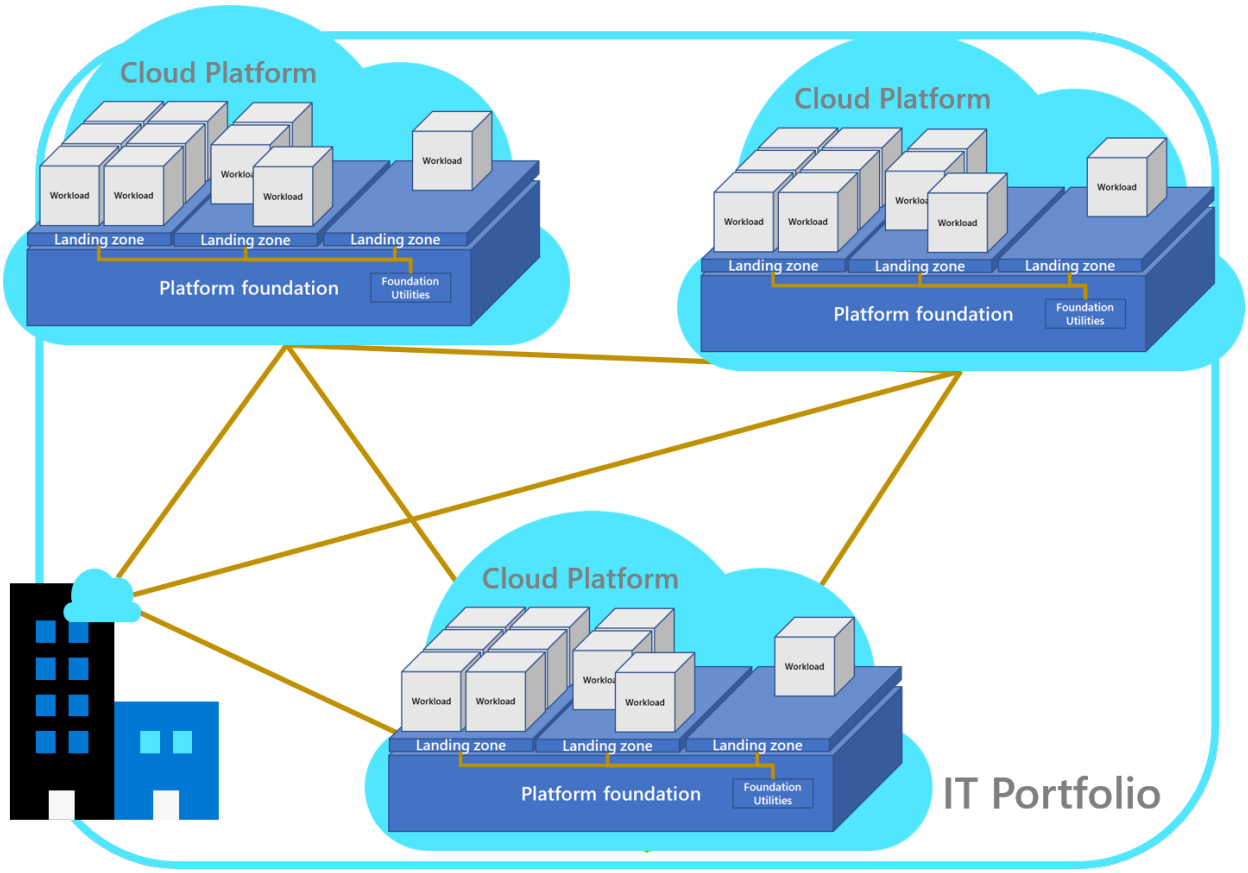
- **Asset:** The smallest unit of technical function that supports a workload or solution.
- **Workload:** The smallest unit of IT support for the business. A workload is a collection of assets

(infrastructure, applications, and data) that supports a common business goal or the execution of a common business process.

When you're deploying your first workload, the workload and its assets might be the only defined scope. The other layers might be explicitly defined as more workloads are deployed.

IT portfolio

When companies support workloads through matrixed approaches or centralized approaches, a broader hierarchy likely exists to support those workloads:



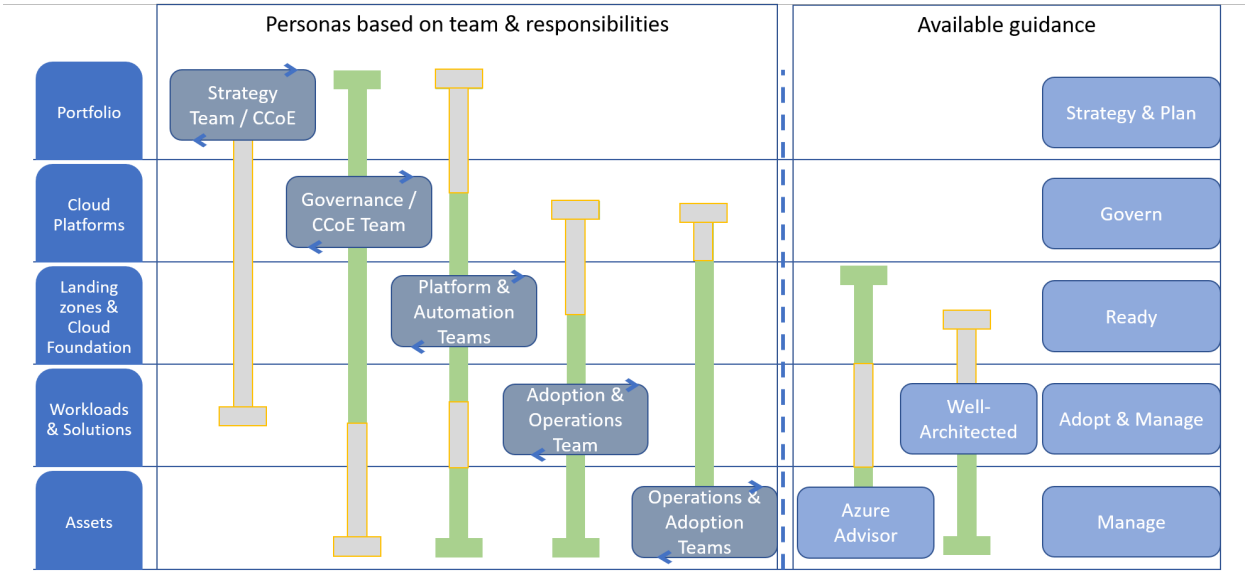
- **Landing zones:** Landing zones provide workloads with the necessary *foundational utilities* (or shared plumbing) that are provided from a *platform foundation* that's required to support one or more workloads. Landing zones are so critical in the cloud that the entire Ready methodology of the Cloud Adoption Framework focuses on landing zones. For a more detailed definition, see What is a landing zone?
- **Foundational utilities:** These shared IT services are required for workloads to operate within the technology and business portfolio.
- **Platform foundation:** This organizational construct centralizes foundational solutions and helps ensure that those controls are enforced for all landing zones.
- **Cloud platforms:** Depending on the overall strategy for supporting the full *portfolio*, customers might need multiple cloud platforms with distinct deployments of the platform foundation to govern multiple regions, hybrid solutions, or even multicloud solutions.
- **Portfolio:** Through a technology lens, the portfolio is a collection of workloads, assets, and supporting resources that span all cloud platforms. Through a business lens, the portfolio is the collection of projects, people, processes, and investments that support and manage the technology portfolio to drive business outcomes. Together, these two lenses capture the *portfolio*.

Hierarchy accountability and guidance

An accountable team manages each layer of the portfolio hierarchy. The following diagram shows the mapping between the accountable team and the guidance to support its business decisions, technical decisions, and

technical implementation.

> **NOTE**
> The teams mentioned in the following list might be virtual teams or individuals. For some variants of this hierarchy, some of the accountable teams can be collapsed as described later in the accountability variants.



- **Portfolio:** The cloud strategy team and the cloud center of excellence (CCoE) use the Strategy and Plan methodologies to guide decisions that affect the overall portfolio. The cloud strategy team is accountable for the enterprise level of the cloud portfolio hierarchy. The cloud strategy team should also be informed of decisions about the environment, landing zones, and high-priority workloads.
- **Cloud platforms:** The cloud governance team is accountable for the disciplines that ensure consistency across each environment in alignment with the Govern methodology. The cloud governance team is accountable for governance of all resources in all environments. The cloud governance team should be consulted on changes that might require an exception or change to governing policies. The cloud governance team should also be informed of progress with workload and asset adoption.
- **Landing zones and cloud foundation:** The cloud platform team is accountable for developing the landing zones and platform utilities that support adoption. The cloud automation team is accountable for automating the development of, and ongoing support for, those landing zones and platform utilities. Both teams use the Ready methodology to guide implementation. Both teams should be informed of progress with workload adoption and any changes to the enterprise or environment.
- **Workloads:** Adoption happens at the workload level. Cloud adoption teams use the Migrate and Innovate methodologies to establish scalable processes to accelerate adoption. After adoption is complete, the ownership of workloads is likely transferred to a cloud operations team that uses the Manage methodology to guide operations management. Both teams should be comfortable using the Microsoft Azure Well-Architected Framework to make detailed architectural decisions that affect the workloads they support. Both teams should be informed of changes to landing zones and environments. Both teams might occasionally contribute to landing zone features.
- **Assets:** Assets are typically the responsibility of the cloud operations team. That team uses the management baseline in the Manage methodology to guide operations management decisions. It should also use Azure Advisor and the Azure Well-Architected Framework to make detailed resource and architectural changes that are required to deliver on operations requirements.

Accountability variants

- **Single environment:** When an enterprise needs only one environment, a CCoE is typically not required.
- **Single landing zone:** If an environment has only a single landing zone, the governance and platform

capabilities can likely be combined into one team.

- **Single workload:** Some businesses need only one workload, or few workloads, in a single landing zone and a single environment. In those cases, there's little need for a separation of duties between governance, platform, and operations teams.

Common workload and accountability examples

The following examples illustrate the portfolio hierarchy.

COTS workloads

Traditionally, enterprises have favored commercial-off-the-shelf (COTS) software solutions to power business processes. These solutions are installed, configured, and then operated. There is little change to the solutions architecture after configuration.

In these scenarios, any cloud adoption of COTS solutions ends with a transition to a cloud operations team. The cloud operations team then becomes the technical owner for that software and assumes accountability for managing configuration, cost, patching cycles, and other operational needs.

These workloads include accounting packages, logistics software, or industry-specific solutions. In Microsoft terminology, the vendors of these packages are called independent software vendors (ISVs). Many ISVs offer a service to deploy and maintain an instance of their software package in your subscriptions. They might also offer a version of the software package that runs in their own cloud-hosted environment, providing a platform as a service (PaaS) alternative to the workload.

With the exception of PaaS offerings, cloud operations teams are responsible for ensuring basic operational compliance requirements for those workloads. A cloud operations team should work with the cloud governance team to align cost, performance, and other architecture pillars.

In development with active revisions

When a COTS solution or PaaS offering isn't aligned to the business need, or no solution exists, enterprises build custom-developed workloads. Typically, a small percentage of the IT portfolio uses this workload approach. But these workloads tend to drive a disproportionately high percentage of IT's contribution to business outcomes, especially outcomes related to new revenue streams. These workloads tend to map well to new innovation ideas.

Given various movements that are rooted in agile methodologies and DevOps practices, these workloads favor a business/DevOps alignment over traditional IT management. For these workloads, there might not be a handoff to the cloud operations team for several years. In those cases, the development team serves as the technical owner of the workload.

Due to extensive time and associated capital constraints, custom development options are often limited to high-value opportunities. Typical examples include application innovations, deep data analysis, or mission-critical business functions.

Break/fix or sunset development

After a custom-developed workload reaches peak maturity, the development team might be reassigned to other projects. In these cases, technical ownership typically transitions to a cloud operations team. When there's a need for small fixes, the operations team will enlist developers to resolve the error.

In some cases, the development team moves to a project that will eventually replace the current workload. Alternatively, the team might move on because the business opportunity supported by the workload is being phased out. These are examples of sunset scenarios, where the cloud operations team serves as the technical owner until the workload is no longer needed.

In both scenarios, the cloud operations team typically serves as the long-term technical owner and decision maker. That team will likely enlist application developers when operational changes require significant

architectural changes.

Mission-critical workloads

In every company, a few workloads are too important to the business for them to fail. With these mission-critical workloads, there are usually operations and development owners with various levels of responsibility. Those teams should align operational changes and architectural changes to minimize disruptions to the production solution.

These scenarios require a strong focus on separation of duties. To achieve separation of duties, the operations team will generally hold accountability for day-to-day operational changes in the production environment. When those operational changes require an architectural change, they'll be completed by the development or adoption team in a nonproduction environment, before the operations team applies the changes to production.

Examples of mission-critical workloads with a required separation of duties include workloads like SAP, Oracle, or other enterprise resource planning (ERP) solutions, which span multiple business units in the company.

Strategy portfolio alignment

It's important to understand the strategic objectives of the cloud adoption effort and align the portfolio to support that transformation. A few common types of strategic portfolio alignment help shape the structure of the portfolio hierarchy. The following sections provide examples of the portfolio alignment and impact on the portfolio hierarchy.

Innovation or development-led portfolio

Some companies, especially fast-growing established startups, have a higher-than-average percentage of custom development projects. In development-heavy portfolios, the environment, landing zone, and workloads are often compressed, so there might be specific environments (either production or nonproduction) for specific workloads. This results in a 1:1 ratio between environment, landing zone, and workload.

Because the environment hosts custom solutions, the DevOps pipeline and application-level reporting might replace the need for operations and governance functions. For those customers, a reduced focus on operations, governance, or other supporting roles is likely. A stronger emphasis on the responsibilities of the cloud adoption and cloud automation teams is also typical.

Portfolio alignment: The IT portfolio will likely focus on workloads and workload owners to drive critical architecture decisions. Those teams are likely to find more value in the Azure Well-Architected Framework guidance during adoption and operations activities.

Boundary definitions: The logical boundaries, even at an enterprise level, will likely focus on production and nonproduction environment segmentation. There might also be clear segmentation between products in the company's software portfolio. At times, there might also be segmentation between development and hosted customer instances.

Operations-led portfolio

Multinational enterprise organizations with more established IT operations teams typically have a stronger focus on governance and operations than development. In these organizations, a higher percentage of workloads typically align to the COTS or break/fix categories, maintained by technical owners within the cloud operations team.

Portfolio alignment: The IT portfolio will be workload aligned, but those workloads are then aligned to operating units or business units. There might also be organization around funding models, industry, or other business segmentation requirements.

Boundary definitions: Landing zones will likely group applications into application archetypes to keep similar operations in a similar segmentation. Environments will likely refer to physical constructs like datacenter, nation, cloud-provider region, or other operational organization standards.

Migration-led portfolio

Similar to operations-led portfolios, a portfolio that is largely built through migration will be based on specific business drivers that led to the migration of existing assets. Typically, the datacenter is the biggest factor in those drivers.

Portfolio alignment: The IT portfolio might be workload aligned, but it's more likely asset aligned. If transitions to IT operations have happened in the company's history, many active-use assets might not be easily mapped to a funded workload. In these cases, many assets might not have a defined workload or clear workload owner until late in the migration process.

Boundary definitions: Landing zones will likely group applications into boundaries that reflect on-premises segmentation. Though not a best practice, environments often match the on-premises datacenter name and landing zones that represent network segmentation practices. It's a better practice to adhere to segmentation that more closely aligns with an operations-led portfolio.

Governance-led portfolio

Alignment to governance teams should happen as early as possible. Through governance practices and cloud governance tooling, portfolios and environmental boundaries can best balance the needs of innovation, operations, and migration efforts.

Portfolio alignment: Governance-led portfolios tend to include data points that capture innovation and operations details, such as workload, operations owner, data classification, and operational criticality. These data points create a well-rounded view of the portfolio.

Boundary definitions: Boundaries in a governance-led portfolio tend to favor operations over innovation, while using a management-group-driven hierarchy that maps to criteria for business units and development environments. At each level of the hierarchy, a cloud governance boundary can have different degrees of policy enforcement to allow for development and creative flexibility. At the same time, production-grade requirements can be applied to production subscriptions to ensure separation of duties and consistent operations.

How do Azure products support the portfolio hierarchy?

5/18/2021 • 2 minutes to read • Edit Online

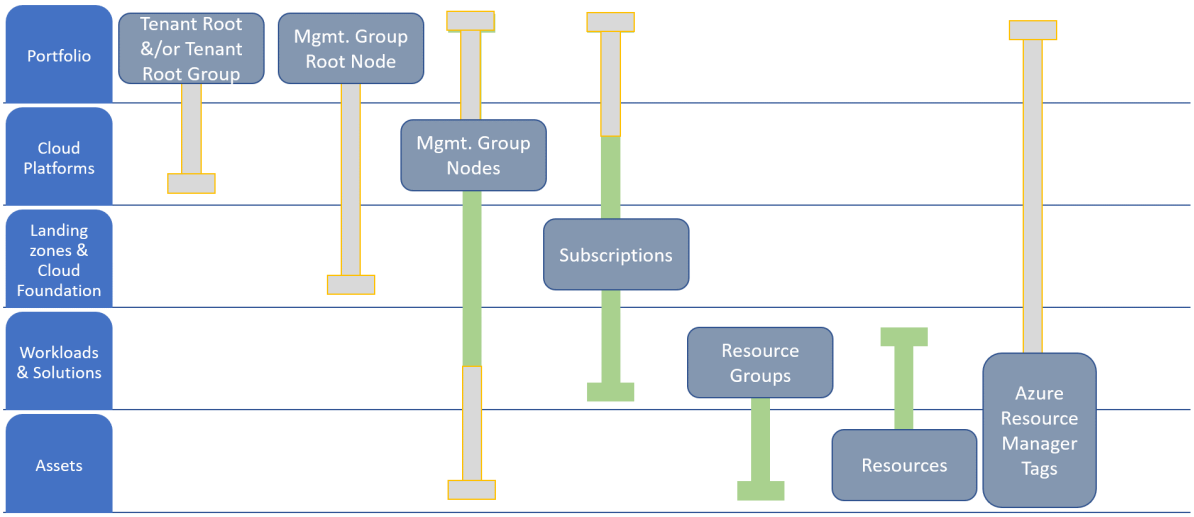
In Understanding and aligning the portfolio hierarchy, a set of definitions for the portfolio hierarchy and role mapping established a hierarchy of scope for most portfolio approaches. As described in that article, you might not need each of the outlined levels or *scopes*. Minimizing the number of layers reduces complexity, so these layers shouldn't all be viewed as a requirement.

This article shows how each level or scope of the hierarchy is supported in Azure through organizational tools, deployment and governance tools, and some solutions in the Microsoft Cloud Adoption Framework for Azure.

Organizing the hierarchy in Azure

Azure Resource Manager includes several organizational approaches that help organize assets at each level of the cloud hierarchy.

The slide bars in the following diagram demonstrate common variants in alignment. The gray parts of the slide bars are common but should be used only for specific business requirements. The points after the image describe a suggested best practice.



- **Portfolio:** The enterprise or business unit probably won't contain any technical assets but might affect cost decisions. The enterprise and business units are represented in the root nodes of the management group hierarchy.
- **Cloud platforms:** Each environment has its own node in the management group hierarchy.
- **Landing zones and cloud foundation:** Each landing zone is represented as a subscription. Likewise, platform foundations are contained in their own subscriptions. Some subscription designs might call for a subscription per cloud or per workload, which would change the organizing tool for each.
- **Workloads:** Each workload is represented as a resource group. Resource groups are often used to represent solutions, deployments, or other technical groupings of assets.
- **Assets:** Each asset is inherently represented as a resource in Azure.

Organizing with tags

Deviations from the best practice are common. You can record them by tagging all assets. Use a tag to represent each of the relevant layers of the hierarchy. For more information, see Recommended naming and tagging conventions.

Get started: Accelerate migration

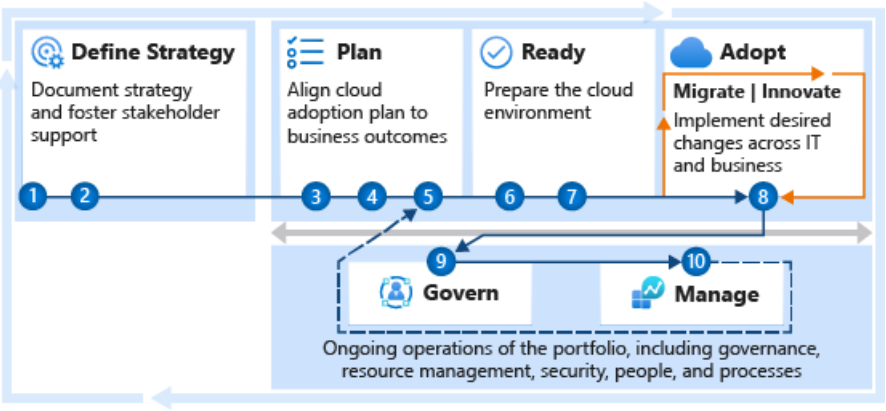
7/29/2021 • 9 minutes to read • Edit Online

Proper alignment of business and IT stakeholders helps to overcome migration roadblocks and accelerate migration efforts. This article provides recommended steps for:

- Stakeholder alignment
- Migration planning
- Deploying a landing zone
- Migrating your first 10 workloads

It also helps you implement proper governance and management processes.

Use this guide to streamline the processes and materials required for aligning an overall migration effort. The guide uses the methodologies of the Cloud Adoption Framework that are highlighted in this illustration.



If your migration scenario is atypical, you can get a personalized assessment of your organization's migration readiness by using the strategic migration and readiness tool (SMART) assessment. Use it to identify the guidance that best aligns to your current needs.

Get started

The technical effort and process required to migrate workloads is relatively straightforward. It's important to complete the migration process efficiently. Strategic migration readiness has an even bigger impact on the timelines and successful completion of the overall migration.

To accelerate adoption, you must take steps to support the cloud adoption team during migration. This guide outlines these iterative tasks to help customers start on the right path toward any cloud migration. To show the importance of the supporting steps, migration is listed as step 10 in this article. In practice, the cloud adoption team is likely to begin their first pilot migration in parallel with steps 4 or 5.

Step 1: Align stakeholders

To avoid common migration blockers, create a clear and concise business strategy for migration. Stakeholder alignment on motivations and expected business outcomes shapes decisions made by the cloud adoption team.

- Motivations: The first step to strategic alignment is to gain agreement on the motivations that drive the migration effort. Start by understanding and categorizing motivations and common themes from various stakeholders across business and IT.
- Business outcomes: After motivations are aligned, it's possible to capture the desired business outcomes. This

information provides clear metrics you can use to measure the overall transformation.

Deliverables:

- Use the strategy and plan template to record motivations and desired business outcomes.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Cloud strategy team	• Cloud adoption team • Cloud center of excellence or central IT team

Step 2: Align partner support

Partners, Microsoft Services, or various Microsoft programs are available to support you throughout the migration process.

- Understand partnership options to find the right level of partnership and support.

Deliverables:

- Establish terms and conditions or other contractual agreements before you engage supporting partners.
- Identify approved partners in the strategy and plan template.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Cloud strategy team	• Cloud adoption team • Cloud center of excellence or central IT team

Step 3: Gather data and analyze assets and workloads

Use discovery and assessment to improve technical alignment and create an action plan for executing your strategy. During this step, validate the business case using data about the current state environment. Then perform quantitative analysis and a deep qualitative assessment of the highest priority workloads.

- Inventory existing systems: Use a programmatic data-driven approach to understand the current state. Discover and gather data to enable all assessment activities.
- Incremental rationalization: Streamline assessment efforts to focus on a qualitative analysis of all assets, possibly even to support the business case. Then add a deep qualitative analysis for the first 10 workloads to be migrated.

Deliverables:

- Raw data on existing inventory.
- Quantitative analysis on existing inventory to refine the business justification.
- Qualitative analysis of the first 10 workloads.
- Business justification documented in the strategy and plan template.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Cloud adoption team	• Cloud strategy team

Step 4: Make a business case

Making the business case for migration is likely to be an iterative conversation among stakeholders. In this first pass at building the business case, evaluate the initial high-level return from a potential cloud migration. The goal of this step is to ensure that all stakeholders align around one simple question: based on the available data, is the overall adoption of the cloud a wise business decision?

- Building a cloud migration business case is a good starting point for developing a migration business case. Clarity on formulas and tools can aid in business justification.

Deliverables:

- Use the strategy and plan template to record business justification.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Cloud strategy team	• Cloud adoption team

Step 5: Create a migration plan

A cloud adoption plan provides an accelerated approach to developing a project backlog. The backlog can then be modified to reflect discovery results, rationalization, needed skills, and partner contracting.

- Cloud adoption plan: Define your cloud adoption plan using the basic template.
- Workload alignment: Define workloads in the backlog.
- Effort alignment: Align assets and workloads in the backlog to clearly define effort for prioritized workloads.
- People and time alignment: Establish iteration, velocity (people's time), and releases for the migrated workloads.

Deliverables:

- Deploy the backlog template.
- Update the template to reflect the first 10 workloads to be migrated.
- Update people and velocity to estimate release timing.
- Timeline risks:
  - Lack of familiarity with Azure DevOps can slow the deployment process.
  - Complexity and data available for each workload can also affect timelines.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Cloud adoption team	• Cloud strategy team

Step 6: Build a skills readiness plan

Existing employees can play a hands-on role in the migration effort, but additional skills might be required. In this step, find ways to develop those skills or use partners to add to those skills.

- Build a skills-readiness plan. Quickly evaluate your existing skills to identify what other skills the team should develop.

Deliverables:

- Add a skills-readiness plan to the strategy and plan template.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Cloud adoption team	• Cloud strategy team

Step 7: Deploy and align a landing zone

All migrated assets are deployed to a landing zone. The landing zone start simple to support smaller workloads, then scales to address more complex workloads over time.

- Choose a landing zone: Find the right approach to deploying a landing zone based on your adoption pattern. Then deploy that standardized code base.
- Expand your landing zone: Whatever your starting point, identify gaps in the deployed landing zone and add required components for resource organization, security, governance, compliance, and operations.

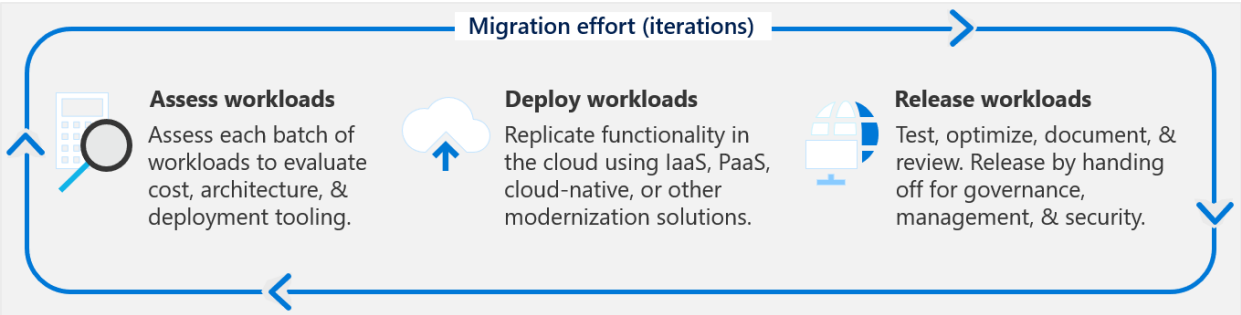
Deliverables:

- Deploy your first landing zone for deploying initial low-risk migrations.
- Develop a refactoring plan with the cloud center of excellence or the central IT team.
- Timeline risks:
  - Governance, operations, and security requirements for the first 10 workloads can slow this process.
  - Refactoring the first landing zone and subsequent landing zones takes longer, but it should happen in parallel with migration efforts.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Cloud platform team	• Cloud adoption team • Cloud center of excellence or central IT team

Step 8: Migrate your first 10 workloads

The technical effort required to migrate your first 10 workloads is relatively straightforward. It's also an iterative process that you repeat as you migrate more assets. In this process, you assess your workloads, deploy your workloads, and then release them to your production environment.



Cloud migration tools enable migrating all virtual machines in a datacenter in one pass or iteration. It's more common to migrate a smaller number of workloads during each iteration. Breaking up the migration into smaller increments requires more planning, but it reduces technical risks and the impact of organizational change management.

With each iteration, the cloud adoption team gets better at migrating workloads. These steps help the technical

team mature their capabilities:

1. Migrate your first workload in a pure infrastructure as a service (IaaS) approach by using the tools outlined in the Azure migration guide.
2. Expand tooling options to use migration and modernization by using the migration examples.
3. Develop your technical strategy by using broader approaches outlined in Azure cloud migration best practices.
4. Improve consistency, reliability, and performance through an efficient migration-factory approach as outlined in Migration process improvements.

Deliverables:

Continuous improvement of the adoption team's ability to migrate workloads.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Cloud adoption team	• Cloud strategy team • Cloud center of excellence or central IT team

Step 9: Hand off production workloads to cloud governance

Governance is a key factor to the long-term success of any migration effort. Speed to migration and business impact is important. But speed without governance can be dangerous. Your organization needs to make decisions about governance that align to your adoption patterns and your governance and compliance needs.

- Governance approach: This methodology outlines a process for thinking about your corporate policy and processes. After determining your approach, you can build the disciplines required to enable governance across your enterprise cloud adoption efforts.
- Initial governance foundation: Understand the disciplines needed to create a governance minimum viable product (MVP) that serves as the foundation for all adoption.
- Governance benchmark: Identify gaps in your organization's current state of governance. Get a personalized benchmark report and curated guidance on how to get started.

Deliverables:

- Deploy an initial governance foundation.
- Complete a governance benchmark to plan for future improvements.
- Timeline risk:
  - Improvement of policies and governance implementation can add one to four weeks per discipline.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Cloud governance team	• Cloud strategy team • Cloud center of excellence or central IT team

Step 10: Hand off production workloads to cloud operations

Operations management is another requirement to reach migration success. Migrating individual workloads to the cloud without an understanding of ongoing enterprise operations is a risky decision. In parallel with migration, you should start planning for longer-term operations.

- Define business commitments
- Establish a management baseline
- Expand the management baseline
- Advanced operations and design principles

Deliverables:

- Deploy a management baseline.
- Complete the operations management workbook.
- Identify any workloads that require an Microsoft Azure Well-Architected Review assessment.
- Timeline risks:
  - Review the workbook: estimate one hour per application owner.
  - Complete the Microsoft Azure Well-Architected Review assessment: estimate one hour per application.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Cloud operations team	• Cloud strategy team • Cloud center of excellence or central IT team

Value statement

These steps help teams accelerate their migration efforts through better change management and stakeholder alignment. These steps also remove common blockers and realize business value more quickly.

Next steps

The Cloud Adoption Framework is a lifecycle solution that helps you begin a migration journey. It also helps mature the teams that support migration efforts. The following teams can use these next steps to continue to mature their capabilities. These parallel processes aren't linear and shouldn't considered blockers. Instead, each is a parallel value stream to help improve your organization's overall cloud readiness.

TEAM	NEXT ITERATION
Cloud adoption team	Use the migration model to learn about moving toward a migration factory that provides efficient ongoing migration capabilities.
Cloud strategy team	Iteratively improve the Strategy methodology and the Plan methodology along with the adoption plan. Review these overviews and continue iterating on your business and technical strategies.
Cloud platform team	Revisit the Ready methodology to continue to advance the overall cloud platform that supports migration or other adoption efforts.
Cloud governance team	Use the Govern methodology to continue to improve governance processes, policies, and disciplines.
Cloud operations team	Build on the Manage methodology to provide richer operations in Azure.

If your migration scenario is atypical, you can get a personalized assessment of your organization's migration

readiness by using the strategic migration and readiness tool (SMART) assessment. The answers you provide help identify which guidance aligns best with your current needs.

Get started: Accelerate new product and service innovation in the cloud

7/21/2021 • 10 minutes to read • Edit Online

Creating new products and services in the cloud requires a different approach than migration requires. The Innovate methodology of the Cloud Adoption Framework establishes an approach that guides the development of new products and services.

This guide uses the sections of the Cloud Adoption Framework that are highlighted in the following illustration. Innovation is less predictable than a standard migration, but it still fits within the context of the broader cloud adoption plan. This guide can help your enterprise provide the support needed to innovate and provide a structure for creating a balanced portfolio throughout cloud adoption.

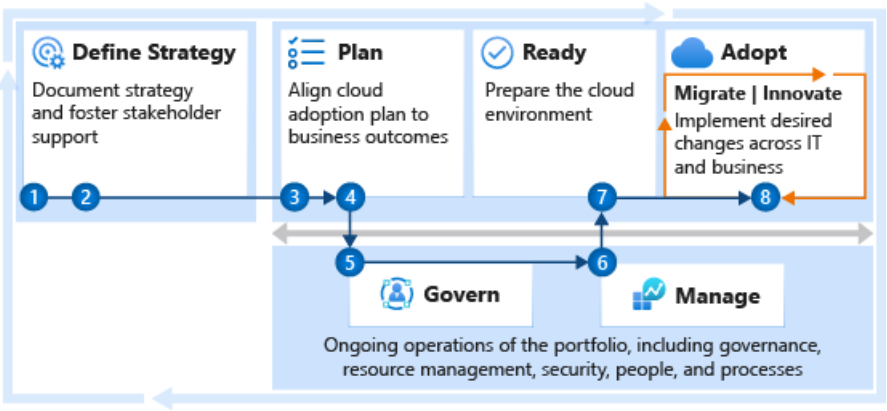


Figure 1: The Cloud Adoption Framework methodologies, including the Innovate methodology.

Step 1: Document the business strategy

To avoid common blockers, create a clear and concise business strategy for innovation. Stakeholder alignment on motivations and expected business outcomes shapes decisions that the cloud adoption team makes.

Deliverables:

- Use the strategy and plan template to record motivations and desired business outcomes.

Guidance to support deliverable completion:

- Motivations: The first step to strategic alignment is to gain agreement on the motivations that drive the innovation effort. Start by understanding and categorizing motivations and common themes from stakeholders across business and IT.
- Business outcomes: After motivations are aligned, it's possible to capture the desired business outcomes. This information provides clear metrics that you can use to measure the overall transformation.
- Balancing the portfolio: Innovation isn't the right adoption path for every workload. This approach to adoption is more relevant to new custom-built applications or workloads that *require* rearchitecture or full rebuilds. When motivations heavily favor innovation for all workloads, it's important to evaluate the portfolio to ensure that those investments can produce the desired return on investment. Modernization of specific resources and small-scale rebuilding efforts can be innovative but might be better served by following Get started: Accelerate migration.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Cloud strategy team	• Cloud adoption team • Cloud center of excellence or central IT team

Step 2: Evaluate the business justification

In this first pass at building the business case, evaluate the initial high-level return from a potential cloud adoption effort. The goal of this step is to align all stakeholders around one simple question: based on the available data, is the overall adoption of the cloud a wise business decision? Building on that question, the team can better align on how this innovation project helps meet the users' projected needs within the goal of adopting the cloud.

Deliverables:

- Use the strategy and plan template to record the business justification.

Guidance to support deliverable completion:

- Business justification: Before you evaluate each opportunity to innovate in the cloud, complete a high-level business justification to establish stakeholder alignment for the overall adoption plan.
- Business value consensus: Quantifying the value of an innovation can be difficult early in the process. The exercise in this article can aid in evaluating alignment on the business value of a specific innovation effort.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Cloud strategy team	• Cloud adoption team

Step 3: Gather data and analyze assets and workloads

In most enterprises, innovation can be accelerated through the use of existing assets like applications, virtual machines (VMs), and data. When you plan for innovation, it's important to understand how and when those assets are migrated to the cloud.

Deliverables:

- Get raw data on existing inventory like applications, VMs, and data.
- If the proposed innovation has dependencies on existing inventory, complete the following deliverables:
  - Quantitative analysis on any supporting inventory required to support the planned innovation.
  - Qualitative analysis of any supporting workloads required to deliver the innovation.
- Calculate the cost of new inventory required to support the innovation effort.
- Update the business justification in the strategy and plan template with refined calculations.

Guidance to support deliverable completion:

Discovery and assessment provide a deeper level of technical alignment. You can then create an action plan for migrating any dependent workloads that the planned innovation requires. This scenario is common when companies have existing data sources, centralized applications, or service layers that are necessary for delivering innovation within the context of the rest of the enterprise.

When there are dependent systems, the following articles can guide the discovery and assessment:

- Inventory existing systems: Understanding the current state from a programmatic, data-driven approach is

the first step. Discover and gather data to enable all assessment activities.

- Incremental rationalization: Streamline assessment efforts to focus on a qualitative analysis of all assets, possibly even to support the business case. Then add a deep qualitative analysis for the first 10 workloads.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Cloud adoption team	• Cloud strategy team

Step 4: Plan for migration of dependent assets

When new innovation depends on existing workloads or assets, a cloud adoption plan provides an accelerated approach to developing a project backlog. The backlog can then be modified to reflect discovery results, rationalization, needed skills, and partner contracting.

Deliverables:

- Deploy the backlog template.
- Update the template to reflect the first 10 workloads to be migrated.
- Update people and velocity (people's time) to estimate release timing.
- Timeline risks:
  - Lack of familiarity with Azure DevOps can slow the deployment process.
  - Complexity and data available for each workload can also affect timelines.

Guidance to support deliverable completion:

- Cloud adoption plan: Define your plan using the basic template.
- Workload alignment: Define workloads in the backlog.
- Effort alignment: Align assets and workloads in the backlog to clearly define the effort for prioritized workloads.
- People and time alignment: Establish iteration, velocity, and releases for the workloads.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Cloud adoption team	• Cloud strategy team

Step 5: Align governance requirements to your adoption plan

Discussing planned innovations with the governance team helps you avoid many blockers before they arise. Sometimes, innovative new solutions might require practices that are discouraged in sound governance practices. Some of those required features might even be blocked through automated tooling for governance enforcement.

Deliverables:

- Create transparency and understanding between innovation needs and governance constraints.
- When necessary, update policies and processes to reflect any changes or exceptions to existing governance constraints.

Guidance to support deliverable completion:

These links help the adoption team understand the approach of the cloud governance team:

- Governance approach: This methodology outlines a process for thinking about corporate policy and processes. Then you can build the disciplines required to deliver on governance across your cloud enterprise efforts.
- Definition of corporate policy: Identify and mitigate business risks.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Cloud governance team • Cloud adoption team	• Cloud strategy team • Cloud center of excellence or central IT team

Step 6: Define operational needs and business commitments

Define the plan for long-term operational responsibilities for the planned innovation. Will the established management baseline meet your operational needs? If not, evaluate options for funding operations that are specific to the technology that supports this innovation.

Deliverables:

- Complete the Microsoft Azure Architecture Review to assess various architecture and operation decisions.
- Adjust the operations management workbook to reflect any required advanced operations.

Guidance to support deliverable completion:

- Expand the management baseline: This section of the Cloud Adoption Framework guides you through various transitions into operational management in the cloud.
- Get specific with advanced operations: Discover ways to go beyond your management baseline.
- If advanced operations are required to support your operations needs, evaluate the business commitments to determine operational responsibilities for both teams.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Cloud operations team • Cloud adoption team	• Cloud strategy team • Cloud center of excellence or central IT team

Step 7: Deploy an aligned landing zone

All assets hosted in the cloud live within a landing zone. That landing zone might have explicit governance, security, and operational requirements. Or, it might be a new subscription without support from other teams. In either scenario, it's important to start with a landing zone that aligns to governance and operational requirements from the beginning.

Starting with an approved landing zone helps your team to discover policy violations early during development versus when the solution is released to production. Early discovery helps your team to remove blockers and gives adoption and governance teams enough time to make changes.

Deliverables:

- Deploy a first landing zone for initial, low-risk experimentation during early innovation.
- Develop a plan to refactor with the cloud center of excellence or the central IT team to ensure governance, security, and operational alignment.
- Timeline risks:

- Governance, operations, and security requirements for the first 10 workloads can slow this process. Refactoring the first landing zone and later landing zones takes longer, but it should happen in parallel with migration efforts.

Guidance to support deliverable completion:

- Choose a landing zone: Use this section to find the right approach to deploying a landing zone based on your adoption pattern. Then deploy that standardized code base.
- Expand your landing zone: Regardless of the starting point, identify gaps in the deployed landing zone to add required components for resource organization, security, governance, compliance, and operations.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Cloud platform team • Cloud adoption team	• Cloud adoption team • Cloud center of excellence or central IT team

Step 8: Innovate in the cloud

The Innovate methodology provides guidance on the tools and product management approaches most commonly used to innovate in the cloud. These steps help you get started with this approach.

Deliverables:

- Technology-based solutions that enrich your customers' lives and drive value for the business.
- Processes and tools to iterate on those solutions faster and add more value by using the cloud:
  - Iterative development approaches.
  - Custom-built applications.
  - Technology-based experiences.
  - Integration of physical products and technology by using IoT.
  - Ambient intelligence: integration of nonintrusive technology into an environment.
  - Azure Cognitive Services: big data, AI, machine learning, and predictive solutions.

Guidance to support deliverable completion:

- Create hypothesis with business value consensus: Before you decide on technical solutions, identify how new innovations can drive business value and come up with a hypothesis about customer needs.
- Build your first MVP: Once a hypothesis has enough value potential to be built into your application, the build process starts. Development sprints should be as quick as possible, to allow quick verification or rejection of the hypothesis, or to fine tune the way in which the required functionality is to be integrated in the application.
- Measure and learn: You want to verify the accuracy of your hypothesis as soon as possible. A minimum viable product (MVP) is a preliminary version of the new functionality, that offers just enough functionality to gather feedback that confirms if you are moving in the right direction.
- Expand digital innovation: To refine your hypothesis using the innovation disciplines or the digital inventions which include democratize data, engage via applications, empower adoption, interact with devices, and predict and influence. These inventions are a core part of the Innovate methodology.

Step 9: Assess the innovation maturity of your organization

To support the development of your innovation strategy, the AI readiness assessment tool is a free resource that helps organizations assess their ability to create and own AI-based systems. There are four levels of maturity: foundational, approaching, aspirational, and mature. Each level includes a specific set of characteristics to help

determine your organization's ability to adopt specific types of AI solutions, mitigate associated risks, and implement strategies.

The assessment takes 5 to 10 minutes and measures your organization's capability across four categories: strategy, culture, organizational characteristics, and capabilities. Measuring these categories allows the AI readiness assessment tool to compute your organization's score and provide an estimate of the AI innovation maturity on a curve.

Deliverables:

- Use the Gartner AI Maturity Model to assess your organization's AI maturity to create AI-based systems.

Guidance to support deliverable completion:

- When the assessment is complete, the tool's output will provide a score that estimates the status of AI innovation maturity.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Cloud adoption team	• Cloud center of excellence • Cloud center of excellence or central IT team

Value statement

The steps outlined in this guide can help you and your teams create innovative solutions in the cloud that create business value, are governed appropriately, and are well architected.

Next steps

The Cloud Adoption Framework is a lifecycle solution. It can help you begin an innovation journey. It can help your organization to start an innovation journey and to advance the maturity of the teams that support innovation efforts.

The following teams can use these next steps to continue to advance the maturity of their efforts. These parallel processes aren't linear and shouldn't be viewed as blockers. Instead, each is a parallel value stream to help mature your company's overall cloud readiness.

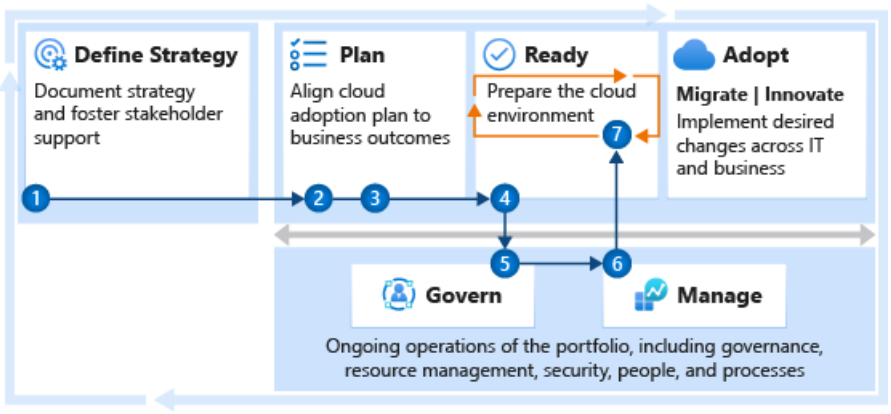
TEAM	NEXT ITERATION
Cloud adoption team	Process improvements provide insight about approaches to deliver on innovations that affect customers and drive recurring adoption.
Cloud strategy team	The Strategy methodology and the Plan methodology are iterative processes that evolve with the adoption plan. Return to these overview pages and continue to iterate on your business and technical strategies.
Cloud platform team	Revisit the Ready methodology to continue to advance the overall cloud platform that supports migration or other adoption efforts.
Cloud governance team	Use the Govern methodology to continue to improve governance processes, policies, and disciplines.

TEAM	NEXT ITERATION
Cloud operations team	Build on the Manage methodology to provide richer operations in Azure.

Get started: Environment design and configuration

7/21/2021 • 7 minutes to read • Edit Online

Environment design and configuration are the most common blockers to adoption efforts that are focused on migration or innovation. Quickly implementing a design that supports your long-term adoption plan can be difficult. This article establishes an approach and series of steps that help to overcome common blockers and accelerate your adoption efforts.



The technical effort required to create an effective environmental design and configuration can be complex. You can manage the scope to improve the odds of success for the cloud platform team. The greatest challenge is alignment among multiple stakeholders. Some of these stakeholders have the authority to stop or slow the adoption efforts. These steps outline ways to quickly meet short-term objectives and establish long-term success.

Step 1: Document the business strategy

To avoid common migration blockers, make sure that you have a clear and concise business strategy. Stakeholder alignment on motivations, expected business outcomes, and the business justification is important throughout adoption and environment configuration.

A clear and concise business strategy helps the cloud platform team understand what's important and what should be prioritized when they're making environmental configuration decisions. In particular, it helps the teams make decisions when they're forced to choose between speed of innovation or adherence to controls.

Deliverables:

- Use the strategy and plan template to record motivations, desired business outcomes, and high-level business justification.

Guidance to support deliverable completion:

- Understand business motivations: The first step to strategic alignment is to agree on the motivations that drive the migration effort. Start by understanding and categorizing motivations and common themes from various stakeholders across business and IT.
- Document business outcomes: After motivations are aligned, you can capture the desired business outcomes. This information provides clear metrics you can use to measure the overall transformation.
- Build a cloud migration business case: Start developing a business case for migration, including clear guidance on the formulas and tools that help your business justification.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS	INFORMED TEAMS
• Cloud strategy team	• Cloud adoption team • Cloud center of excellence or central IT team	• Cloud platform team

Step 2: Assess the digital estate

Discovery and assessment provide a deeper level of technical alignment, which helps you create an action plan you can use to deliver on the strategy. During this step, you validate the business case by using data about the current state of the environment. Then you perform quantitative analysis of that data and a deep qualitative assessment of the highest priority workloads.

The output of the digital estate assessment provides the cloud platform team with a clear view of the end-state environment and the requirements that are needed to support the adoption plan.

Deliverables:

- Raw data on the existing inventory.
- Quantitative analysis of the existing inventory to refine the business justification.
- Qualitative analysis of the first 10 workloads.
- Updated business justification in the strategy and plan template.

Guidance to support deliverable completion:

- Inventory existing systems: Understanding the current state from a programmatic, data-driven approach is the first step. Find and gather data to enable all assessment activities.
- Incremental rationalization: Streamline assessment efforts to focus on a qualitative analysis of all assets, possibly even to support the business case. Then add a deep qualitative analysis for the first 10 workloads to be migrated.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS	INFORMED TEAMS
• Cloud adoption team	• Cloud strategy team	• Cloud platform team

Step 3: Create a cloud adoption plan

Your cloud adoption plan provides an accelerated approach to developing a project backlog. The backlog can then be modified to reflect assessment results, rationalization, needed skills, and partner contracting.

A review of the short-term cloud adoption plan and backlog helps the cloud platform team understand the needs of the environment for the next few months. This background helps them to tighten the "definition of done" for the first few landing zones.

Deliverables:

- Deploy the backlog template.
- Update the template to reflect the first 10 workloads to be migrated.
- Update people and velocity (people's time) to estimate release timing.
- Timeline risks:
  - Lack of familiarity with Azure DevOps can slow the deployment process.

- Complexity and data available for each workload can also affect timelines.

Guidance to support deliverable completion:

- Cloud adoption plan: Define your plan using the basic template.
- Workload alignment: Define workloads in the backlog.
- Effort alignment: Align assets and workloads in the backlog to clearly define efforts for prioritized workloads.
- People and time alignment: Establish iteration, velocity, and releases for the migrated workloads.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS	INFORMED TEAMS
• Cloud adoption team	• Cloud strategy team • Cloud platform team	• Cloud platform team

Step 4: Deploy the first landing zone

Initially, the cloud adoption team needs a landing zone that can support the requirements of the first wave of workloads. Over time, the landing zone scales to address more complex workloads. For now, start with a landing zone that enables early learning for the cloud platform team and the cloud adoption team.

Deliverables:

- Deploy a first landing zone for initial low-risk migrations.
- Develop a plan to refactor with the cloud center of excellence or the central IT team.
- Timeline risks:
  - Governance, operations, and security requirements for the first 10 workloads can slow this process. Actual refactoring of the first landing zone and subsequent landing zones takes longer, but it should happen in parallel with migration efforts.

Guidance to support deliverable completion:

- Choose a landing zone: Use this section to find the right approach to deploying a landing zone based on your short-term adoption plan. Then deploy that standardized code base.
- Expand your landing zone: Don't attempt to meet long-term governance, security, or operation constraints yet, unless they're required to support the short-term adoption plan.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Cloud platform team	• Cloud adoption team • Cloud center of excellence or central IT team

Step 5: Deploy an initial governance foundation

Governance is a key factor to the long-term success of any migration effort. Speed to migration and business impact is important. But speed without governance can be dangerous. Your organization needs to make decisions about governance that align to your adoption patterns and your governance and compliance needs.

As those decisions are made, they feed back into the parallel efforts of the cloud platform team.

Deliverables:

- Deploy an initial governance foundation.
- Complete a governance benchmark to plan for future improvements.
- Timeline risks:
  - Improvement of policies and governance implementation can add one to four weeks per discipline.

Guidance to support deliverable completion:

- Governance approach: This methodology outlines a process for thinking about corporate policy and processes. Then build the disciplines required to deliver on governance across your cloud enterprise adoption efforts.
- Governance benchmark tool: Find gaps in your current state so that you can plan for the future.
- Initial governance foundation: Understand the governance disciplines that are required to create a governance minimum viable product (MVP) to serve as the foundation for all adoption.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS	CONSULTED TEAMS
• Cloud governance team	• Cloud strategy team • Cloud center of excellence or central IT team	• Cloud platform team

Step 6: Implement an operations baseline

Migrating to the cloud without understanding ongoing operations is risky. In parallel with migration, start planning for longer-term operations management. Feed those plans back into the parallel efforts of the cloud platform team.

Deliverables:

- Deploy a management baseline.
- Complete the operations management workbook.
- Identify any workloads that require an Microsoft Azure Well-Architected Review assessment.
- Timeline risks:
  - Review the workbook: estimate one hour per application owner.
  - Complete the Microsoft Azure Well-Architected Review assessment: estimate one hour per application.

Guidance to support deliverable completion:

- Establish a management baseline
- Define business commitments
- Expand the management baseline
- Get specific with advanced operations

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS	CONSULTED TEAMS
• Cloud operations team	• Cloud strategy team • Cloud center of excellence or central IT team	• Cloud platform team

Step 7: Expand the landing zone

As the cloud adoption team begins their first few migrations, the cloud platform team can begin building toward the end-state environment configuration with the support of the cloud governance and cloud operations teams. Depending on the pace of the cloud adoption plan, this process might need to happen in iterative releases. Functionality might be added ahead of the requirements of the adoption plan.

Deliverables:

- Adopt a test-driven development approach to refactoring landing zones.
- Improve landing zone governance.
- Expand landing zone operations.
- Implement landing zone security.

Guidance to support deliverable completion:

- Refactor landing zones
- Test-driven development of landing zones
- Expand landing zone governance
- Expand landing zone operations
- Expand landing zone security

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Cloud platform team	• Cloud adoption team • Cloud center of excellence or central IT team

Value statement

The steps outlined in this guide can help you and your teams accelerate their path to an enterprise-ready cloud environment that's properly configured.

Next steps

Consider these next steps in a future iteration to build on your initial efforts:

- Environmental technical readiness learning paths
- Migration environment planning checklist

Enable customer success with a sound operating model and organizational alignment

7/21/2021 • 2 minutes to read • Edit Online

Customer success in cloud adoption efforts often has little to do with technical skills or adoption-related projects. Your operating model creates opportunities to enable adoption or roadblocks that might slow down cloud adoption.

Alignment

As you drive innovation, alignment between business and technical teams is paramount to the success of your solution.

- For business stakeholders, we've created the Microsoft AI Business School to support business strategy development and provide example best practices.
- For technical stakeholders, the Microsoft AI learning paths are available to help you build new AI skills.

Blockers

When adoption of the cloud is slowed or stalled, it might be wise to evaluate your operating model to enable continued success. When success is inconsistent from workload to workload or project to project, the operating model might be misaligned. If more than one project is stalled by blocking policies, outdated processes, or misalignment of people, the operating model is likely blocking success.

Opportunities

Beyond the common blockers, a few key opportunities can be scaled across the portfolio through incremental improvements to your operating model. In particular, customers commonly want to scale operational excellence, cost optimization, security, reliability, performance, or people management. Scaling these conversations at the portfolio level can help bring best practices for specific workload-focused teams to all other projects and workloads.

Getting started guides to enable teams through an operating model

The following guides will help you get started with operating model alignment and improve over time.

GUIDE	DESCRIPTION
How do we deliver operational excellence during cloud transformation?	The steps in this guide will help the strategy team lead organizational change management to consistently ensure operational excellence.
How do we manage enterprise costs?	Start optimizing enterprise costs and manage cost across the environment.
How do we consistently secure the enterprise cloud environment?	This getting started guide can help ensure that the proper security requirements have been applied across the enterprise to minimize risk of breach and accelerate recovery when a breach occurs.

GUIDE	DESCRIPTION
How do we apply the right controls to improve reliability?	This getting started guide helps minimize disruptions related to inconsistencies in configuration, resource organization, security baselines, or resource protection policies.
How do we ensure performance across the enterprise?	This getting started guide can help you establish processes for maintaining performance across the enterprise.
How do we align our organization?	This getting started guide can help you establish an appropriately staffed organizational structure.

Shared architecture principles

The core principles of a well-managed operating model are based on a set of common architecture principles. The getting started guidance in this article series will help supporting teams as they scale these principles across the cloud platform and throughout the portfolio of workloads.



Figure 1: The principles of shared architecture.

These principles are shared across Azure Advisor, the Microsoft Azure Well-Architected Framework, and solutions in the Azure Architecture Center:

- Azure Advisor evaluates the principles for individual assets across solutions, workloads, and the full portfolio.
- Azure Architecture Center applies these principles to develop and manage specific technical solutions.

- Microsoft Azure Well-Architected Framework helps balance these principles across a workload, to guide architecture decisions.
- Cloud Adoption Framework ensures that the principles scale across the portfolio to enable adoption teams through a well-managed environment.

Get started: Deliver operational excellence during digital transformation

7/21/2021 • 5 minutes to read • Edit Online

How do you ensure operational excellence during digital transformation? Operational excellence is a business function that directly affects IT decisions. To achieve operational excellence, you must focus on customer and stakeholder value by keeping an eye on revenue, risk, and cost impacts.

This organizational change management approach requires:

- A defined strategy.
- Clear business outcomes.
- Change management planning.

From a cloud perspective, you can manage the impact of risk and cost by making post-adoption changes and continuously refining operational processes. Areas to monitor include systems automation, IT operations management practices, and Resource Consistency discipline throughout the cloud adoption lifecycle.

The steps in this article can help the strategy team lead the organizational change management that's required to consistently ensure operational excellence.

Operational excellence across the enterprise and portfolio starts with peer processes of strategy and planning to align and report on organizational change management expectations. The following steps help technical teams deliver the disciplines required to achieve operational excellence.



Step 1: Define a strategy to guide digital transformation and operational excellence expectations

A clear business strategy is the foundation for any digital transformation and operational excellence effort. IT can reduce costs and streamline IT processes. Without a clear strategy, it's difficult to understand how those changes might affect the business outcomes identified in the broader transformation effort.

Deliverables:

- Record motivations, outcomes, and business justification in the strategy and plan template.
- Ensure learning metrics are well understood and included in the business outcomes section. Those metrics guide operational excellence activities and reporting within IT.

Guidance to support deliverable completion:

- Understand motivations: Critical business events and some migration motivations tend to be cost sensitive. These areas can increase the importance of cost control for all later efforts. Other forward-looking motivations related to innovation or growth through migration might be focused more on top-line revenue. Understanding motivations helps you prioritize your cost management.
- Business outcomes: Some fiscal outcomes tend to be extremely cost sensitive. When the desired outcomes map to fiscal metrics, you should invest early in the Cost Management governance discipline.
- Business justification: The business justification serves as a high-level view of the overall financial plan for cloud adoption. It can be a good source for initial budgeting efforts.
- Learning metrics: To maintain alignment between the overarching business strategy and the more tactical change-management plans, establish learning metrics. These metrics should be designed to show iterative and incremental progress toward the plan.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Cloud strategy team	• Cloud adoption team • Cloud governance team • Cloud operations team • Cloud center of excellence or central IT team

Step 2: Develop an organizational change management plan to span cloud adoption

Organizational change management is an iterative approach to subtly realign people, processes, and technology to support a holistic business strategy. In the case of operational excellence for digital transformation, this approach often centers on an IT-centric cloud adoption plan.

Deliverables:

- Update the strategy and plan template to reflect change that's needed to achieve the desired strategy. The changes recorded can include:
  - An assessment of the existing digital estate.
  - A cloud adoption plan that reflects the required changes and the work involved.
  - The organizational changes that are required to deliver on the plan.
  - A plan for addressing the skills that are needed to enable the existing team to successfully complete the required work.

Guidance to support deliverable completion:

- Gather inventory: Establish a source of data for analysis of the digital estate prior to adoption.
- Best practice: Azure Migrate: Use Azure Migrate to gather inventory.
- Incremental rationalization: During incremental rationalization, a quantitative analysis identifies cloud candidates for budgeting purposes.
- Align cost models and forecast models: Use Azure Cost Management + Billing to align cost and forecast models by creating budgets.
- Build your cloud adoption plan: Build a plan with actionable workload, assets, and timeline details.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Cloud strategy team	• Cloud adoption team • Cloud governance team • Cloud operations team • Cloud center of excellence or central IT team

Step 3: Manage change across cloud adoption efforts

Realization of business outcomes is the result of continuous delivery of adoption waves. Those waves could include migration and innovation cycles. In either case, delivery on operational excellence starts with regular cycles of change management.

Each wave (or release, in agile terms) delivers a set of workloads to the cloud. As each wave of adoption is completed, the cloud strategy team reports on progress toward learning metrics, business outcomes, and the overall strategy. Likewise, as each wave of adoption is completed, the adoption teams need backlog updates that reflect the prioritized workloads in the plan. These updates are based on any changes to business plans and customer needs.

Deliverables:

- Continuous testing and improvements to the strategy and change management plan based on changing market conditions and completion of the most recent wave of technical change.

Guidance to support deliverable completion:

- Release planning: Approaches to change management through release planning.
- Incremental rationalization: Iterative approach to change management. The focus is on managing the release backlog to support manageable waves of change.
- Power of 10 approach: Limits the change management plan. The focus is on detailed analysis and planning of a continuous base of 10 workloads to balance incremental change and iterative adoption efforts.
- Align iteration paths: Update and add details at each release to ensure current iteration paths.
- Assess workloads: The efforts of the cloud adoption team to evaluate and act on the most recent set of migration priorities.
- Business value consensus: The cloud adoption team's efforts to ensure business value alignment at each release of new innovation.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Cloud strategy team	• Cloud adoption team

Value statement

The previous steps outline a business-led approach to establish operational excellence requirements throughout digital transformation. This approach provides a consistent foundation that carries through other operating model functions.



Next steps to delivering operational excellence across the portfolio

Operational excellence requires a disciplined approach to reliability, performance, security, and cost optimization. Use the remaining guidance in this series to implement these principles through consistent approaches to automation.

- **Cost optimization:** Continuously optimize operating costs by using the getting started guide on managing enterprise costs
- **Security:** Reduce risk by integrating enterprise security across the portfolio by using the getting started guide on implementing security across the portfolio.
- **Performance management:** Ensure IT asset performance supports business processes by using the getting started guide on performance management across the enterprise.
- **Reliability:** Improve reliability and reduce business disruptions by using the getting started guide on implementing controls to create reliability.

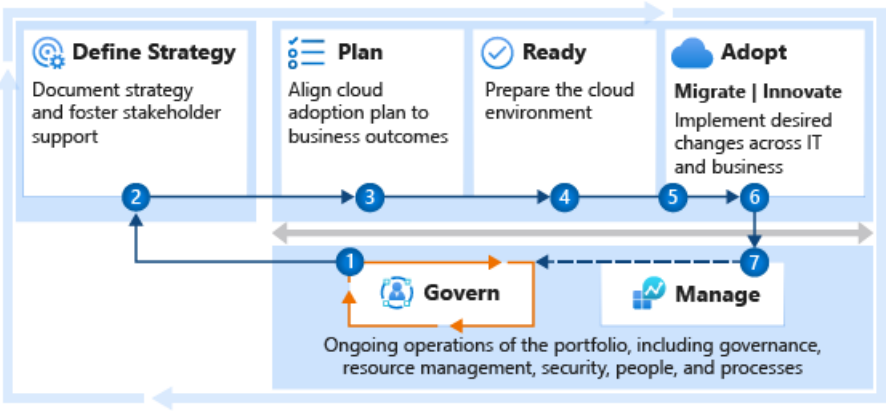
Get started: Manage cloud costs

7/21/2021 • 8 minutes to read • Edit Online

The Cost Management discipline of cloud governance focuses on establishing budgets, monitoring cost allocation patterns, and implementing controls to improve cloud spending behaviors across the IT portfolio. Enterprise cost optimization involves many other roles and functions to minimize cost and balance the demands of scale, performance, security, and reliability. This article maps those various supporting functions into a getting started guide that helps create alignment among the involved teams.

However, enterprise cost optimization involves many other roles and functions to minimize cost and balance the demands of scale, performance, security, and reliability. This article maps those supporting functions to help create alignment between the involved teams.

Governance is the cornerstone of cost optimization within any large enterprise. The following section outlines cost optimization guidance within the context of governance. The subsequent steps help each team take actions that target its role in cost optimization. Together, these steps will help your organization get started on a journey toward cost optimization.



Step 1: Optimize enterprise costs

The cloud governance team is well prepared to evaluate and act on overspending or unplanned spending through a combination of monitoring performance, reducing resource sizing, and safely terminating unused resources. Enterprise cost optimization starts with a shared team understanding of the tools, processes, and dependencies required to wisely act on cost concerns at an environment level.

Deliverables:

- Implement wise changes to your Cost Management policies across the enterprise.
- Document your Cost Management policies, processes, and design guidance in the Cost Management discipline template.

These deliverables are the result of a few recurring tasks:

- Ensure strategic alignment with the cloud strategy team (which includes workload stakeholders across the portfolio).
- Optimize cost across the environment:
  - Manually or automatically shut down unused VMs.
  - Delete or deallocate stopped VMs.
  - Ensure proper resource sizing.

    - Align spending to budget expectations.
- Validate any architectural change by using the Microsoft Azure Well-Architected Review to facilitate a conversation with technical owners of the workloads.

Guidance to support deliverable completion:

- Ensure that all workloads and resources follow proper naming and tagging conventions. Enforce tagging conventions by using Azure Policy with a specific emphasis on tags for "cost center" and "technical owner."
- On a regular basis, review and apply Cost Management discipline best practices to guide analysis and improvements across the enterprise. Important governance practices include:
    - Acting on general cost best practices to reduce sizing and costs and to stop unused machines.
    - Applying hybrid use benefits to reduce licensing costs.
    - Aligning reserved instances to reduce resource costs.
    - Monitoring resource utilization to minimize impacts on resource performance.
    - Reducing nonproduction costs through policies to govern nonproduction environments.
    - Acting on cost optimization recommendations.
- Trade-offs at the workload level might be needed to implement effective cost optimization changes. The Microsoft Azure Well-Architected Framework and Microsoft Azure Well-Architected Review can help guide those conversations with the technical owner of a specific workload.
- If you're new to cloud governance, establish governance policies, processes, and disciplines using the Govern methodology.
- If you're new to the Cost Management discipline, consider following the Cost Management discipline improvements article, with a focus on the Implementation section.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Cloud governance team	• Cloud strategy team • Cloud adoption team • Cloud center of excellence or central IT team

The governance team can detect and drive significant cost optimization across most enterprises. Basic, data-driven resource sizing can have an immediate and measurable impact on costs.

As discussed in Build a cost-conscious organization, an enterprise-wide focus on cost management and cost optimization can deliver much more value. The following steps demonstrate ways the various teams can help build a cost-conscious organization.

Step 2: Define a strategy

Strategic decisions directly affect cost controls, rippling through the adoption lifecycle and into long-term operations. Strategic clarity will improve cost optimization efforts, driven by the governance team.

Deliverables:

- Record motivations, outcomes, and business justification in the strategy and plan template.
- Create your first budget by using Azure Cost Management + Billing.

Guidance to support deliverable completion:

- Understand motivations. Critical business events and some migration motivations tend to be cost sensitive,

increasing the importance of cost control for all later efforts. Other forward-looking motivations related to innovation or growth through migration might focus more on top-line revenue. Understanding motivations will help you decide how high to prioritize your cost management.

- Business outcomes. Some fiscal outcomes tend to be extremely cost-sensitive. When the desired outcomes map to fiscal metrics, you should invest in the Cost Management governance discipline very early.
- Business justification. The business justification serves as a high-level view of the financial plan for cloud adoption. This is a good source for initial budgeting efforts.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Cloud strategy team	• Cloud governance team • Cloud adoption team • Cloud center of excellence or central IT team

Step 3: Develop a cloud adoption plan

The adoption plan provides clarity on the timeline of activities during adoption. Aligning the plan and the digital estate analysis allows you to forecast your monthly growth in spending. It also helps your cloud governance team align processes and identify spending patterns.

Deliverables:

- Complete steps 1 through 6 of building a cloud adoption plan.
- Work with your cloud governance team to refine budgets and create realistic spending forecasts.

Guidance to support deliverable completion:

- Gather inventory. Establish a source of data for analysis of the digital estate before adoption.
- Best practice: Azure Migrate. Use Azure Migrate to gather inventory.
- Incremental rationalization. During incremental rationalization and quantitative analysis, identify cloud candidates for budgeting purposes.
- Align cost models and forecast models. Use Azure Cost Management + Billing to align cost and forecast models by creating budgets.
- Build your cloud adoption plan. Build a plan with actionable workload, assets, and timeline details. This plan provides the basis for spending over time (or cost forecasting). *Spending over time* is the initial baseline for all actionable optimization analysis within the Cost Management governance discipline.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Cloud adoption team	• Cloud strategy team • Cloud governance team • Cloud center of excellence or central IT team

Step 4: Implement best practices for landing zones

The Ready methodology of the Microsoft Cloud Adoption Framework for Azure focuses heavily on the development of landing zones to host workloads in the cloud. During implementation of landing zones, an organization should consider various decisions for cost optimization.

Deliverables:

- Deploy one or more landing zones that can host workloads in the short-term adoption plan.
- Ensure that all landing zones meet cost optimization decisions and cost management requirements.

Guidance to support deliverable completion:

- Track costs. Establish a well-managed environment hierarchy, provide the right level of cost access, and use additional cost management resources in each landing zone.
- Optimize your cloud investment. Understand best practices for optimizing investments.
- Create and manage budgets. Understand best practices for creating and managing budgets.
- Optimize costs from recommendations. Understand best practices for using recommendations that will optimize costs.
- Monitor usage and spending. Understand best practices for monitoring usage and spending within a landing zone.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Cloud adoption team	• Cloud strategy team • Cloud governance team • Cloud center of excellence or central IT team

Step 5: Complete waves of migration effort

Migration is a repeatable process executed by the cloud adoption team. Throughout this process, there are many opportunities to optimize costs across your portfolio. Many of these in-process decisions are applied to a small group of workloads during each migration wave or iteration.

Deliverables:

- Benchmark, test, resize, and deploy a collection of fully optimized workloads.

Guidance to support deliverable completion:

- Migration-focused cost-control mechanisms provides insights about the cloud-native cost optimization controls that help during migration.
- Best practices for optimizing cost of migrated workloads contains a checklist of 14 best practices to follow before and after migration to maximize cost optimization of each workload release.

Long-term operational costs are a common theme in each area of migration process improvements. This list of process improvements is organized by the phase of the migration process:

- Prerequisites provides information on managing change and the backlog, which influences both budgeted and actual cloud costs.
- Assess provides six specific processes, from validating assumptions to understanding partner options. Each process influences cloud optimization opportunities.
- Migrate contains one process suggestion about remediating assets. This suggestion provides an opportunity to optimize the as-configured state, in favor of an optimized solution.
- Promote focuses heavily on testing, resizing, validating, and releasing migrated assets, along with decommissioning retired assets. This is the first clear point at which forecasts and budgets can be tested against actual performance and configuration.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Cloud adoption team	• Cloud strategy team • Cloud governance team • Cloud center of excellence or central IT team

Step 6: Drive customer-focused innovation

Innovation and development of new products require a much deeper degree of architectural review. The Cloud Adoption Framework provides details on the innovation process and product management thinking. Cost optimization decisions about innovations are largely out of scope in this guidance.

Deliverables:

- Make key architectural decisions about innovations to balance cost and other critical design considerations.

Guidance to support deliverable completion:

- Use the Microsoft Azure Well-Architected Review to understand the balance in architecture decisions.
- Review the Microsoft Azure Well-Architected Framework for deeper guidance on cost optimization during innovation.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Cloud adoption team	• Cloud strategy team • Cloud governance team • Cloud center of excellence or central IT team

Step 7: Implement sound operations

Establishing a solid management baseline will help you collect data and create operational alerts. This effort can aid in detecting opportunities to optimize costs. It will create a balance between resiliency and cost optimization.

Deliverables:

- Monitor your enterprise environment for ongoing recommendations to optimize costs, aligned to the criticality and resiliency classifications of each workload.

Guidance to support deliverable completion:

- Create business alignment to gain clarity regarding criticality and appetite for resiliency investments.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Cloud operations team	• Cloud strategy team • Cloud governance team • Cloud center of excellence or central IT team

Value statement

Following these steps helps you build a cost-conscious organization. Simplify cost optimization by using shared ownership and driving collaboration with the right teams at the right times.

Get started: Implement security across the enterprise environment

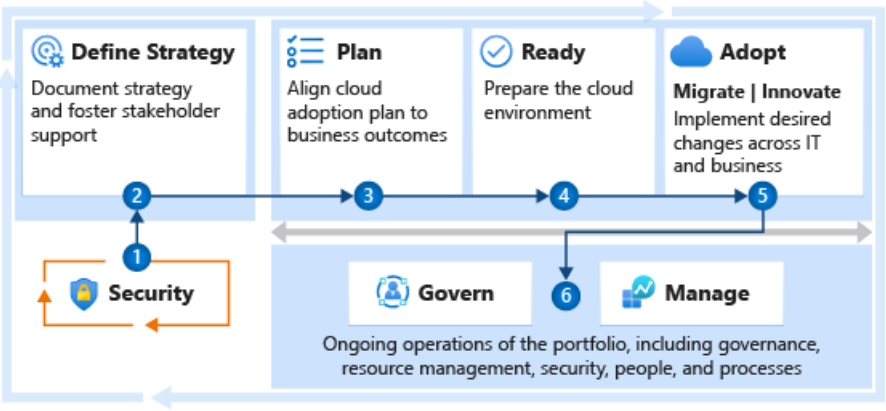
7/21/2021 • 13 minutes to read • Edit Online

Security helps create assurances of confidentiality, integrity, and availability for a business. Security efforts have a critical focus on protecting against the potential impact to operations caused by both internal and external malicious and unintentional acts.

This getting started guide outlines the key steps that will mitigate or avoid the business risk from cybersecurity attacks. It can help you rapidly establish essential security practices in the cloud and integrate security into your cloud adoption process.

The steps in this guide are intended for all roles that support security assurances for cloud environments and landing zones. Tasks include immediate risk mitigation priorities, guidance on building a modern security strategy, operationalizing the approach, and executing on that strategy.

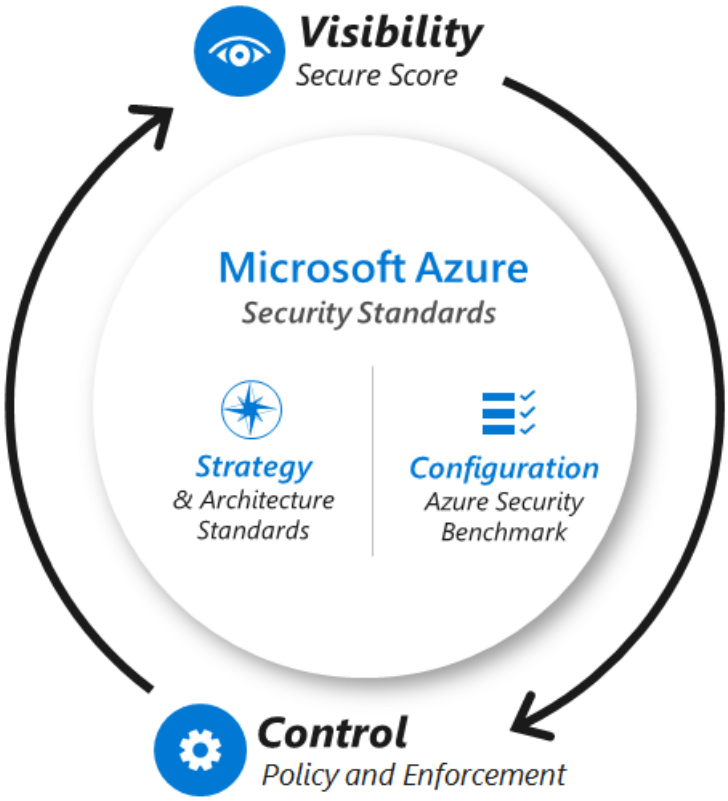
This guide includes elements from across the Microsoft Cloud Adoption Framework for Azure:



Adhering to the steps in this guide will help you integrate security at critical points in the process. The goal is to avoid obstacles in cloud adoption and reduce unnecessary business or operational disruption.

Microsoft has built capabilities and resources to help accelerate your implementation of this security guidance on Microsoft Azure. You'll see these resources referenced throughout this guide. They're designed to help you establish, monitor, and enforce security, and they're frequently updated and reviewed.

The following diagram shows a holistic approach for using security guidance and platform tooling to establish security visibility and control over your cloud assets in Azure. We recommend this approach.



Use these steps to plan and execute your strategy for securing your cloud assets and using the cloud to modernize security operations.

Step 1: Establish essential security practices

Security in the cloud starts with applying the most important security practices to the people, process, and technology elements of your system. Additionally, some architectural decisions are foundational and are very difficult to change later so should be carefully applied.

Whether you're already operating in the cloud or you're planning for future adoption, we recommend that you follow these 11 essential security practices (in addition to meeting any explicit regulatory compliance requirements).

People:

1. Educate teams about the cloud security journey
2. Educate teams on cloud security technology

Process:

3. Assign accountability for cloud security decisions
4. Update incident response processes for cloud
5. Establish security posture management

Technology:

6. Require passwordless or multifactor authentication
7. Integrate native firewall and network security
8. Integrate native threat detection

Foundational architecture decisions:

9. Standardize on a single directory and identity
10. Use identity-based access control (instead of keys)
11. Establish a single unified security strategy

> **NOTE**
> Each organization should define its own minimum standards. Risk posture and subsequent tolerance to that risk can vary widely based on industry, culture, and other factors. For example, a bank might not tolerate any potential damage to its reputation from even a minor attack on a test system. Some organizations would gladly accept that same risk if it accelerated their digital transformation by three to six months.

Step 2: Modernize the security strategy

Effective security in the cloud requires a strategy that reflects the current threat environment and the nature of the cloud platform that's hosting the enterprise assets. A clear strategy improves the effort of all teams to provide a secure and sustainable enterprise cloud environment. The security strategy must enable defined business outcomes, reduce risk to an acceptable level, and enable employees to be productive.

A cloud security strategy provides guidance to all teams working on the technology, processes, and people readiness for this adoption. The strategy should inform the cloud architecture and technical capabilities, guide the security architecture and capabilities, and influence the training and education of teams.

Deliverables:

The strategy step should result in a document that can easily be communicated to many stakeholders within the organization. The stakeholders can potentially include executives on the organization's leadership team.

We recommended capturing the strategy in a presentation to facilitate easy discussion and updating. This presentation can be supported with a document, depending on the culture and preferences.

- **Strategy presentation:** You might have a single strategy presentation, or you might choose to also create summary versions for leadership audiences.
  - **Full presentation:** This should include the full set of elements for the security strategy in the main presentation or in optional reference slides.
  - **Executive summaries:** Versions to use with senior executives and board members might contain only critical elements relevant to their role, such as risk appetite, top priorities, or accepted risks.
- You can also record motivations, outcomes, and business justifications in the strategy and plan template.

Best practices for building security strategy:

Successful programs incorporate these elements into their security strategy process:

- **Align closely to business strategy:** Security's charter is to protect business value. It's critical to align all security efforts to that purpose and minimize internal conflict.
  - **Build a shared understanding** of business, IT, and security requirements.
  - **Integrate security early into cloud adoption** to avoid last-minute crises from avoidable risks.
  - **Use an agile approach** to immediately establish minimum security requirements and continuously improve security assurances over time.
  - **Encourage security culture change** through intentional proactive leadership actions.

  For more information, see Transformations, mindsets, and expectations.
- **Modernize security strategy:** The security strategy should include considerations for all aspects of modern technology environment, current threat landscape, and security community resources.

- Adapt to the shared responsibility model of the cloud.
- Include all cloud types and multicloud deployments.
- Prefer native cloud controls to avoid unnecessary and harmful friction.
- Integrate the security community to keep up with the pace of attacker evolution.

Related resources for additional context:

- Evolution of threat environment, roles, and digital strategies
- Transformation of security, strategies, tools, and threats
- Strategy considerations for the Cloud Adoption Framework:
  - Modernize your security strategy
  - Cybersecurity resilience
  - How cloud is changing security relationships and responsibilities

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Security leadership team (chief information security officer (CISO) or equivalent)	• Cloud strategy team • Cloud security team • Cloud adoption team • Cloud center of excellence or central IT team

Strategy approval:

Executives and business leaders with accountability for outcomes or risks of business lines within the organization should approve this strategy. This group might include the board of directors, depending on the organization.

Step 3: Develop a security plan

Planning puts the security strategy into action by defining outcomes, milestones, timelines, and task owners. This plan also outlines the roles and responsibilities of the teams.

Security planning and cloud adoption planning should not be done in isolation. It's critical to invite the cloud security team into the planning cycles early, to avoid work stoppage or increased risk from security issues being discovered too late. Security planning works best with in-depth knowledge and awareness of the digital estate and existing IT portfolio that comes from being fully integrated into the cloud planning process.

Deliverables:

- **Security plan:** A security plan should be part of the main planning documentation for the cloud. It might be a document that uses the strategy and plan template, a detailed slide deck, or a project file. Or it might be a combination of these formats, depending on the organization's size, culture, and standard practices.

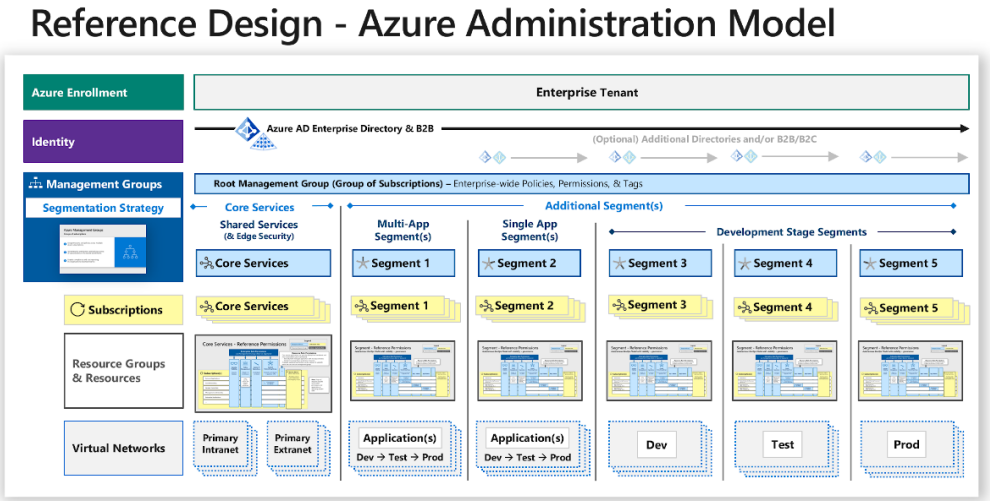
  The security plan should include all of these elements:

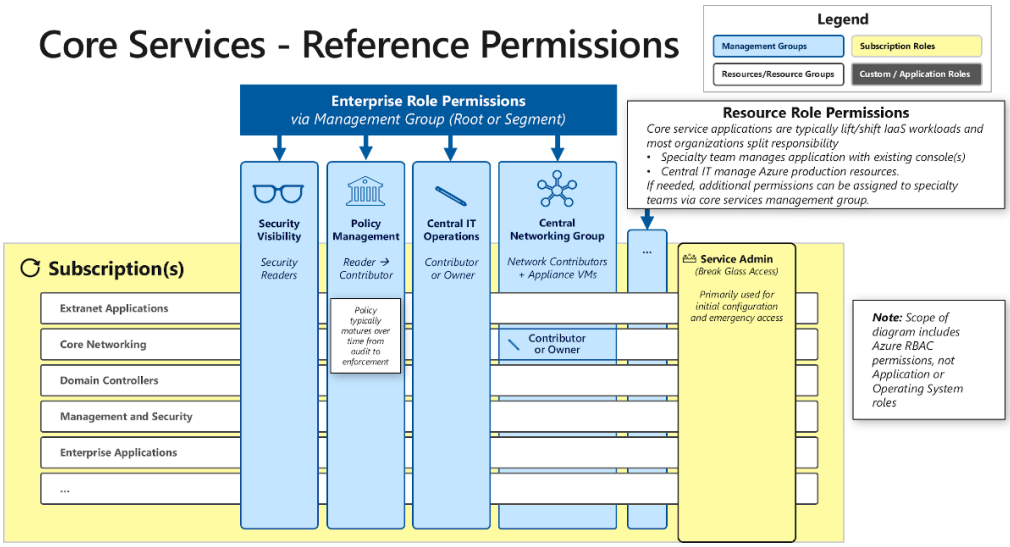
  - **Organizational functions plan**, so teams know how current security roles and responsibilities will change with the move to the cloud.
  - **Security skills plan** to support team members as they navigate the significant changes in technology, roles, and responsibilities.
  - **Technical security architecture and capabilities roadmap** to guide technical teams.

    Microsoft provides reference architectures and technology capabilities to help you as you build

your architecture and roadmap, including:

- Azure components and reference model to accelerate planning and design of Azure security roles.





- Microsoft cybersecurity reference architecture to build a cybersecurity architecture for a hybrid enterprise that spans on-premises and cloud resources.
- Security operations center (SOC) reference architecture to modernize security detection, response, and recovery.
- Zero-trust user access reference architecture to modernize access control architecture for cloud generation.
- Azure Security Center and Microsoft cloud application security to help secure cloud assets.

- **Security awareness and education plan**, so all teams have basic critical security knowledge.
- **Asset sensitivity marking** to designate sensitive assets by using a taxonomy aligned to business impact. The taxonomy is built jointly by business stakeholders, security teams, and other interested parties.
- **Security changes to the cloud plan:** Update other sections of the cloud adoption plan to reflect changes triggered by the security plan.

Best practices for security planning:

Your security plan is likely to be more successful if your planning takes the approach of:

- **Assume a hybrid environment:** That includes software as a service (SaaS) applications and on-premises environments. It also includes multiple cloud infrastructure as a service (IaaS) and platform as a service (PaaS) providers, if applicable.
- **Adopt agile security:** Establish minimum security requirements first and move all noncritical items to a prioritized list of next steps. This should not be a traditional, detailed plan of 3-5 years. The cloud and threat environment change too fast to make that type of plan useful. Your plan should focus on developing the beginning steps and end state:
  - **Quick wins** for the immediate future that will deliver a high impact before longer-term initiatives begin. The time frame can be 3-12 months, depending on organizational culture, standard practices, and other factors.
  - **Clear vision** of the desired end state to guide each team's planning process (which might take multiple years to achieve).
- **Share the plan broadly:** Increase awareness of, feedback from, and buy-in by stakeholders.
- **Meet the strategic outcomes:** Ensure that your plan aligns to and accomplishes the strategic outcomes described in the security strategy.
- **Set ownership, accountability, and deadlines:** Ensure that the owners for each task are identified and are committed to completing that task in a specific time frame.
- **Connect with the human side of security:** Engage people during this period of transformation and new expectations by:
  - **Actively supporting team member transformation** with clear communication and coaching on:
    - What skills they need to learn.
    - Why they need to learn the skills (and the benefits of doing so).
    - How to get this knowledge (and provide resources to help them learn).

    You can document the plan by using the strategy and plan template. And you can use online Microsoft security training to help with education of your team members.
  - **Making security awareness engaging** to help people genuinely connect with their part of keeping the organization safe.
- **Review Microsoft learnings and guidance:** Microsoft has published insights and perspectives to help your organization plan its transformation to the cloud and a modern security strategy. The material includes recorded training, documentation, and security best practices and recommended standards.

  For technical guidance to help build your plan and architecture, see the Microsoft security documentation.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Cloud security team	• Cloud strategy team • Cloud governance team • Any risk teams in your organization • Cloud center of excellence or central IT team

Security plan approval:

The security leadership team (CISO or equivalent) should approve the plan.

Step 4: Secure new workloads

It's a lot easier to start in a secure state than to retrofit security later into your environment. We strongly recommend starting with a secure configuration to ensure that workloads are migrated to, and developed and tested in, a secure environment.

During landing zone implementation, many decisions can affect security and risk profiles. The cloud security team should review the landing zone configuration to ensure that it meets the security standards and requirements in your organization's security baselines.

Deliverables:

- Ensure that new landing zones meet the organization's compliance and security requirements.

Guidance to support deliverable completion:

- **Blend existing requirements and cloud recommendations:** Start with recommended guidance and then adapt this to your unique security requirements. We have seen challenges with trying to enforce existing on-premises policies and standards, because these often refer to outdated technology or security approaches.

  Microsoft has published guidance to help you build your security baselines:

  - Azure security standards for strategy and architecture: Strategy and architectural recommendations to shape your environment's security posture.
  - Azure security benchmarks: Specific configuration recommendations for securing Azure environments.
  - Azure security baseline training.
- **Provide guardrails:** Safeguards should include automated policy auditing and enforcement. For these new environments, teams should strive to both audit and enforce the organization's security baselines. These efforts can help minimize security surprises during the development of workloads, as well as continuous integration and continuous deployment (CI/CD) of workloads.

  Microsoft provides several native capabilities in Azure to enable this:

  - Secure score: Use a scored assessment of your Azure security posture to track security efforts and projects in your organization.
  - Azure Blueprints: Cloud architects and centralized IT groups can define a repeatable set of Azure resources that implements and adheres to an organization's standards, patterns, and requirements.
  - Azure Policy: This is the foundation of the visibility and control capabilities that the other services use. Azure Policy is integrated into Azure Resource Manager, so you can audit changes and enforce policies across any resource in Azure before, during, or after its creation.
  - Improve landing zone operations: Use best practices for improving security within a landing zone.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Cloud security team	• Cloud adoption team • Cloud platform team • Cloud strategy team • Cloud governance team • Cloud center of excellence or central IT team

Step 5: Secure existing cloud workloads

Many organizations have already deployed assets to enterprise cloud environments without applying the security best practices, creating increased business risk.

After you ensure that new applications and landing zones follow security best practices, you should focus on bringing existing environments up to the same standards.

Deliverables:

- Ensure that all existing cloud environments and landing zones meet the organization's compliance and security requirements.
- Test operational readiness of production deployments by using policies for security baselines.
- Validate adherence to design guidance and security requirements for security baselines.

Guidance to support deliverable completion:

- Use the same security baselines that you built in Step 4 as your ideal state. You might have to adjust some policy settings to only audit instead of enforcing them.
- Balance operational and security risk. Because these environments might host production systems that enable critical business processes, you might need to implement security improvements incrementally to avoid risking operational downtime.
- Prioritize the discovery and remediation of security risk by business criticality. Start with workloads that have a high business impact if compromised and workloads that have a high exposure to risk.

For more information, see Identify and classify business-critical applications.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Cloud adoption team	• Cloud adoption team • Cloud strategy team • Cloud security team • Cloud governance team • Cloud center of excellence or central IT team

Step 6: Govern to manage and improve security posture

Like all modern disciplines, security is an iterative process that should focus on continuous improvement. Security posture can also decay if organizations don't sustain focus on it over time.

Consistent application of security requirements comes from sound governance disciplines and automated solutions. After the cloud security team defines the security baselines, those requirements should be audited to ensure they're applied consistently to all cloud environments (and enforced where applicable).

Deliverables:

- Ensure that the organization's security baselines are applied to all relevant systems. Audit anomalies by using a secure score or a similar mechanism.
- Document your Security Baseline policies, processes, and design guidance in the Security Baseline discipline template.

Guidance to support deliverable completion:

- Use the same security baselines and auditing mechanisms that you built in Step 4 as technical components of monitoring the baselines. Complement these baselines with people and process controls to ensure consistency.
- Ensure that all workloads and resources follow proper naming and tagging conventions. Enforce tagging conventions by using Azure Policy, with a specific emphasis on tags for data sensitivity.
- If you're new to cloud governance, establish governance policies, processes, and disciplines by using the Govern methodology.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Cloud governance team	• Cloud strategy team • Cloud security team • Cloud center of excellence or central IT team

Next steps

The steps in this guide have helped you implement the strategy, controls, processes, skills, and culture needed to consistently manage security risks across the enterprise.

As you continue into the operations mode of cloud security, consider these next steps:

- Review Microsoft security documentation. It provides technical guidance to help security professionals build and improve cybersecurity strategy, architecture, and prioritized roadmaps.
- Review security information in Built-in security controls for Azure services.
- Review Azure security tools and services in Security services and technologies available on Azure.
- Review the Microsoft Trust Center. It contains extensive guidance, reports, and related documentation that can help you perform risk assessments as part of your regulatory compliance processes.
- Review third-party tools available to facilitate meeting your security requirements. For more information, see Integrate security solutions in Azure Security Center.

Get started: Improve reliability with the right controls

7/21/2021 • 7 minutes to read • Edit Online

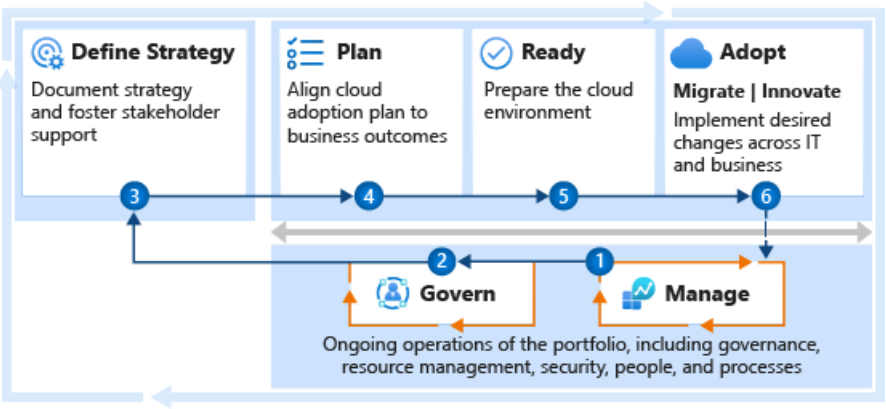
How do you apply the right controls to improve reliability? This article helps you minimize disruptions related to:

- Inconsistencies in configuration.
- Resource organization.
- Security baselines.
- Resource protection.

The steps in this article help the operations team balance reliability and cost across the IT portfolio. This article also helps the governance team to ensure that balance is applied consistently. Reliability also depends on other roles and functions. This article maps supporting functions to help you create alignment among the involved teams.

Operations management and governance are equal partners in enterprise reliability. The decisions you make about operational practices set the baseline for reliability. The approaches used to govern the overall environment ensure consistency across all resources.

The first two steps in this article help both teams get started. They're listed sequentially, but you can perform them in parallel. The subsequent steps help you get the entire enterprise started on a shared journey toward more reliable solutions throughout the enterprise.



Step 1: Establish operations management requirements

Not all workloads are created equal. In any environment, there are workloads that have a direct and constant impact on the business. There are also supporting business processes and workloads that have a smaller impact on the overall business. In this step, the cloud operations team identifies and implements initial requirements to support the overall IT portfolio.

Deliverables:

- Implement a management baseline to define the standard operations that are required for all production workloads.
- Negotiate business commitments with the cloud strategy team to develop a plan for advanced operations and resiliency requirements.
- Expand your management baseline, if additional operations are required for the majority of workloads.

- Apply advanced operations requirements to landing zones and resources that support the workloads that are most critical.
- Document operations decisions across the IT portfolio in the operations management workbook.

Guidance to support deliverable completion:

- Management baseline:
  - Inventory and visibility: Cloud-native tools can help you collect data and configure alerts. The tools also can help you implement the monitoring platform that best fits your operating model.
  - Operational compliance: The highest percentages of outages tend to come from changes to resource configuration or poor maintenance practices. Follow the Azure server management guide to implement cloud-native tools to manage patching and changes to resource configuration.
  - Protection and recovery: Outages are inevitable on any platform. When a disruption occurs, be prepared with backup and recovery solutions to minimize the duration.
- Advanced operations: Use the management baseline as the foundation for your business alignment conversations. It helps you to clearly discuss criticality, business impact, and operations commitments. Business alignment helps quantify and validate requests for an enhanced baseline, management of specific technology platforms, or workload-specific operations.
- **Guide an architecture review:** Architecture changes at the workload level might be required to meet operations requirements. The Microsoft Azure Well-Architected Framework and Microsoft Azure Well-Architected Review can help guide those conversations with the technical owner of a specific workload.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Cloud operations team	• Cloud strategy team • Cloud adoption team • Cloud governance team • Cloud center of excellence or central IT team

Step 2: Consistently apply the management baseline

Enterprise reliability requires consistent application of the management baseline. That consistency comes from appropriate corporate policy, IT processes, and automated tools. These resources govern the implementation of the management baseline for all affected resources.

Deliverables:

- Ensure proper application of the management baseline for all affected systems.
- Document your Resource Consistency policies, processes, and design guidance in the Resource Consistency discipline template.

Guidance to support deliverable completion:

- Ensure all workloads and resources follow proper naming and tagging conventions. Enforce tagging conventions by using Azure Policy, with a specific emphasis on tags for criticality.
- If you're new to cloud governance, establish governance policies, processes, and disciplines by using the Govern methodology.
- If you're new to the Cost Management discipline, follow the guidance in the Cost Management discipline improvements article. Focus on the Implementation section.

> **NOTE**
>
> **Steps to start reliability partnerships with other teams:** Various decisions throughout the cloud adoption lifecycle can have a direct impact on reliability. The following steps outline the partnerships and supporting efforts required to deliver consistent reliability across the IT portfolio.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Cloud governance team	• Cloud strategy team • Cloud operations team • Cloud center of excellence or central IT team

Step 3: Define your strategy

Strategic decisions directly affect reliability. They ripple through the adoption lifecycle and into long-term operations. Strategic clarity improves reliability efforts.

Deliverables:

- Record motivations, outcomes, and business justification in the strategy and plan template.
- Ensure the management baseline provides operational support that aligns to the strategic direction of cloud adoption.

Guidance to support deliverable completion:

- Understand motivations: Critical business events and some migration motivations tend to be cost sensitive. These areas can increase the importance of cost control for all later efforts. Other forward-looking motivations related to innovation or growth through migration might be focused more on top-line revenue. Understanding motivations helps you prioritize your cost management.
- Business outcomes: Some fiscal outcomes tend to be extremely cost sensitive. When the desired outcomes map to fiscal metrics, you should invest early in the Cost Management governance discipline.
- Business justification: The business justification serves as a high-level view of the overall financial plan for cloud adoption. It can be a good source for initial budgeting efforts.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Cloud strategy team	• Cloud governance team • Cloud operations team • Cloud center of excellence or central IT team

Step 4: Develop a cloud adoption plan

The digital estate (or analysis of the existing IT portfolio) can help you to validate the business justification. It can provide a refined view of the overall IT portfolio. The adoption plan provides clarity on the timeline of activities during adoption.

When you align the adoption plan with the digital estate analysis, you can plan for future operations management dependencies. Understanding the adoption plan also invites the cloud operations team into the development cycles. They can evaluate and plan for any changes to the management baseline that are required to provide workload operations.

Deliverables:

- Update the strategy and plan template to reflect changes that are needed to achieve the desired strategy. The changes recorded can include:
  - An assessment of the existing digital estate.
  - A cloud adoption plan that reflects the required changes and the work involved.
  - The organizational change that's required to deliver on the plan.
  - A plan for addressing the skills that are needed to enable the existing team to successfully complete the required work.
- Work with the governance team to align cost models and forecast models. This process includes efforts to start optimizing spend through quantitative analysis.

Guidance to support deliverable completion:

- Gather inventory: Establish a source of data for analysis of the digital estate prior to adoption.
- Best practice: Azure Migrate: Use Azure Migrate to gather inventory.
- Incremental rationalization: During incremental rationalization, a quantitative analysis can identify cloud candidates for budgeting purposes.
- Align cost models and forecast models: Use Azure Cost Management + Billing to align cost and forecast models by creating budgets.
- Build your cloud adoption plan: Build a plan with actionable workload, assets, and timeline details.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Cloud strategy team	• Cloud adoption team • Cloud governance team • Cloud operations team • Cloud center of excellence or central IT team

Step 5: Implement landing zone best practices

The Ready methodology of the Cloud Adoption Framework focuses heavily on the development of landing zones to host workloads in the cloud. During landing zone implementation, multiple decisions could affect operations. Consult the cloud operations team to help review the landing zone for operations improvements. Also consult the cloud governance team to understand Resource Consistency policies and design guidance that might affect the landing zone design.

Deliverables:

- Deploy one or more landing zones capable of hosting workloads in the short-term adoption plan.
- Ensure that all landing zones meet operations decisions and resource consistency requirements.

Guidance to support deliverable completion:

- Improve landing zone operations: Best practices for improving operations within a given landing zone.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Cloud adoption team	• Cloud operations team • Cloud strategy team • Cloud governance team • Cloud center of excellence or central IT team

Step 6: Complete waves of adoption effort and change

Long-term operations can be affected by the decisions made during migration and innovation efforts. Maintaining consistent alignment early in adoption processes helps to remove barriers to production releases. It also reduces the effort that's required to introduce new solutions into operations management practices.

Deliverables:

- Test operational readiness of production deployments by using Resource Consistency policies.
- Validate adherence to resource consistency design guidance and operations requirements.
- Document any advanced operations requirements in the operations management workbook.

Guidance to support deliverable completion:

- Environmental readiness checklist
- Pre-promotion checklist
- Production release checklist

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Cloud adoption team	• Cloud strategy team • Cloud operations team • Cloud governance team • Cloud center of excellence or central IT team

Value statement

These steps help you to implement the controls and processes that are needed to ensure reliability across the enterprise and all hosted resources.

Get started: Ensure consistent performance across a portfolio

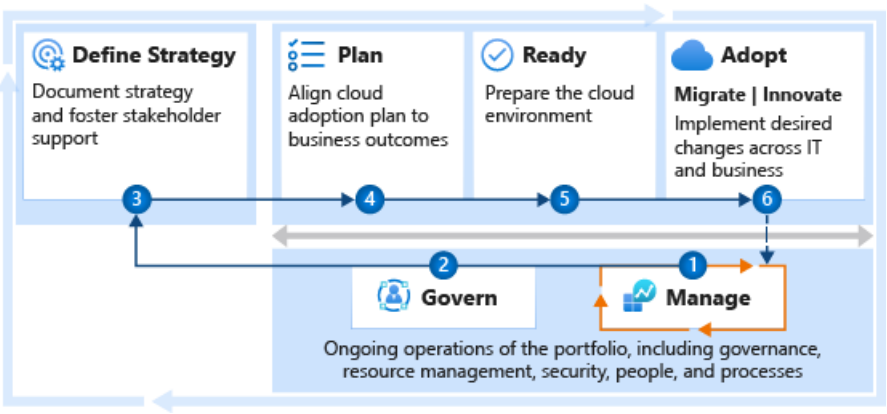
7/21/2021 • 6 minutes to read • Edit Online

How do you ensure adequate performance across a portfolio of workloads? The steps in this guide can help you establish processes for maintaining that level of performance.

Performance also depends on other roles and functions. This article maps those supporting functions to help you create alignment among the involved teams.

Centralized operations management is the most common approach to consistent performance across the portfolio. Decisions about operational practices define the operations baseline and any holistic enhancements.

The first step in this guide helps the operations team get started. The subsequent steps help the entire enterprise get started on a shared journey toward enterprise performance across the portfolio of workloads.



Step 1: Establish operations management requirements

The operations management baseline, outlined in the Microsoft Cloud Adoption Framework for Azure, provides a set of controls and cloud-native operations tools to ensure consistent operations. Expanding that baseline with automation tooling provides performance monitoring and automation to meet consistent performance requirements across the portfolio.

Deliverables:

- Enhance the management baseline to include automated remediation tasks related to deviations from performance expectations.
- When workload-specific data patterns or architecture changes are needed to meet performance requirements, use workload-specific operations to provide greater performance controls.
- Document operational decisions across the IT portfolio in the operations management workbook. Focus on including performance automation decisions in the `Operational Compliance` section of the `Baseline` tab.

Guidance to support deliverable completion:

- The enhanced management baseline article outlines examples of using tools like Azure Automation to add performance-related enhancements. This approach can aid in maintaining consistent performance through basic modifications to the size and scale of supporting assets.
- Workload-specific operations uses the Microsoft Azure Well-Architected Review to provide guidance on automation for a specific workload. This approach to performance management is particularly useful when

workload-specific data should drive operational actions.

- The preceding guidance assumes that an existing implementation of a management baseline supports the full portfolio of workloads.

> **NOTE**
> Various decisions throughout the cloud adoption lifecycle can have a direct impact on performance. The following steps help outline the partnerships and supporting efforts required to deliver performance across the IT portfolio.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Cloud operations team	• Cloud strategy team • Cloud adoption team • Cloud governance team • Cloud center of excellence or central IT team

Step 2: Consistent application of the management baseline

As the management baseline is improved, it's important to ensure that those improvements carry through to the Resource Consistency governance discipline. Doing so ensures the application of the enhanced baseline in all managed environments.

Deliverables:

- Ensure proper application of the enhanced management baseline for all affected systems.
- Document your policies, processes, and design guidance for resource consistency in the Resource Consistency discipline template.

Guidance to support deliverable completion:

- Ensure that all workloads and resources follow proper naming and tagging conventions. Enforce tagging conventions by using Azure Policy, with a specific emphasis on tags for "criticality."
- If you're new to cloud governance, establish governance policies, processes, and disciplines by using the Govern methodology.
- If you're new to the Cost Management discipline, consider following the article about Cost Management discipline improvements, with a focus on the Implementation section.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Cloud governance team	• Cloud strategy team • Cloud operations team • Cloud center of excellence or central IT team

Step 3: Define strategy

Strategic decisions directly affect performance, rippling through the adoption lifecycle and into long-term operations. Strategic clarity improves performance efforts across the portfolio. That clarity also helps the operations team understand which workloads need a degree of workload specialization and advanced operations.

Deliverables:

- Record motivations, outcomes, and business justification in the strategy and plan template.
- Ensure that the management baseline provides operational support that aligns with the strategic direction of cloud adoption.

Guidance to support deliverable completion:

- Understand motivations: Critical business events and some migration motivations tend to be cost sensitive, which increases the importance of cost control for all later efforts. Other forward-looking motivations related to innovation or growth through migration might be focused more on top-line revenue. Understanding motivations can help you decide how high to prioritize your cost management.
- Business outcomes: Some fiscal outcomes tend to be extremely cost sensitive. When the desired outcomes map to fiscal metrics, you should invest in the Cost Management governance discipline early.
- Business justification: The business justification serves as a high-level view of the financial plan for cloud adoption. This can be a good source for initial budgeting efforts.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Cloud strategy team	• Cloud governance team • Cloud operations team • Cloud center of excellence or central IT team

Step 4: Assess and plan for workload adoption

The digital estate (or analysis of the existing IT portfolio) can aid in validating the business justification and provide a refined view of the IT portfolio. The adoption plan provides clarity on the timeline of activities during adoption. Aligning that plan and digital estate analysis provides a means of planning for future dependencies on operations management.

Understanding the plan also invites the cloud operations team into the development cycle. The team can then evaluate and plan for any changes to the management baseline that are required to provide workload operations.

Deliverables:

- Update the strategy and plan template to reflect changes triggered by the digital estate analysis.
- Work with the cloud operations team to clearly define the criticality and business impact of each workload in the near-term and long-term adoption plan.
- Work with the cloud operations team to establish a timeline for operations readiness.

Guidance to support deliverable completion:

- Gather inventory: Establish a source of data for analysis of the digital estate before adoption.
- Best practice: Azure Migrate: Use Azure Migrate to gather inventory.
- Incremental rationalization: During incremental rationalization, use a quantitative analysis to identify cloud candidates for budgeting purposes.
- Align cost models and forecast models: Use Azure Cost Management + Billing to align cost and forecast models by creating budgets.
- Build your cloud adoption plan: Build a plan with actionable workload, asset, and timeline details.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Cloud strategy team	• Cloud governance team • Cloud operations team • Cloud center of excellence or central IT team

Step 5: Expand the landing zones

The Ready methodology of the Cloud Adoption Framework focuses heavily on the development of landing zones to host workloads in the cloud. During landing zone implementation, various decisions can affect operations.

Consult the cloud operations team to help review the landing zone for operations improvements. Also consult the cloud governance team to understand "resource consistency" policies and design guidance, which can affect the landing zone design.

Deliverables:

- Deploy one or more landing zones that can host workloads in the short-term adoption plan.
- Ensure that all landing zones meet operations decisions and resource consistency requirements.

Guidance to support deliverable completion:

- Improve landing zone operations: Best practices for improving operations within a landing zone.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Cloud adoption team	• Cloud operations team • Cloud strategy team • Cloud governance team • Cloud center of excellence or central IT team

Step 6: Adoption

Long-term operations might be affected by the decisions that you make during migration and innovation efforts. Maintaining consistent alignment early in adoption processes helps remove barriers to production release. It also reduces the effort required to onboard new solutions into operations management practices.

Deliverables:

- Test operational readiness of production deployments by using Resource Consistency policies.
- Validate adherence to design guidance for resource consistency and to operations requirements.
- Document any advanced operations requirements in the operations management workbook.

Guidance to support deliverable completion:

- Environmental readiness checklist
- Pre-promotion checklist
- Production release checklist

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Cloud adoption team	• Cloud strategy team • Cloud operations team • Cloud strategy team • Cloud governance team • Cloud center of excellence or central IT team

Value statement

The preceding steps will help you implement controls and processes to ensure performance across the enterprise and all hosted resources.

Get started: Align your organization

7/21/2021 • 3 minutes to read • Edit Online

Successful cloud adoption is the result of properly skilled people doing the appropriate types of work, in alignment with clearly defined business goals, and in a well-managed environment. To deliver an effective cloud operating model, it's important to establish appropriately staffed organizational structures. This article outlines such an approach.

Step 1: Understand the functions required for successful cloud teams

The following list outlines the minimum functionality that's required for your organization to succeed at cloud adoption and long-term operations. After you become familiar with the cloud teams and their functions, you can align them with the organizational structure that best fits your staffing and cloud maturity level.

- Cloud adoption functions deliver technical solutions.
- Cloud strategy functions align technical change with business needs.
- Cloud operations functions support and operate adopted solutions.
- Cloud center of excellence (CCoE) functions improve quality, speed, and resiliency of adoption.
- Cloud governance functions manage risk.
- Cloud platform functions operate and mature the platform.
- Cloud automation functions accelerate adoption and innovation.
- Cloud security functions manage security risks.

Step 2: Map people to the required functions

The next step is to map specific people to the necessary functions. To do so, answer the following questions:

- What person or group will be responsible for completing technical tasks in the cloud adoption plan?
- What person will be accountable for the team's ability to deliver technical changes?
- What person or group will be responsible for implementing protective governance mechanisms?
- What person will be accountable for defining those governance controls?
- Are there other functions or people that will have accountability or responsibility within the cloud adoption plan?

After you've documented the answers to these questions, see Plans for skills readiness to help define your plans to prepare these people for forthcoming work.

Step 3: Determine how teams align within your organization

The following organizational structures don't necessarily have to map to an organizational chart (org chart). Org charts generally reflect command and control management structures. Conversely, the following organizational structures are designed to capture alignment of roles and responsibilities.

In an agile matrix organization, these structures might be best represented as virtual teams. There's nothing to suggest that virtual teams couldn't be represented in an org chart, but a formal org chart isn't necessary to produce an effective operating model.

Determine how the following models fit your organizational structures:

- **Org chart alignment:** Management hierarchies, manager responsibilities, and staff alignment will align

with organizational structures.

- **Virtual teams:** Management structures and org charts remain unchanged. Instead, virtual teams will be created and tasked with the required functionality.
- **Mixed model:** More commonly, a mixture of org chart and virtual team alignment will be required to deliver on cloud transformation goals.

Step 4: Establish team structures

During every cloud adoption effort, certain functions must be provided by at least one person. These assignments and team structures can develop organically, or they can be intentionally designed to match a defined team structure.

To help create balance across cloud adoption efforts, we recommend that you start with a minimum of two teams. The following two teams are responsible for various functions throughout the adoption effort:

- **Cloud adoption team:** This team is accountable for technical solutions, business alignment, project management, and operations for the solutions that are adopted.
- **Cloud governance team:** To balance the cloud adoption team, a cloud governance team is dedicated to ensuring excellence in the solutions that are adopted. The cloud governance team is accountable for platform maturity, platform operations, governance, and automation.



This proven approach is considered a minimum viable product (MVP), because it might not be sustainable. Each team wears many hats, as outlined in the RACI (responsible, accountable, consulted, and informed) charts.

As adoption needs grow, so does the need to create balance and structure. To meet those needs, companies often follow a process of maturing their organizational structures.

Watch this video to get an overview of common team structures at various stages of organizational maturity.

Step 5: Align RACI charts

At each level of maturity, accountability for various cloud functions shifts to new teams. This shifting of accountability enables faster migration and innovation cycles by removing and automating barriers to change. To align assignments properly, the RACI alignment article shows a RACI chart for each organizational structure.

Additional information

- Adapt existing roles, skills, and processes for the cloud
- Organizational antipatterns: silos and fiefdoms
- Download the RACI template

Get started: Build a cloud strategy team

7/21/2021 • 11 minutes to read • Edit Online

To be successful, every cloud adoption journey needs to involve some level of strategic planning. This getting started guide is designed to help you establish a dedicated team or virtual team that can build and deliver on a solid cloud strategy.

The first step in the journey is to decide whether you need a strategy team, or whether your existing team members can deliver on cloud strategy as a distributed responsibility.

Whichever approach you choose, you'll want to create a cloud strategy team that defines motivations and business outcomes, and that validates and maintains alignment between business priorities and cloud adoption efforts. When the business outcomes affect business functions, the strategy team should include business leaders from across the organization. The goal of the cloud strategy team is to produce tangible business results that are enabled by cloud technologies. Overall, this team ensures that cloud adoption efforts progress in alignment with business outcomes. Whenever possible, business outcomes and the cloud strategy team's efforts should both be defined early in the process.



> **NOTE**
>
> This article discusses a *strategy facilitator*, a key player in the cloud-adoption process. The role is commonly held by a program manager, architect, or consultant. As the cloud strategy team forms and gets started, the strategy facilitator is temporarily accountable for creating alignment and keeping the team aligned with business goals. The strategy facilitator is often the person most accountable for the success of the cloud adoption journey.

Step 1: Determine whether a cloud strategy team is needed

A cloud strategy team delivers on a required capability in the cloud, referred to as the cloud strategy capability. Forming a cloud strategy team requires a defined group of dedicated business leaders, stakeholders, and program managers to meet on a regular, recurring basis to advance the strategy that drives cloud adoption.

Deliverables:

- Determine whether your business requires a cloud strategy team.

Guidance to support deliverable completion:

Creating a cloud strategy team is often necessary for the following reasons:

REASON	CONSIDERATIONS
Cloud adoption is important to the business.	• The cloud adoption effort has board-level visibility. • Success of the cloud adoption effort will improve market positioning, customer retention, or revenue. • The programs in the adoption portfolio map directly to strategic business outcomes. • The portfolio of workloads in this adoption effort is strategic and mission-critical and could affect multiple business units.
Cloud adoption requires ongoing executive support.	• The cloud adoption effort will affect how you manage organizational change. • The effort will require additional training from multiple business users and could interrupt certain business functions. • The existing IT operations team or vendor is motivated to remain in an existing datacenter. • The existing IT team hasn't fully bought into the effort.
Cloud adoption presents risk to the business.	• Failure to complete the migration within the specified time window will result in negative market impact or increased hosting costs. • Workloads slated for adoption need to be protected from data leakage that could affect business success or customer security. • Metrics that are being used to measure the cloud effort are business aligned, creating a dependency and risk on the technical success.

If any or all of the preceding reasons represent your existing business considerations, the information in the rest of this article will help you establish your cloud strategy team.

Accountable person or team:

- The strategy facilitator is accountable for determining whether a cloud strategy team is needed.

What if I don't need a cloud strategy team?

Review the cloud strategy functions that are required to deliver on cloud strategy needs. Not every organization requires a dedicated team or virtual team to help meet its strategic needs. In your RACI (responsible, accountable, consulted, and informed) template, list the core accountabilities of the strategy, and identify the person on your team who will be accountable for each. If one person will take on all of those accountabilities, simple replace "cloud strategy" with that person's name in the RACI template.

Step 2: Establish the cloud strategy team

The cloud strategy team serves as a recurring alignment point between business leaders and IT leaders. Based on the levels of importance, risk, and executive support that drive the need for a strategy team, participation in and the composition of the team might vary.

Deliverables:

- Identify the appropriate organizations or individuals who are willing to share in the accountability and responsibility for driving the cloud adoption strategy.

Guidance to support deliverable completion:

- Document and share your reasoning from step 1 to identify stakeholders who will benefit from regular

involvement and will be able to help drive the strategy.

- For ideas about who might be a good fit, see Cloud strategy functions.
- To validate the alignment and bandwidth from each potential participant, review the minimum scope and deliverable for this capability.
- To establish the right RACI chart based on you current team structures, review the various RACI configuration examples, or select one of the example tabs at the bottom of the RACI template.
- Document the results in the RACI template in the `Org Alignment` worksheet.

Accountable person or team:

- The strategy facilitator is accountable for establishing the cloud strategy team.

Step 3: Establish a cadence

Early in the cloud adoption journey, your team will require frequent interaction and iterative strategy reviews. As adoption starts, that frequency will lessen, transitioning to a focus on status and validation or adjustment of the backlog priorities.

Steps 4, 5, and 6 should be completed within four to six weeks. The remaining steps will be completed in subsequent meetings. More frequent meetings should be maintained until the team begins step 7.

Deliverables:

Review suggested meeting cadences and schedule meetings with all strategy team participants.

Guidance to support deliverable completion:

- Review the suggested short-term and long-term meeting cadences to align each of the documented participants.

Accountable person or team:

- The strategy facilitator is accountable for establishing an appropriate cadence for the cloud strategy team.

Step 4: Establish a motivation-driven strategy

Cloud adoption journeys include approaches to both migration and innovation. When technical teams define the strategy, it's common for the strategy to be driven by the team members' current skills and strengths. Such a strategy is likely to be a technical success, but it risks producing limited business impact.

The first objective of the cloud strategy team is to define a high-level strategy that's based on business motivations. This can usually be completed in a one-hour workshop with all of the cloud strategy team members. It also requires a minimum of one additional hour to review the business motivations with various technical teams and affected stakeholders.

During the first workshop, each member of the team should prioritize its motivations in the understand motivations article and share its top priorities. The strategy facilitator helps guide one or more rounds of conversation until a theme emerges in the direction of migration or innovation. There will likely be motivations in the top 3 list from both categories, which might require the team to go deeper on its list before a clear pattern leans one way or another.

This exercise will surface conversations that can help build alignment among the team members. The deliverable will help guide the rest of the strategy and the resulting plan.

Deliverables:

- Record motivations in the strategy and plan template.

Guidance to support deliverable completion:

- Understand motivations: Critical business events and some migration motivations tend to be cost sensitive, which increases the importance of cost control for all subsequent efforts. Other forward-looking motivations that are related to innovation or growth through migration might be focused more on top-line revenue. Understanding motivations helps team members decide how high to prioritize your cost management.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Cloud strategy team	• Cloud governance team • Cloud adoption team • Cloud center of excellence or central IT team

Step 5: Establish business outcomes

Each member of the cloud strategy team is asked to define one or more business outcomes by applying specific metrics to measure business success. If a certain metric can be improved as a direct result of the cloud adoption effort, a team member is asked to share the expected impact. If the cloud adoption effort doesn't affect the metric but will enable the business to better drive the metric, that too should be documented.

Many leaders on the cloud strategy team might need to decompose their core metrics to identify an outcome that can be affected or influenced by the cloud adoption effort. If the team members' outcomes can't be affected or influenced by this journey, it might be hard for them to maintain their level of interest in the program. The facilitator should work with each leader to develop an aligned metric and reevaluate whether that team member is the right person to participate on the strategy team.

Impacts on business outcomes can take time. These types of changes typically move slower than technical changes. To maintain transparency, the strategy team should agree on shorter-term learning metrics. These metrics might include technical and other changes that can be reviewed at each team meeting to demonstrate progress toward technical goals and business outcomes.

Deliverables:

- Identify at least one expected business outcome per member of the cloud strategy team.
- Refine the list of members to align expected time commitments with expected outcomes.
- Align on a set of short-term and mid-term metrics to support ongoing progress reports.

Guidance to support deliverable completion:

- Record business outcomes in the strategy and plan template.
- Business outcomes: Some fiscal outcomes tend to be extremely cost sensitive. When the desired outcomes map to fiscal metrics, it can be wise to invest very early in the Cost Management governance discipline.
- Learning metrics help bridge the gap between business outcomes and technical adoption efforts.

Accountable team:

The cloud strategy team is accountable for defining business outcomes. The team can use specific metrics to measure the success of the business outcomes.

Step 6: Decide whether to proceed or cancel based on the business justification

Your business justification can help with planning, long-term return expectations, and expectations about total cost of ownership (TCO). In this step, the cloud strategy team should agree on the minimum amount of analysis

required to help the strategy team align on a go-forward decision. Strategic alignment might require deep planning and TCO analysis. Most cloud strategy teams will find a simple cost analysis sufficient to align on direction.

Each member of the strategy team should review common myths and approaches to business justification. This can help the team communicate the specific analysis that's expected from the supporting teams. After the team communicates its expectations, it can reduce its time investment and meeting frequency. The team will maintain accountability for completing the strategy until the business justification and digital estate analysis have been agreed upon.

Deliverables:

- Kick off the business justification effort with your supporting teams.
- Meet with the supporting teams monthly (or as needed) until the strategy team can align on a go/no go decision to proceed with cloud adoption.

Guidance to support deliverable completion:

- The business justification serves as a high-level view of the overall financial plan for cloud adoption. This can be a good source for initial budgeting efforts.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Cloud strategy team	• Cloud governance team • Cloud adoption team • Cloud center of excellence or central IT team

Step 7: Support adoption through a regular cadence

After a go-forward decision has been agreed upon with the cloud strategy team, the team can transition into a less intense and less frequent meeting cadence. The expectations of the team also shift at this point. After the journey moves from strategic definition to adoption efforts (plan, ready, adopt), the strategy team is expected to focus on prioritization and strategic support.

Deliverables:

- **Prioritization:** When the existing digital estate is rationalized, the strategy team helps establish waves of migration or innovation priorities. This helps the technical implementation teams focus on actions that drive the greatest business value.
- **Evaluate risks:** As cloud adoption grows, new forms of adoption expose new risks. The strategy team is responsible for helping evaluate those new risks. The expectation of the strategy team is to evaluate new risks and determine whether the business can tolerate the risks or it needs policies that eliminate or mitigate them.
- **Review budget and spend:** As cloud adoption increases, so will budgets for various workloads in the portfolio. On a monthly basis, the cloud strategy team should review actual spend against budget to identify issues that need to be resolved. Detecting and addressing budgetary changes early will help prevent sticker shock later in the adoption lifecycle.
- **Business planning:** When the adoption teams complete their migration or innovation efforts, additional business planning will be required to maximize return from the new technology solutions. Such planning might include user training, business process modifications, or other post-adoption activities.
- **Executive support:** Cloud adoption will result in organizational change. This is most visible within the IT organization. At times, various teams or team members might need additional support from the strategy team to understand the changes, develop new skills, and understand how to best operate within the new

models.

Guidance to support deliverable completion:

- Incremental rationalization: Consider an agile approach to rationalization that properly aligns late-bound technical decisions.
- The five Rs of rationalization: Understand the various rationalization options.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Cloud strategy team	• Cloud governance team • Cloud adoption team • Cloud center of excellence or central IT team

What's next

Strategy and planning are important. Nothing is actionable until you identify the cloud adoption functions that are needed on your team. It's important to understand these key capabilities before you begin your adoption efforts.

Align your strategy with the cloud adoption functions by working with the adoption team or individuals who are responsible for these functions.

Learn to align responsibilities across teams by developing a cross-team matrix that identifies RACI parties. Download and modify the RACI template.

Get started: Build a cloud adoption team

7/21/2021 • 6 minutes to read • Edit Online

Cloud adoption teams are the modern-day equivalent of technical implementation teams or project teams. The nature of the cloud might require more fluid team structures.

Some cloud adoption teams focus exclusively on cloud migration, and others focus on innovations that take advantage of cloud technologies. Some teams include the broad technical expertise that's required to complete large adoption efforts, such as a full datacenter migration, and others have a tighter technical focus.

A smaller team might move between projects to accomplish specific goals. For example, a team of data platform specialists might focus on helping convert SQL Database virtual machines (VMs) to SQL PaaS instances.

As cloud adoption expands, customers benefit from a team that's dedicated to the cloud platform function. That team uses automated deployment and code reuse to accelerate successful adoption. People focused on a cloud platform function can implement infrastructure, application patterns, governance, and other supporting assets to drive further efficiencies and consistency, and to instill cloud principles in your organization. Small organizations and small adoption teams don't have the luxury of a dedicated cloud platform team. We recommend that you establish an automation capability in your adoption team to begin building this important cloud muscle.



Step 1: Determine the type of adoption team you need

Cloud adoption teams tend to perform one or more of the following types of adoption:

- Migration of existing workloads
- Modernization of existing workloads and assets
- Architectural change to existing workloads and assets
- Development of new workloads

The adoption of any IT portfolio will likely require a mixture of these types of efforts. Unfortunately, each type requires different skills and mindsets. The more specialized an adoption team, the more effective and efficient the team will be when it delivers that type of work. Conversely, mastery of all the implementation options across cloud adoption can be overwhelming for these more specialized teams.

When you're first building a cloud adoption team, aligning with one of the Adopt methodologies will help accelerate the development of the team's collective skills.

Deliverables:

- Determine whether the team aligns better with the Migrate methodology or the Innovate methodology.

- Each methodology has a four-step onboarding experience to help the team understand the tools and processes required to get really good at that effort. Invest time as a team going through the first few steps to understand which tools and scenarios you're most likely to need in early iterations.
- Align responsibilities across teams by developing a cross-team matrix that identifies *responsible, accountable, consulted, and informed (RACI)* parties. Update your company's RACI template to help others understand who's on the team and which methodology the team will focus on delivering.

Guidance to support deliverable completion:

- Migrate methodology overview describes the process, tools, and approaches for migrating and modernizing a portfolio of workloads.
- Innovate methodology overview describes the process, tools, and approaches for adding cloud-native workloads to the portfolio.
- Understand motivations behind this effort to see whether they're better aligned with migration or innovation efforts.

Step 2: Align your team with other supporting teams

If your company's cloud adoption effort is mature enough to have supporting teams, you might be able to find a list of the teams and subject matter experts in your company's version of the RACI template, including cloud governance, cloud operations, a cloud center of excellence, or other similar teams.

Deliverables:

- Review design guidance, operational baselines, policies, and processes from the various supporting teams to understand the guardrails that have been established for guiding cloud adoption.
- Review the guidance with other cloud adoption teams to understand any limitations you might encounter as a result of those guardrails.

Guidance to support deliverable completion:

- Evaluate corporate policy outlines the steps to define corporate policy, which might limit decisions that the team can safely make in the company's cloud environment.
- Governance disciplines outlines the types of controls or disciplined processes that the governance team has likely implemented to allow for safe, compliant adoption of the cloud.
- The Manage methodology outlines the considerations that go into a cloud operations baseline for providing basic operations management.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• The cloud strategy team is accountable for maintaining a clear RACI structure across the cloud adoption lifecycle.	Review guidance and requirements from: • Cloud governance team • Cloud operations team • Cloud center of excellence or central IT team • Other cloud adoption teams or individuals listed in the RACI

Step 3: Begin your adoption journey

Depending on the type of adoption team you're a member of, you'll get started with one of these guides:

- Get started: Migrate workloads to the cloud
- Get started: Build new products or services

These guides provide guidance for various teams listed alongside their varying degrees of accountability and responsibility. Use the guides to understand how your team fits into the rest of the journey. Also use them to understand the levels of support you can expect to get from around the company.

In the end, the cloud adoption team is accountable for delivery across their assigned migration efforts or new product development. Although supporting teams are accountable for ensuring that each step is completed, it's the responsibility of each cloud adoption team to ensure that the supporting team is getting the support it needs to be successful. If the accountable team doesn't yet exist or needs more support to deliver on its accountable steps, the adoption team is encouraged to partner with other teams to complete its deliverables.

Deliverables:

- Become increasingly better at delivering on the methodology associated with your adoption approach.
- Support other teams in the completion of their accountable steps, even if those steps are blockers to your adoption efforts.

Guidance to support deliverable completion:

- In the getting started guide for migration, the adoption team is accountable for delivery of step 10: Migrate your first workload.
- In the getting started guide for new products, the adoption team is accountable for delivery of step 8: Innovate in the cloud.

All other steps on those checklists are designed to make the effort more manageable.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Cloud adoption team	• Cloud governance team • Cloud operations team • Cloud center of excellence or central IT team • Cloud strategy team

Step 4: Expand your skills with scenarios and best practices

After one or two iterations, the cloud adoption team will understand the basics of their primary methodology. From there, the team will likely be ready to take on additional scenarios and start implementing some additional best practices.

Deliverables:

- Increase skills and experience to address more complex adoption scenarios.

Guidance to support deliverable completion:

The team can review and expand their skills by reviewing the following guidance:

- Migrate new types of workloads or solve more complex migration challenges through scenarios and best practices.
- Innovate by using new cloud-native solutions, or solve more complex innovation challenges through scenarios and best practices.

Accountable team:

- The cloud adoption team is accountable for expanding its skills.

Step 5: Build a cloud adoption factory

As the team becomes more familiar with various adoption scenarios, it will be able to do more and do it faster. This section of guidance will take the team's adoption abilities to the next level.

The cloud adoption factory approach looks at the processes behind adoption efforts. Because of a lack of understanding and clear communication, most of the time burden that's related to migration and innovation comes from a high volume of meetings. Clearly defining processes and interactions at various phases of the cloud adoption journey will remove cultural and political blockers.

Deliverables:

- Improve delivery processes to create a highly optimized adoption factory.

Guidance to support deliverable completion:

- Process guidance that supports migration efforts can be found in the process improvements section of the Migrate methodology.
- In the Innovate methodology, the guidance focuses on innovation processes that result in less technology and more effective product development.

Accountable team:

- The cloud adoption team is accountable for building the processes that take adoption to the next level.

What's next

Cloud adoption is a great goal, but ungoverned adoption can produce unexpected results. To accelerate adoption and best practices, as you're reducing business and technical risks, align cloud adoption with cloud governance functions.

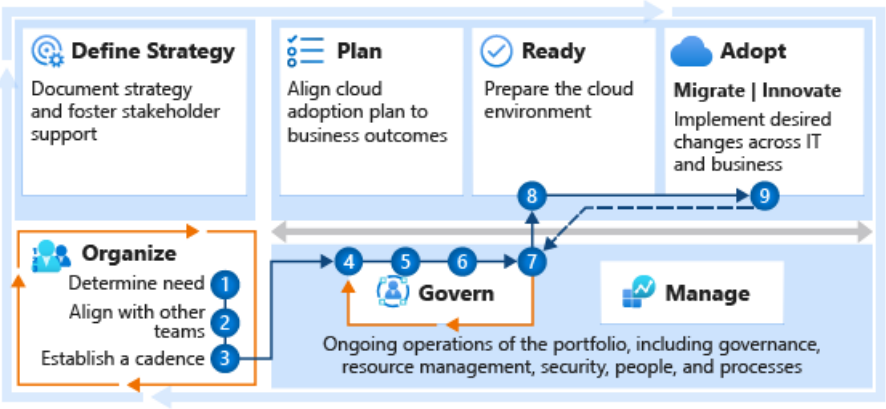
Aligning with the cloud governance team creates balance across cloud adoption efforts, but this is considered a minimum viable product (MVP), because it might not be sustainable. Each team is wearing many hats, as outlined in the RACI charts.

Learn more about overcoming organizational antipatterns: silos and fiefdoms.

Get started: Build a cloud governance team

7/21/2021 • 6 minutes to read • Edit Online

A cloud governance team ensures that cloud-adoption risks and risk tolerance are properly evaluated and managed. The team identifies risks that can't be tolerated by the business, and it converts risks into governing corporate policies.



Step 1: Determine whether a cloud governance team is needed

The official guidance in the Cloud Adoption Framework is to always create a cloud governance team. At first, the team might be very small. Regardless of its size, its role is important. If a team isn't needed, a group or individual on an existing adoption team should agree to fulfill the responsibilities associated with cloud governance functions.

Deliverables:

- Determine whether you need a cloud governance team.
- Align responsibilities across teams by developing a cross-team matrix that identifies *responsible, accountable, consulted, and informed (RACI)* parties. Document the decision and the responsible individuals using the RACI template in the `Org Alignment` worksheet.

Guidance to support deliverable completion:

- Cloud governance functions might already be spread across multiple individuals or teams. Having a team that goes by the title "cloud governance team" isn't important, but the required capabilities should reside with an accountable party or team.
- If the company's long-term cloud adoption strategy can be delivered from one landing zone in one cloud environment, the amount of governance and operations efforts might be small enough for delivery by one person or one team. That team is unlikely to be called cloud governance, because it serves many functions beyond cloud governance. Even for that team, this getting started guide can help ensure that it can deliver on this important function of governance.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Cloud strategy team	• Cloud adoption team

Step 2: Align with other teams

The governance team ensures consistency and adherence to a set of common policies. Those policies come from an ongoing alignment with other teams.

Before it establishes policies or automated cloud governance, the cloud governance team should meet with other teams that are identified in the RACI template. This will help ensure alignment on critical topics, such as security, cost, performance, operations, and deployment. Steps 4 and 5 can help facilitate the alignment.

Deliverables:

- Discuss current-state implementation and ongoing adoption plans with each team.

Guidance to support deliverable completion:

- Review your company's strategy and plan template with members of the cloud strategy team to understand motivations, metrics, and strategy.
- Review your company's cloud adoption plan with members of the cloud adoption team to understand timelines and prioritization.
- Review the operation team's operations management workbook to understand the operational requirements and commitments that have been established with the business.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Cloud governance team	• Cloud strategy team • Cloud adoption team • Cloud operations team • Cloud center of excellence or central IT team

Step 3: Establish a cadence with other teams

Cloud adoption generally comes in waves, or releases. A regular cadence that's aligned with those releases lets the cloud governance team look ahead and understand the risks that will be introduced in the next wave. Staying engaged with the strategy, adoption, and operations teams during planning and review also helps the governance team stay ahead of coming risks.

Deliverables:

- Establish a cadence with the supporting teams. If possible, align that cadence with release and planning cycles.
- Establish a separate cadence directly with the cloud strategy team (or various team members) to review risks that are associated with the next wave of adoption and gauge the team's level of tolerance for those risks.

Guidance to support deliverable completion:

- For more guidance on cadences for meetings, see Cloud governance functions.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Cloud governance team	• Cloud strategy team • Cloud adoption team • Cloud operations team

Step 4: Review the methodology

To help establish a future vision for governance and a working approach to that vision, review the Govern methodology of the Cloud Adoption Framework.

Deliverables:

- Gain an understanding of the methodology, approach, and implementation that supports the Govern methodology.

Guidance to support deliverable completion:

- Review the Govern methodology.

Accountable team:

- The cloud governance team is accountable for establishing a vision and approach to governance.

Step 5: Complete the governance benchmark

Governance is a broad topic. A short assessment can help the team understand where to get started.

Deliverables:

- Complete the governance benchmark assessment, based on conversations with various stakeholders. Or ask other teams to complete the assessment on their own.

Guidance to support deliverable completion:

- Use the governance benchmark to assess your governance needs and priorities.

Accountable team:

- The cloud governance team should understand the gaps that are identified in the governance benchmark and then provide direction on governance that helps address the gaps.

Step 6: Implement the initial governance best practice and configuration

The Govern methodology includes two approaches to an initial governance foundation. Review each approach, and implement the one that most closely matches your needs.

Deliverables:

- Deploy the basic governance tools and organization configurations that are required to govern the environment during the next few waves of adoption efforts.

Guidance to support deliverable completion:

- For guidance on configuration and implementation, review establish an initial cloud governance foundation.

Accountable team:

- The cloud governance team is accountable for the review and implementation of governance best practices and an initial governance foundation.

Step 7: Continuously improve governance maturity

Governance needs grow as additional cloud adoption efforts are completed. Stay aligned with the ongoing adoption plan to ensure that the governance approach can maintain the proper levels of governance and control.

Deliverables:

- Implement governance improvements to guard against changing risks and governance needs.

Guidance to support deliverable completion:

- To help improve the initial governance foundation, implement expanded governance scenarios.

Accountable team:

- The cloud governance team is accountable for aligning with ongoing adoption plans.

Step 8: Evaluate landing zone changes

As landing zones are deployed and expanded, new risks or governance violations might emerge. Periodically review landing zone configurations to identify any deviations from policy that aren't caught by the cloud-native governance tools. Ensure that each landing zone deployment adheres to guidelines for landing zone governance.

Deliverables:

- Help the cloud platform team develop improvements to the landing zone, which must comply with governance policies.

Guidance to support deliverable completion:

- Improve landing zone governance.

Accountable team:

- The cloud governance team should make sure that each landing zone deployment adheres to governance guidelines.

Step 9: Adoption handoffs

As new adoption efforts are completed, the cloud adoption team hands off operational responsibilities to the cloud operations team and cloud governance teams. Stay aligned with adoption release cadences to ensure proper documentation and policy alignment, and to help the team assume responsibility for the workloads.

Deliverables:

- Regularly review and accept handoffs from other cloud adoption teams.

Guidance to support deliverable completion:

- Establish a process for onboarding new workloads and resources.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Cloud adoption teams	• Cloud governance team • Cloud operations team

What's next

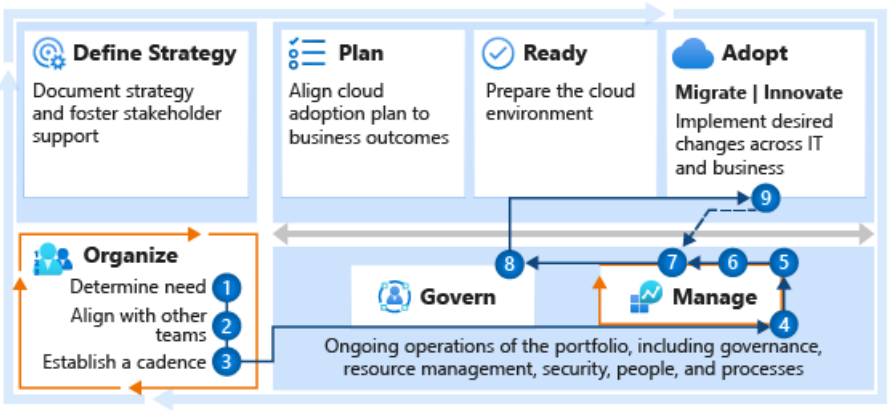
All companies are unique, and so are their governance needs. Choose the level of maturity that fits your organization, and use the Cloud Adoption Framework to guide the practices, processes, and tooling that can help you get there.

As cloud governance matures, teams are empowered to adopt the cloud at a faster pace. Continuous cloud adoption efforts tend to trigger maturity in IT operations. To ensure that governance is a part of operations development, either develop a cloud operations team or sync with your existing cloud operations team.

Get started: Build a cloud operations team

7/21/2021 • 6 minutes to read • Edit Online

An operations team focuses on monitoring, repairing, and remediating issues related to traditional IT operations and assets. In the cloud, many of the capital costs and operations activities are transferred to the cloud provider, giving IT operations the opportunity to improve and provide significant additional value.



Step 1: Determine whether a cloud operations team is needed

Before you can release any workloads into production, an agreement must be reached on the accountability for delivery of cloud operations functions. For some portfolios, operational responsibilities might belong to the DevOps and cloud adoption teams. In other cases, a managed service provider with cloud operations experience might assume ongoing operational duties.

If no DevOps or service-provider operations agreements are in place, it's safe to assume that someone within IT will need to commit to ongoing operational duties regarding the management of production workloads.

Deliverables:

- Determine whether you need a cloud operations team.
- Align responsibilities across teams by developing a cross-team matrix that identifies *responsible, accountable, consulted, and informed (RACI)* parties. Document the decision and responsible individuals in the RACI template in the `Org Alignment` worksheet.

Guidance to support deliverable completion:

- Cloud operations functions might be spread across multiple individuals or teams already. Decide whether a cloud operations team is required. Some level of operations is always needed for production workloads.
- If the company's long-term cloud adoption strategy can be delivered from one landing zone in one cloud environment, the governance and operations efforts might be small enough to be delivered by one person or one team. That team is unlikely to be called cloud operations, because it will serve many functions. For that individual or team, the following guidance can help ensure that it can deliver on this important function of operations.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Cloud strategy team	• Cloud adoption team • Cloud governance team

Step 2: Align with other teams

The cloud operations team inherits operational responsibilities for all workloads in the production portfolio. Those responsibilities can vary between workloads, based on expectations and the commitments the team has made to business stakeholders. The architectural decisions made by migration-focused and innovation-focused cloud adoption teams also influence the team's operational commitments.

Before the cloud operations team implements any ongoing operations practices, it's important for it to align with other teams. The team should meet with other teams that are identified in the RACI template to ensure alignment on critical topics, such as security, cost, performance, governance, adoption, and deployment. Steps 4 and 5 can help facilitate this alignment.

Deliverables:

- Discuss current-state implementation and ongoing adoption plans with each team.

Guidance to support deliverable completion:

- To understand team motivations, metrics, and strategy, review your company's strategy and plan template with members of the cloud strategy team.
- To understand timelines and prioritization, review your company's cloud adoption plan with members of the cloud adoption team.
- To understand the operational requirements and commitments that the team has established with the business, begin developing the operations management workbook.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Cloud operations team	• Cloud strategy team • Cloud adoption team • Cloud governance team • Cloud center of excellence or central IT team

Step 3: Establish a cadence with other teams

Cloud adoption generally comes in waves, or releases. A regular cadence that's aligned with those releases lets the cloud operations team prepare for the handoffs at the end of the next wave. Staying engaged with the strategy, adoption, and governance teams during planning and review helps the operations team stay ahead of the coming operational demands.

Deliverables:

- Establish a cadence with the supporting teams. If possible, align that cadence with release and planning cycles.
- Establish a separate cadence directly with the cloud strategy team or its various team members to review any operational requirements that are associated with the next wave of adoption.

Guidance to support deliverable completion:

- For additional guidance on cadences for meetings, see the "deliverables" section of cloud operations functions.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Cloud operations team	• Cloud strategy team • Cloud adoption team • Cloud governance team

Step 4: Review the methodology

To help establish a future vision for operations management and a working approach to achieve that vision, review the Manage methodology of the Cloud Adoption Framework.

Deliverables:

- Gain an understanding of the methodology, approach, and implementation that supports the Manage methodology.

Guidance to support deliverable completion:

- Review the Manage methodology of the Cloud Adoption Framework.

Accountable team:

- The cloud operations team is accountable for the vision and approach to operations management.

Step 5: Implement the operations baseline

If operations practices aren't already deployed to your cloud environments, start with the operations baseline. That baseline will implement cloud-native, no-ops/low-ops practices to provide a base level of operational protection.

Deliverables:

- Deploy the basic Azure server-management configurations that are required for operating the environment during the next few waves of adoption efforts.

Guidance to support deliverable completion:

- Implement the operations baseline configuration.

Accountable team:

- The cloud operations team is accountable for implementing the operations baseline.

Step 6: Align business commitments

Review the team's operations baseline commitments with the business stakeholders. This baseline helps you evaluate the general requirements for the majority of workloads. The process also helps you identify the stakeholders for various workloads and enables you to document their ongoing operational expectations.

Deliverables:

- Document the expectations of business stakeholders.
- Determine whether advanced operations are required for specific workloads or platforms.

Guidance to support deliverable completion:

- Create business alignment in the cloud.
- Document the portfolio and operations expectations in the operations management workbook.

Accountable team:

- The cloud operations team should understand the business expectations, and it's accountable for ongoing alignment with those expectations.

Step 7: Operations maturity

By continually making operational improvements, the team can:

- Enhance the operations baseline.
- Improve platform operations.
- Implement workload-specific operations.

As additional workloads are transitioned to cloud operations, the need for operations improvements become clearer.

Deliverables:

- Improve operations maturity to support commitments to business stakeholders.

Guidance to support deliverable completion:

- Evaluate the best options for advanced operations management.

Accountable team:

- The cloud operations team is accountable for operational improvements and maturity over time.

Step 8: Scale operations consistency through governance

As operations planning continues to mature, the team should coordinate with the cloud governance team regularly to apply operations requirements across the portfolio.

Deliverables:

- Help the cloud governance team implement new requirements for resource consistency.

Guidance to support deliverable completion:

- Review the governance guide for improving resource consistency.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Cloud governance team	• Cloud operations team

Step 9: Adoption handoffs

As new adoption efforts are completed, the cloud adoption team hands off operational responsibilities to the cloud operations and cloud governance teams. To ensure proper documentation and policy alignment, and to assume responsibility for the workloads, the team should stay aligned with adoption releases.

Deliverables:

- Regularly review and accept handoffs from cloud adoption teams.

Guidance to support deliverable completion:

- Establish a process for onboarding new workloads and resources.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Cloud adoption teams	• Cloud governance team • Cloud operations team

What's next

As adoption and operations scale, it's important to define and automate governance best practices that extend existing IT requirements. Forming a cloud center of excellence (CCoE) team is an important step toward scaling cloud adoption, cloud operations, and cloud governance efforts.

Learn more about:

- Cloud center of excellence functions
- Organizational antipatterns: silos and fiefdoms

Align responsibilities across teams by developing a cross-team matrix that identifies RACI parties. Download and modify the RACI template.

Introduction to hybrid and multicloud

8/9/2021 • 6 minutes to read • Edit Online

Microsoft Azure provides all of the products and features required to help you build and operate your technology solutions in the cloud. We also understand that there are sound business reasons that may drive the necessity of using multiple private and/or public clouds. As a first step in your hybrid and multicloud journey, this article outlines and expands on Microsoft's unique perspective on important cloud computing terms.

Defining hybrid and multicloud

A hybrid cloud is a type of cloud computing that combines a private cloud (on-premises infrastructure), with a public cloud (computing services offered by third-party providers over the public internet). Hybrid clouds allow data and applications to consistently move between the two cloud environments. Many organizations choose a hybrid cloud strategy because of business requirements, such as meeting regulatory and data sovereignty requirements, maximizing on-premises technology investments, or addressing latency issues.

The hybrid cloud is evolving to include edge workloads. Cloud-managed edge computing devices bring the computing power of the public cloud to the private cloud, closer to where the IoT devices reside, including data residing in applications, connected devices, and mobile consumer services. Reducing latency by moving workloads to the edge, devices spend less time communicating with the cloud, and can operate reliably in extended offline periods. Expanded compute, storage, and service availability provides experience-driven resources closer to your customers.

Multicloud computing refers to the use of multiple cloud computing services from more than one cloud provider (including private and public clouds), in a heterogeneous environment. A multicloud strategy provides greater flexibility, and mitigates risk. Choose services from different cloud providers that are best suited for a specific task, or take advantage of services offered by a particular cloud provider in a specific location.

Hybrid and multicloud narrative

This scenario follows a common hybrid and multicloud narrative, and provides guidance on what you can do differently to be successful during your organization's cloud adoption effort. This general narrative is not restricted to a single cloud adoption methodology, but takes a view of the entire cloud adoption journey.

A hybrid cloud platform gives your organization many advantages: greater flexibility, control, and scalability, with more deployment options, global scale, integrated cross-platform security, unified compliance, and improved workload, operational, and cost efficiencies across the enterprise, **consistently achieving more value from existing infrastructure**. When computing and processing demand fluctuates, hybrid cloud computing enables you to seamlessly scale up your on-premises infrastructure to the public cloud to handle any overflow, without giving third-party datacenters access to the entirety of your data. By running certain workloads in the cloud, your organization gains the flexibility and innovation the public cloud provides, while retaining highly sensitive data in your own datacenter to meet client needs, or remain in compliance with regulatory requirements.

This allows you to scale computing resources, while modernizing and protecting mission-critical applications and data. Eliminate the need to make massive capital expenditures to meet short-term spikes in demand, or being forced to free up local resources for more sensitive data. With cloud billing models, your organization will only pay for resources you temporarily use, instead of having to purchase, program, and maintain additional resources and equipment that could remain idle for long periods.

Another capital expenditure that could be eliminated is in offsite disaster recovery and backup infrastructure

investments. Public cloud for BCDR strategies is a compelling option for those on-premises workloads and associated data not restricted in some way from residing in a public cloud. By using public cloud for BCDR customers take advantage of the major investments in privacy and security, scale on demand and ease and speed of recovery.

Companies are spreading resources across on-premises, multiple clouds, and the edge. Customers have four common needs that we often hear about:

1. Visibility into the health of all existing and future infrastructure and applications in a single pane of glass.
2. Difficulty integrating on-premises policies and updates with cloud infrastructure. Organizations understand the need for implementing a governance standard,
3. A wide range of skills across on-premises and cloud, because there are often different application development teams in the organization. Customers are looking for consistent interoperability between the two so they can unify development practices.
4. Desire to manage security posture, without heavily modifying current operations. Cloud and multicloud compounds this challenge, which can decrease trust and increase apprehension.

Consider the deployment of cloud-native services in a hybrid and multicloud environment. Cloud services are often strictly contextualized as simply "moving data and applications to the public cloud". A hybrid strategy fully supports customer operations that preclude the use of the public cloud for some workloads, such as highly regulated industries like government infrastructure, healthcare, and financial services. Depending on geography and data sovereignty regulations, internal and customer data may be required to remain within the boundaries of on-premises datacenters. Data latency sensitivity requires compute to be close to source data in on-premises datacenters, and internet connectivity disruptions are expected, or have critical implications. In these scenarios, hybrid solutions that bring cloud services, decreased management overhead (maintaining these services on-premises), and a pay-as-you-go cloud billing model can be deployed in on-premises datacenters.

Hybrid and multicloud motivations

As a true enterprise-grade cloud provider, Azure supports your business objectives across public, hybrid, and multicloud environments. This series will discuss different best practices that can help facilitate various cloud mixes ranging from 100% Azure environments to environments that have little, or no, Azure infrastructure in place.

We recognize that there are many valid reasons for customers to choose to distribute their digital estate across hybrid and multicloud environments. Here are some common business drivers:

- Minimize or avoid single cloud provider lock-in
- Business units, subsidiaries, or acquired companies have already adopted different cloud platforms
- Different cloud providers may have regulatory and data sovereignty requirements in different countries
- Improve business continuity and disaster recovery by duplicating workloads across two cloud providers
- Maximize performance by running applications close to user locations, which may require hybrid or multicloud adoption
- Enable easy migration for some data platforms or industry-specific applications by adopting multicloud strategies

Hybrid and multicloud concerns

Some of the motivations listed above can become business transformations with a sound hybrid and multicloud adoption strategy.

Others require significant effort predeployment and post-deployment efforts to realize those innovative benefits. Cloud provider lock-in, for instance, is possible. But to avoid lock-in, organizations are required to limit their vision for cloud adoption. Many of the most beneficial products and features in a cloud provider are not

portable to other cloud providers. To achieve portability and minimize lock-in, organizations are often required to limit cloud adoption to basic infrastructure as a service (IaaS) capabilities, or invest heavily in the use of cloud-native technologies like containers or Kubernetes.

After workloads are released and are in production, another common concern associated with hybrid and multicloud adoption surfaces: when organizations attempt to provide operations management support to workloads in new environments, they often have to quickly rethink their practices. Existing operations management platforms (including existing operations management policies and processes), were not built for these types of environments. To account for deviations in cloud environments, companies often end up with disparate operations tooling and operations practices, which multiplies the cost of operations by the number of cloud environments supported.

Next step: Minimize hybrid and multicloud concerns with unified operations

Understand the concept of unified operations before starting your hybrid and multicloud journey; consistent operations practices across all of your cloud environments with a common control plane can help to address many concerns regarding hybrid and multicloud strategies.

Determine whether you need to duplicate operations for each cloud provider or implement a unified operations approach to cloud management before proceeding with hybrid and multicloud adoption at scale.

Introduction to the modern application platform scenario

8/9/2021 • 4 minutes to read • Edit Online

As customers address larger, more sophisticated forms of cloud adoption, their journey to the cloud becomes more complex. Commonly, customers use orchestrated containers to manage pools of workloads in one or more centralized clusters. This article series combines technical and non-technical considerations required to prepare for Kubernetes and container integration into centralized operations and your broader cloud strategy.

Organizations include managed services, application services, and containers in their overall strategy to accelerate developer productivity, reduce operating overhead, make workloads more portable, and modernize legacy workloads.

- **Application platform:** A collection of application dependencies designed to support the execution of the application. Application platforms accelerate development by providing well-defined structures to address many common needs. Application platforms can also create constraints that affect how the application can be built.
- **Application services (PaaS services):** Application services, or platform as a service (PaaS) options for developers, provide an application platform that maximizes the value each application can draw from a cloud environment. These services create a layer of abstraction between the application platform and the underlying cloud infrastructure. This layer of abstraction forces a specific and consistently defined set of application platform constraints, which can only run in specific environments.
- **Containers:** Containers create a similar layer of abstraction between applications runtime requirements and underlying operating system & infrastructure. Unlike PaaS options, containers allow the application runtime to be configured for the needs of the application, regardless of the container host.
- **Container orchestration:** A container orchestrator provides a consistent application runtime for an instance of a container host. This orchestration allows workload-focused operations teams to mature deployment and DevOps practices to facilitate multiple deployment and operations practices, reducing dependencies on centralized or human operations.

This article series will outline how application services and containers can be integrated into your cloud adoption strategy to delivery the following outcomes:

- **Developer acceleration through abstraction:** Both containers and application services, accelerate developer productivity by allowing developers to focus more on code and less on host environment concerns.
- **Reduce operations costs through abstraction:** Standardized container orchestration ensures consistent runtimes across all hosts, which streamlines operations regardless of the workloads developers may deploy. Through standardization, centralized operations teams can extend traditional support and operations practices to container hosts.
- **Workload portability through abstraction:** Moving workloads between container orchestrators allows for workload portability. Some container orchestration platforms work in the public cloud. Other container orchestration platforms are designed for edge or private cloud deployment. Allowing for diverse container orchestration options allows for portability of workloads between container hosts across hybrid, multicloud, edge, and public cloud platforms.
- **Modernize legacy workloads through abstraction:** Legacy workloads may require a layer of abstraction prior to migration or modernization. Customizing the runtime on a container host allows the legacy requirements to be met in a modern cloud environment.

This article series outlines how you can integrate container and container management into your strategy, plan, adoption, and operation phases of your cloud journey.

Components of the scenario

This scenario is designed to guide the end-to-end customer journey, throughout the cloud adoption lifecycle. Completing the journey requires a few key guidance sets:

- **Cloud Adoption Framework:** These articles walk through the smallest set of considerations and implementations of each CAF methodology. Use these articles to prepare decision makers, central IT, and the cloud center of excellence for adoption of containers and container management as a central part of your technology strategy.
- **Azure Well-Architected Framework:** These articles outline the considerations that each workload owner should make when their workloads need to be deployed using containers or container management solutions like Kubernetes.
- **Reference architectures:** These reference solutions aid in accelerating deployment of container solutions using Azure Kubernetes Service (AKS).
- **Featured Azure products:** Learn more about the products that support your container and container management strategy in Azure.
- **Microsoft Learn modules:** Gain the hands-on skills required to implement, maintain, and support container and AKS solutions.

Common customer journeys

AKS reference architectures: The reference architectures listed in the left pane demonstrate how to deploy various proven architectures to manage your container and Kubernetes platforms with the help of Azure Kubernetes Service (AKS). These architectures are the suggested starting point for Kubernetes in Azure.

Migrate existing workloads to AKS: A common use case for AKS in Azure is to modernize existing web-based workloads directly to a container-based or cloud-native solution, instead of traditional migration efforts. The article on migrating to containers will demonstrate how Azure Migrate can accelerate container migration within your standard migration processes.

Centralize deployment and management of containers: The first set of articles in the left pane provides rich guidance on centralization of your container strategy. This article series intends to help central IT or cloud center of excellence teams understand how containers affect your cloud strategy and how to provide consistent centralized support.

Prepare for governance and operation of containers at scale: Enterprise-scale for AKS demonstrates how you can use enterprise-scale landing zones to ensure consistent governance, security, and operations across multiple landing zones for centralized management of containers at scale.

Implement specific Azure products: Accelerate and improve container and Kubernetes capabilities using different kinds of Azure products outlined in the featured products section.

Next step: Integrate modern application platforms into your cloud adoption journey

The following list of articles will take you to guidance at specific points in the cloud adoption journey to help you be successful in the cloud adoption scenario.

- Strategy for modern application platforms
- Plan for modern application platforms
- Review your environment or Azure landing zones

- Migrate workloads to modern application platforms
- Innovate using modern application platform solutions
- Govern modern application platform solutions
- Manage modern application platform solutions

Introduction to an SAP adoption scenario

8/9/2021 • 3 minutes to read • Edit Online

This article series outlines the process for integrating an SAP platform into your cloud adoption efforts.

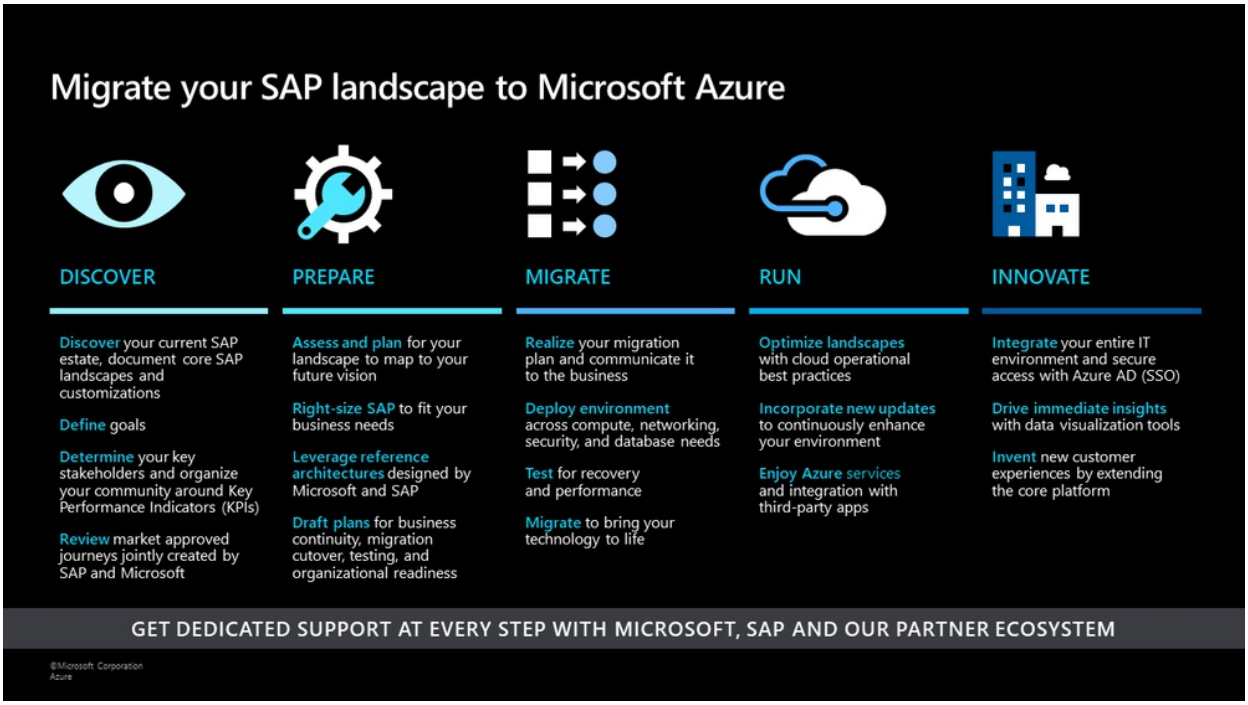
Executive summary of SAP on Azure

These articles describe how SAP workloads impact your overall strategy, cloud adoption plan, and environmental readiness efforts, with detailed guidance on common drift for each effort. Once an environment is established, the series explains the processes for migrating an SAP platform and how to use cloud technologies to innovate on that platform. To support your cloud adoption needs, the series also outlines considerations and best practices for managing governance and operations throughout an SAP deployment.

To accelerate these efforts, the articles also include detailed technical resources that describe how to build an enterprise-scale landing zone that can support your mission-critical SAP needs.

How to align to the SAP migration framework

The SAP on Azure team has produced a comprehensive framework for migrating and innovating with SAP on Azure. That framework is the main guide for organizations that focus on delivering the most successful SAP migration to Azure. If your highest priority is successfully migrating SAP to the cloud, then continue to use that guide.



This article series outlines an SAP scenario for the Cloud Adoption Framework, and it complements the current SAP migration framework. These articles are also designed for a slightly different audience with slightly different goals, and they can help your organization if you're integrating the migration of SAP into your overall cloud adoption plan. As organizations migrate to the cloud, they typically need to migrate, innovate, and manage a range of workloads and technology platforms to accomplish their business goals. Those efforts align with the methodologies in the Cloud Adoption Framework for consistent processes and approaches across various technology platforms and workloads.

If your team is following the guidance in both frameworks, you'll see very similar guidance just in packaging

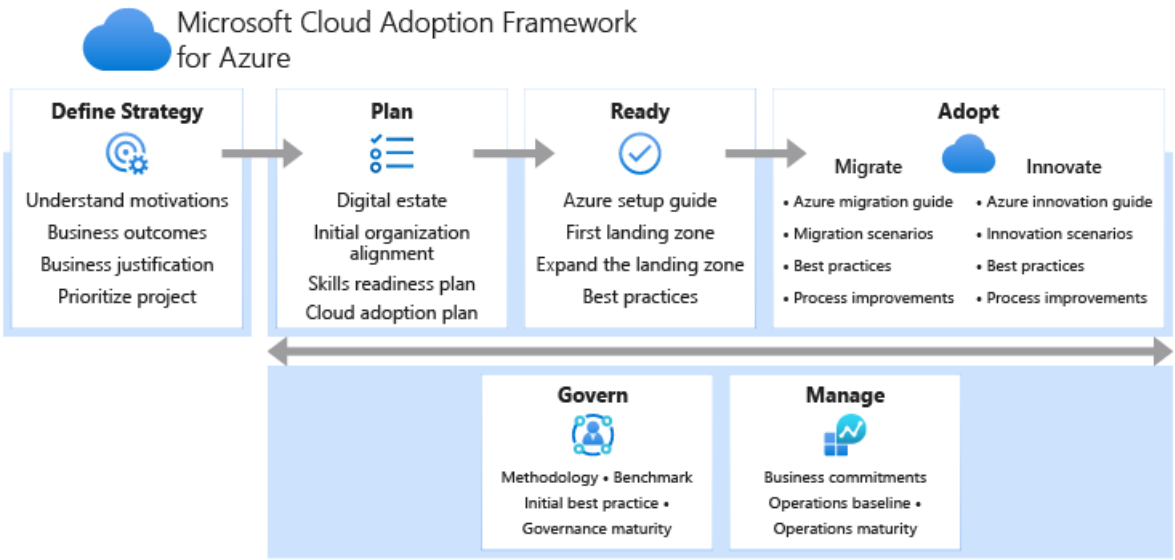
that better aligns with the audience and their objectives. Below is a list of terms for the methodologies that can help both audiences to have a similar conversation:

Discover	When cloud adoption spans multiple technology platforms, it's helpful to separate the discovery process into two conversations: `Strategy`, to engage business executives during strategic alignment, and `Plan`, to prepare the plan based on current and future-state data.
Prepare	The Ready methodology methodology prepares your team and the environment for the coming work. This scenario also integrates Azure landing zones to provide prebuilt and opinionated approaches to help your team to rapidly deploy environments that can support various technology platforms.
Migrate	The Migrate methodology demonstrates how an SAP migration can integrate with other repeatable migration processes.
Run	The Manage methodology shows how a common operations baseline can address many of the run-state concerns addressed in Migrate and also meet the operational needs of other technology platforms.
Innovate	The Innovate methodology outlines how you can take your SAP platform to the next level and integrate cloud-native solutions into your workloads.

The guidance aligns these sources to help your SAP and central IT teams work together during all phases of cloud adoption.

The SAP adoption process

Adopting SAP workloads in the cloud incorporates and addresses most of the methodologies and phases within the Cloud Adoption Framework. Distinct constraints within each phase will require actions specific to SAP, and this article series maps standard processes to SAP-specific tasks.



Next steps

The following articles provide guidance for specific points throughout the cloud adoption journey to help you succeed in adopting SAP in Azure.

- Strategic impact of SAP in the cloud
- Plan for SAP cloud adoption in Azure
- Review your environment or Azure landing zones
- Migrate an SAP platform to Azure
- Innovate with SAP
- Manage SAP

Migrate or deploy Azure Virtual Desktop instances to Azure

8/9/2021 • 2 minutes to read • Edit Online

Migrating an organization's end-user desktops to the cloud is a common scenario in cloud migrations. Doing so helps improve employee productivity and accelerate the migration of various workloads to support the organization's user experience.

Components of the scenario

This scenario is designed to guide the end-to-end customer journey, throughout the cloud adoption lifecycle. Completing the journey requires a few key guidance sets:

- **Cloud Adoption Framework:** These articles walk through the considerations and recommendations of each CAF methodology. Use these articles to prepare decision makers, central IT, and the cloud center of excellence for adoption of Azure Virtual Desktop as a central part of your technology strategy.
- **Reference architectures:** These reference solutions aid in accelerating deployment of Azure Virtual Desktop.
- **Featured Azure products:** Learn more about the products that support your virtual desktop strategy in Azure.
- **Microsoft Learn modules:** Gain the hands-on skills required to implement, maintain, and support a virtual desktop environment.

Common customer journeys

- **Azure Virtual Desktop reference architecture:** The reference architecture listed in the left pane demonstrates how to deploy a proven architecture for Azure Virtual Desktop in your environment. This architecture is a suggested starting point for Azure Virtual Desktop.
- **Migrate existing virtual desktops to Azure:** A common use case for Azure Virtual Desktop is to modernize and existing virtual desktop environment. While the process can vary, there are several components to a successful migration, such as session hosts, user profiles, images and applications. If you're migrating existing VMs, articles on migration outline how tools such as Movere and Azure Migrate can accelerate the migration as part of a standard migration process. However your migration may consist of bringing your golden image into Azure and provisioning a new Azure Virtual Desktop host pool with new session hosts. Additionally you may migrate your existing user profiles into Azure and build new host pools and session hosts as well. A final migration scenario may also include migrating your applications into MSIX app attach format. In all these migration scenarios, customers need to provision a new host pool since there is currently no direct migration of other VDI solutions into Azure Virtual Desktop.
- **Prepare for governance and operations at scale:** Enterprise-scale support for Azure Virtual Desktop demonstrates how you can use enterprise-scale landing zones to ensure consistent governance, security, and operations across multiple landing zones for centralized management of virtual desktop environments.
- **Implement specific Azure products:** Accelerate and improve virtual desktop capabilities using different kinds of Azure products outlined in the featured products section.

Next steps

The following list of articles will take you to guidance at specific points in the cloud adoption journey to help you be successful in the cloud adoption scenario.

- Strategy for Azure Virtual Desktop
- Plan for Azure Virtual Desktop
- Migrate to Azure Virtual Desktop
- Manage an Azure Virtual Desktop environment
- Govern an Azure Virtual Desktop environment

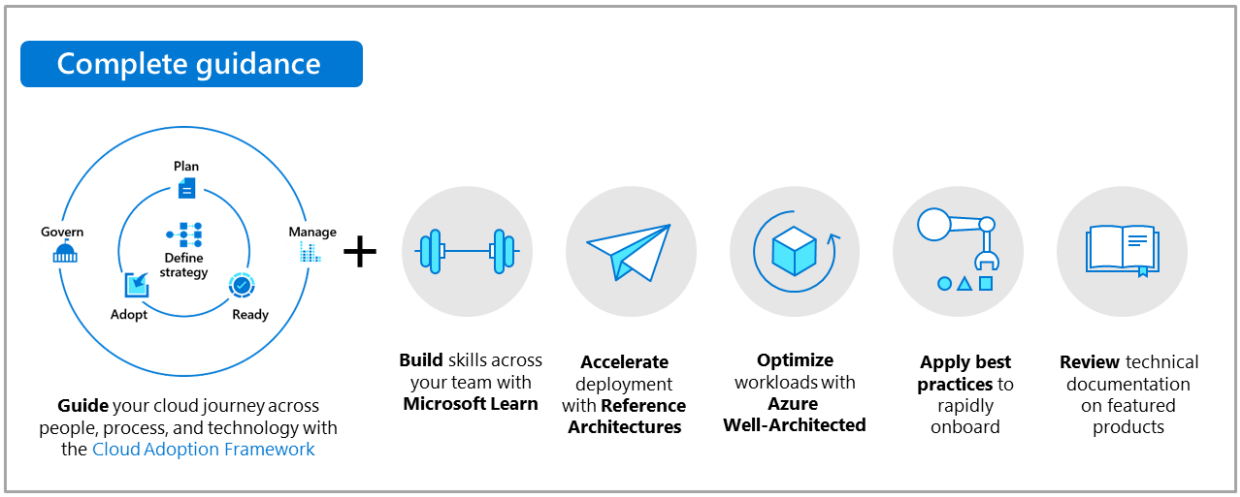
Cloud adoption for the retail industry

8/9/2021 • 2 minutes to read • Edit Online

As retail companies address larger and more sophisticated forms of cloud adoption, their journey to the cloud becomes more complex. This series of articles guides you in planning a retail cloud adoption journey.

Scenario components

This scenario is designed to guide a retail company's cloud adoption journey from beginning to end.



Use the following information to guide you on your journey:

	Microsoft Cloud Adoption Framework for Azure: These articles provide guidance for planning a cloud adoption journey. They are the right starting point for business decision makers, technology leaders, solution architects, and infrastructure architects.
	Microsoft Azure Well-Architected Framework: These articles outline what every retail workload owner should consider when deploying and managing workloads in the cloud.
	Retail solutions: These architecture solutions are developed and recommended for common retail industry scenarios. You can customize them to fit your business needs and accelerate deployment.
	Microsoft Learn modules: Learn the skills to implement, maintain, and support your solution.

Common customer challenges and supporting guidance

Prepare for centralized operations or cloud center of excellence: Review the Cloud Adoption Framework articles. Establish the processes and approaches required to support your workloads.

Monitor assets across an existing retail portfolio: Focus on the govern and manage articles to integrate Azure into your existing operations processes. Use the ready article to deploy those improvements across all of your cloud environments.

Influence changes to individual workloads (Central IT/Cloud Center of Excellence (CCoE)): As controls improve, the central IT teams will need to know the architectures that handle the workloads. Use the Azure Well-Architected Framework guidance to help workload owners improve the handling of the workloads.

Optimize individual workloads (workload teams): Workload owners should start with the Microsoft Azure Well-Architected Review guidance to understand the best ways to handle their workloads to support company business strategies. A central IT or CCoE team that supports the workload can use the guidance for insights into best practices and architectures to accelerate implementation.

Implement specific Azure products: Use Azure products outlined in the featured products section to improve and accelerate your implementation.

Next steps

The following articles can guide your cloud adoption journey and help you succeed in the cloud adoption scenario for the retail industry.

- Cloud adoption strategy for the retail industry
- Cloud adoption plan for the retail industry
- Review your environment or Azure landing zones
- Migrate common retail industry technologies
- Innovation in the retail industry
- Governance in the retail industry
- Management in the retail industry

Cloud adoption antipatterns

3/29/2021 • 2 minutes to read • Edit Online

Customers often experience cloud adoption antipatterns. These missteps commonly occur in design, planning, or implementation when migrating to the cloud. Antipatterns can block innovation and prevent businesses from adopting and realizing goals.

The following table lists antipatterns and the methodologies, or cloud adoption phases, that these patterns occur in. The linked articles provide examples of each antipattern and solutions.

METHODOLOGY	ANTIPATTERN	REFERENCE
Strategy	Inadequate motivation	Antipattern: adopt the cloud without establishing goals
Strategy	Misaligned motivation	Antipattern: fail to communicate motivations
Plan	Wrong cloud operating model	Antipattern: choose the wrong cloud operating model
Plan	Wrong service model	Antipattern: choose the wrong service model
Plan	Replacement instead of modernization	Antipattern: replace architecture
Ready	Preview services in production	Antipattern: assume released services are ready for production
Ready	Inaccurate resiliency and availability assumptions	Antipattern: assume increased resiliency and availability
Ready	IT as cloud provider	Antipattern: become a cloud provider
Adopt	Lack of guardrails	Antipattern: migrate, modernize, or innovate without guardrails
Adopt	Lack of assessments	Antipattern: migrate, modernize, or innovate without an assessment
Adopt	Forced architecture	Antipattern: dictate an architecture
Adopt	Single subscription	Antipattern: use a single subscription
Manage	Neglect of business outcomes	Antipattern: focus on tooling, not business outcomes
Govern	Misaligned shared responsibilities	Antipattern: misunderstand shared responsibilities

METHODOLOGY	ANTIPATTERN	REFERENCE
Govern	Inaccurate out-of-the-box security assumptions	Antipattern: assume out-of-the-box solutions provide security
Govern	Custom compliance or governance frameworks	Antipattern: use a custom compliance or governance framework
Organize	IT cost centers	Antipattern: treat IT as a cost center
Organize	Platform development without business approval	Antipattern: invest in new technology without involving the business
Organize	Core business function outsourcing	Antipattern: outsource core business functions
Organize	Technical decision makers instead of cloud engineers	Antipattern: hire technical decision makers instead of developing cloud engineers

Next steps

- Cloud strategy antipatterns
- Cloud adoption plan antipatterns

Develop a cloud adoption strategy

5/26/2021 • 2 minutes to read • Edit Online

The cloud delivers fundamental technology benefits that can help your enterprise execute multiple business strategies. By using cloud-based approaches, you can improve business agility, reduce costs, accelerate time to market, and enable expansion into new markets. To take advantage of this great potential, start by documenting your business strategy in a way that's both understandable to cloud technicians and palatable to your business stakeholders.

The following steps can help you document your business strategy efficiently. This approach helps you drive adoption efforts that capture targeted business value in a cross-functional model. Then, you can map your cloud adoption strategy to specific cloud capabilities and business strategies to reach your desired state of transformation.

	Define and document your motivations: Meet with key stakeholders and executives to document the motivations behind cloud adoption.
	Document business outcomes: Engage motivated stakeholders and executives to document specific business outcomes.
	Evaluate financial considerations: Learn how to use the cloud to make your IT cost structure more flexible and build a business case to adopt the cloud.
	Understand technical considerations: Discover the technical flexibility, efficiencies, and capabilities that help you build a business case to adopt the cloud.

Use the strategy and plan template to build out your cloud adoption strategy, and to track the output of each of the steps outlined above.

Motivations: Why are we moving to the cloud?

8/25/2021 • 4 minutes to read • Edit Online

"Why are we moving to the cloud?" It's a common question for business and technical stakeholders alike. If the answer is, "Our board (or CIO, or C-level executives) told us to move to the cloud," then it's unlikely that the business will achieve the desired outcomes.

This article discusses a few motivations behind cloud migration that can help produce more successful business outcomes. These options help facilitate a conversation about motivations and, ultimately, business outcomes.

Motivations

Business transformations that are supported by cloud adoption can be driven by various motivations. It's likely that several motivations apply at the same time. The goal of the lists in the following table is to help generate ideas about which motivations are relevant. From there, you can prioritize and assess the potential impacts of the motivations. In this article, your cloud adoption team should meet with various executives and business leaders using the following list to understand which of these motivations are affected by the cloud adoption effort.

CRITICAL BUSINESS EVENTS	MIGRATION	INNOVATION
Datacenter exit	Cost savings	Preparation for new technical capabilities
Merger, acquisition, or divestiture	Reduction in vendor or technical complexity	Building new technical capabilities
Reduction in capital expenses	Optimization of internal operations	Scaling to meet market demands
End of support for mission-critical technologies	Increase in business agility	Scaling to meet geographic demands
Response to regulatory compliance changes	Preparation for new technical capabilities	Improved customer experiences and engagements
New data sovereignty requirements	Scaling to meet market demands	Transformation of products or services
Reduction of disruptions and improvement of IT stability	Scaling to meet geographic demands	Market disruption with new products or services
Reduce carbon footprint	Integration of a complex IT portfolio	Democratization and/or self-service environments

Classify your motivations

Your motivations for cloud adoption will likely fall into multiple categories. As you're building the list of motivations, trends will likely emerge. Motivations tend to be associated more with one classification than with others. Use the predominant classification to help guide the development of your cloud adoption strategy.

When a response to critical business events is the highest priority, it's important to get started with migration early, often in parallel with strategy and planning efforts. Taking this approach requires a growth mindset and a willingness to iteratively improve processes, based on direct lessons learned.

When migration is the highest priority, strategy and planning will play a vital role early in the process. We recommend that you implement the first workload in parallel with planning efforts, to help the team understand

and anticipate any learning curves that are associated with cloud adoption.

When innovation is the highest priority, strategy and planning require more investments early in the process to ensure balance in the portfolio and wise alignment of the investment made during cloud adoption. For more information and guidance, see Understand the innovation journey.

To ensure wiser decision-making, all participants in the migration process should have a clear awareness of their motivations. The following section outlines how customers can guide and affect wiser decisions through consistent, strategic methodologies.

Motivation-driven strategies and business outcomes

This section highlights the *migration* and *innovation* motivations and their corresponding strategies.

Migration

The *migration* motivations listed near the top of the motivations table are the most common, but not necessarily the most significant, reasons for adopting the cloud. These outcomes are important to achieve, but they're most effectively used to transition to other, more useful worldviews. This important first step to cloud adoption is often called a *cloud migration*. The Migrate methodology in the Cloud Adoption Framework outlines the *strategy for executing a cloud migration*.

Some motivations align well with a migrate strategy. Motivations at the top of this list can have less business impact than those toward the bottom. **Strategies with migration driving motivations** have helped organizations to successfully create business outcomes that:

- Increase cost savings. Read the customer story.
- Reduce vendor or technical complexity.
- Optimize internal operations.
- Increase business agility. Read the customer story.
- Prepare for new technical capabilities.
- Scale to market demand.
- Scale to geographic demand. Read the customer story.

Innovation

Data is the new commodity, and modern applications are the supply chain driving that data into various experiences. In today's business market, it's hard to find a transformative product or service that isn't built on top of data, insights, and customer experiences. The Innovate methodology in the Cloud Adoption Framework includes *motivations aligned to a technology strategy* that appear lower in the *Innovation* column of the motivation list above.

The motivations below help IT organizations to focus more on innovation than a migration strategy. **Strategies with innovation driving motivations** have helped organizations to successfully create business outcomes that:

- Increase business agility.
- Prepare for new technical capabilities.
- Build new technical capabilities. Read the customer story.
- Scale to market demand.
- Scale to geographic demand.
- Improve customer experience and engagement. Read the customer story.
- Transform products or services.

Next steps

Understanding projected business outcomes helps facilitate the conversations that you need to have as you document your motivations and supporting metrics, in alignment with your business strategy. Next, read an overview of business outcomes that are commonly associated with a move to the cloud.

Overview of business outcomes

What business outcomes are associated with transformation journeys?

5/18/2021 • 3 minutes to read • Edit Online

The most successful transformation journeys start with a business outcome in mind. Cloud adoption can be a costly and time-consuming effort. Fostering the right level of support from IT and other areas of the business is crucial to success. This article series is designed to help customers identify business outcomes that are concise, defined, and drive observable results or change in business performance, supported by a specific measure.

During any cloud transformation, the ability to speak in terms of business outcomes supports transparency and cross-functional partnerships. The business outcome framework starts with a simple template to help technically minded individuals document and gain consensus. This template can be used with several business stakeholders to collect a variety of business outcomes, which could each be influenced by a company's transformation journey. Feel free to use this template electronically or, better still, draw it on a whiteboard to engage business leaders and stakeholders in outcome-focused discussions.

To learn more about business outcomes and the business outcome template, see Documenting business outcomes, or download the business outcome template.

Optimize your cloud investment using cloud economics

What are the fundamentals of cloud economics in Azure? Whether you're running existing workloads or designing new solutions in Azure, learn best practices guidance to navigate the economics of the cloud for your organization, and optimize your operating costs in Azure based on your specific workloads. Get started with successfully building your cloud business case with key financial and technical guidance, and maximize the full benefit of your cloud investment.

Learn more about how cloud economics works.

Prepare for conversations with different personas

The following are a few business outcomes that tend to trigger conversations with various personas:

- **Finance leadership:** Increase profitability while driving compliance.
- **Marketing:** Acquire and retain customers, build reputation.
- **Sales:** Accelerate sales, improve customer lifetime value.
- **Human resources:** Retain, recruit, and empower employees.
- **Executive leadership:** Meeting market growth requirements and environmental sustainability metrics.

Sample outcomes by category

Speaking in business outcomes can feel like a foreign language to many technically minded individuals. To help ease translation, we curate a set of business outcome examples. You can use the following examples to inspire and demonstrate business outcomes that are based on actual transformation journeys.

To help you find business outcomes more easily, we've separated them into the following categories. This approach tends to drive consensus-building conversations across business units.

Fiscal outcomes

Financial or fiscal performance is the cleanest business outcome for many business leaders, but not the only

one.

View samples of fiscal outcomes.

Agility outcomes

Today's fast-changing business environment places a premium on time. The ability to respond to and drive market change quickly is the fundamental measure of business agility.

View samples of agility outcomes.

Reach outcomes

In a constantly shrinking market, global reach (ability to support global customers and users) can be measured by compliance in geographies that are relevant to the business.

View outcomes related to global reach.

Customer engagement outcomes

Social marketplaces are redefining winners and losers at an unheard-of pace. Responding to user needs is a key measure of customer engagement.

Learn more about customer engagement outcomes.

Performance outcomes

Performance and reliability are assumed. When either falters, reputation damage can be painful and long-lasting.

Learn more about performance outcomes.

Sustainability goals

Organizations are increasingly discussing environmental goals and sustainability targets.

Learn more about sustainability goals.

Each of the business outcomes listed in the preceding categories can help facilitate a focused conversation among your business and technical team members. However, you shouldn't limit your conversations to these generic samples. Understanding the unique needs of your own business, and building outcomes that match, maximizes the value of a cloud transformation.

Next steps

Learn more about fiscal outcomes.

Fiscal outcomes

Data innovations

8/11/2021 • 4 minutes to read • Edit Online

Many companies want to migrate their existing data warehouse to the cloud. They are motivated by a number of factors, including:

- No hardware to buy or maintenance costs.
- No infrastructure to manage.
- The ability to switch to a secure, scalable, and low-cost cloud solution.

For example, the cloud-native, pay-as-you-go service from Azure called Azure Synapse Analytics provides an analytical database management system for organizations. Azure technologies help modernize your data warehouse after it's migrated and extend your analytical capabilities to drive new business value.

A data warehouse migration project involves many components. These include schema, data, extract-transform-load (ETL) pipelines, authorization privileges, users, BI tool semantic access layers, and analytic applications.

After your data warehouse has been migrated to Azure Synapse Analytics, you can take advantage of other technologies in the Microsoft analytical ecosystem. Doing so allows you to not only modernize your data warehouse but also bring together insights produced in other analytical data stores on Azure.

You can broaden ETL processing to ingest data of any type into Azure Data Lake Storage. You can prepare and integrate it at scale by using Azure Data Factory. This produces trusted, commonly understood data assets that can be consumed by your data warehouse, and also accessed by data scientists and other applications. You can build real-time, batch-oriented analytical pipelines. You can also create machine learning models that can deploy to run in batch, in real time on streaming data, and on demand.

In addition, you can use PolyBase to go beyond your data warehouse. This simplifies access to insights being produced in multiple underlying analytical platforms on Azure. You create holistic, integrated views in a logical data warehouse to gain access to streaming, big data, and traditional data warehouse insights from BI tools and applications.

Many companies have had data warehouses running in their datacenters for years, to enable users to produce business intelligence. Data warehouses extract data from known transaction systems, stage the data, and then clean, transform, and integrate it to populate data warehouses.

Use cases, business cases, and technology advances all support how Azure Synapse Analytics can help you with data warehouse migration. The following sections list many of these examples.

Use cases

- Connected product innovation
- Factory of the future
- Clinical analytics
- Compliance analytics
- Cost-based analytics
- Omnichannel optimization
- Personalization
- Intelligent supply chain
- Dynamic pricing
- Procurement analytics

- Digital control tower
- Risk management
- Customer analytics
- Fraud detection
- Claims analytics

Business cases

- Build end-to-end analytics solutions with a single analytics service.
- Use the Azure Synapse Analytics studio, which provides a unified workspace for data prep, data management and analytics, data warehousing, big data, and AI tasks.
- Build and manage pipeline with a no-code visual environment, automate query optimization, build proofs of concept, and use Power BI, all from the same analytics service.
- Deliver your data insights to data warehouses and big data analytics systems.
- For mission-critical workloads, optimize the performance of all queries with intelligent workload management, workload isolation, and limitless concurrency.
- Edit and build Power BI dashboards directly from Azure Synapse Analytics.
- Reduce project development time for BI and machine learning projects.
- Easily share data with just a few clicks by using Azure Data Share integration within Azure Synapse Analytics.
- Implement fine-grained access control with column-level security and native row-level security.
- Automatically protect sensitive data in real time with dynamic data masking.
- Industry-leading security with built-in security features like automated threat detection and always-on data encryption.

Technology advances

- No hardware to buy or maintenance costs so you pay only for what you use.
- No infrastructure to manage, so you can focus on competitive insights.
- Massively parallel SQL query processing with dynamic scalability when you need it, and the option to shut down or pause when you don't.
- Ability to independently scale storage from compute.
- You can avoid unnecessary, expensive upgrades caused by the staging areas on your data warehouse getting too big, taking up storage capacity, and forcing an upgrade. For example, move the staging area to Azure Data Lake Storage. Then process it with an ETL tool like Azure Data Factory or your existing ETL tool running on Azure at lower cost.
- Avoid expensive hardware upgrades by processing ETL workloads in Azure, by using Azure Data Lake Storage and Azure Data Factory. This is often a better solution than running on your existing data warehouse DBMS with SQL query processing doing the work. As staging data volumes increase, more storage and compute power underpinning your on-premises data warehouse is consumed by ETL. This in turn affects the performance of query, reporting, and analysis workloads.
- Avoid building expensive data marts that use storage and databases software licenses on on-premises hardware. You can build them in Azure Synapse Analytics instead. This is especially helpful if your data warehouse is a data vault design, which often causes an increased demand for data marts.
- Avoid the cost of analyzing and storing high-velocity, high-volume data on on-premises hardware. For example, if you need to analyze real-time, machine generated data like click-stream and streaming IoT data in your data warehouse, you can use Azure Synapse Analytics.
- You can avoid paying a premium for storing data on expensive warehouse hardware in the datacenter as your data warehouse grows. Azure Synapse Analytics can store your data in cloud storage at a lower cost.

Next steps

Data democratization

Support business with data innovation

5/18/2021 • 2 minutes to read • Edit Online

Many companies keep data warehouses in their datacenters to help different parts of their business analyze data and make decisions. Sales, marketing, and finance departments rely heavily on these systems in order to produce standard reports and dashboards. Companies also employ business analysts to perform ad hoc querying and analysis of data in data marts. These data marts use self-service business intelligence tools to perform multidimensional analysis.

A business that's supported by data innovation and a modern data estate can empower a broad range of contributors, from an IT stakeholder to a data professional and beyond. They can take action on this repository of centralized data, which is often referred to as "the single source of truth."

Azure Synapse Analytics is a single service for seamless collaboration and accelerated time-to-insight. To understand this service in more detail, first consider the various roles and skills involved in a typical data estate:

Data warehousing: Database admins support the management of data lakes and data warehouses while intelligently optimizing workloads and automatically securing data.

Data integration: Data engineers use a code-free environment to easily connect multiple sources and types of data.

Big data and machine learning: Data scientists build proofs of concept rapidly and provision resources while working in the language of their choice (for example, T-SQL, Python, Scala, .NET, or Spark SQL).

Management and security: IT pros protect and manage data more efficiently, enforce privacy requirements, and secure access to cloud and hybrid configurations.

Business intelligence: Business analysts securely access datasets, build dashboards, and share data within and outside their organization.

An overview of classic data warehouse architecture

The following diagram shows an example of a classic data warehouse architecture.

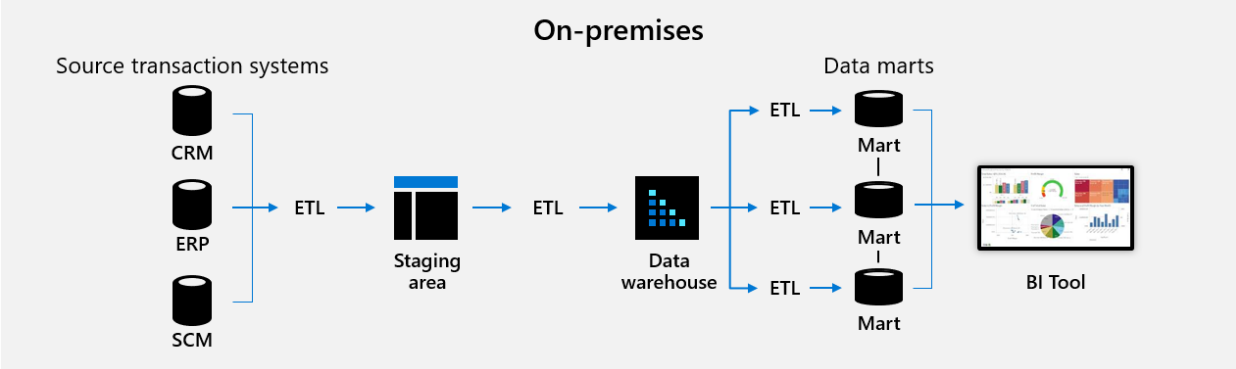


Figure 1: Classic data warehouse architecture.

Known structured data is extracted from core transaction processing systems and copied into a staging area. From there, it's cleaned, transformed, and integrated into production tables in a data warehouse. It's common for several years of historical transaction data to build up here. This provides the data needed to understand changes in sales, customer purchasing behavior, and customer segmentation over time. It also provides yearly financial reporting and analysis to help with decision making.

From there, subsets of data are extracted into data marts to analyze activity associated with a specific business process. This supports decision making in a specific part of the business.

For a business to run efficiently, it needs all types of data for the different skills and roles described earlier. You need raw data that has been cleansed for data scientists to build machine-learning models. You need clean and structured data for a data warehouse to provide reliable performance to business applications and dashboards. Most importantly, you need to be able to go from raw data to insights in minutes, not days.

Azure Synapse Analytics has a native, built-in business intelligence tool with Microsoft Power BI. Here, one service within one interface supports you to quickly transform raw data into a dashboard that displays insights.

Next steps

Data innovations

Examples of fiscal outcomes

5/18/2021 • 8 minutes to read • Edit Online

At the top level, fiscal conversations consist of three basic concepts:

- **Revenue:** Will more money come into the business as a result of the sales of goods or services.
- **Cost:** Will less money be spent in the creation, marketing, sales, or delivery of goods or services.
- **Profit:** Although they're rare, some transformations can both increase revenue and decrease costs. This is a profit outcome.

The remainder of this article explains these fiscal outcomes in the context of a cloud transformation.

Sentara Healthcare moved its data to Azure, providing members a better experience with mobile apps, faster access for clinicians to critical data for faster diagnoses, reduced costs for the IT team, and improved patient care with a more agile, change-ready organization.

> **NOTE**
> The following examples are hypothetical and should not be considered a guarantee of returns when adopting any cloud strategy.

Revenue outcomes

New revenue streams

The cloud can help create opportunities to deliver new products to customers or deliver existing products in a new way. New revenue streams are innovative, entrepreneurial, and exciting for many people in the business world. New revenue streams are also prone to failure and are considered by many companies to be high risk. When revenue-related outcomes are proposed by IT, there will likely be resistance. To add credibility to these outcomes, partner with a business leader who's a proven innovator. Validation of the revenue stream early in the process helps avoid roadblocks from the business.

- **Example:** A company has been selling books for over a hundred years. An employee of the company realizes that the content can be delivered electronically. The employee creates a device that can be sold in the bookstore, which allows the same books to be downloaded directly, driving *$x* in new book sales.

Revenue increases

With global scale and digital reach, the cloud can help businesses to increase revenues from existing revenue streams. Often, this type of outcome comes from an alignment with sales or marketing leadership.

- **Example:** A company that sells widgets could sell more widgets, if the salespeople could securely access the company's digital catalog and stock levels. Unfortunately, that data is only in the company's ERP system, which can be accessed only via a network-connected device. Creating a service façade to interface with the ERP and exposing the catalog list and nonsensitive stock levels to an application in the cloud would allow the salespeople to access the data they need while onsite with a customer. Extending on-premises Active Directory using Azure Active Directory (Azure AD) and integrating role-based access into the application would allow the company to help ensure that the data stays safe. This simple project could affect revenue from an existing product line by *x%*.

Profit increases

Seldom does a single effort simultaneously increase revenue and decrease costs. However, when it does, align

the outcome statements from one or more of the revenue outcomes with one or more of the cost outcomes to communicate the desired outcome.

Cost outcomes

Cost reduction

Cloud computing can reduce capital expenses for hardware and software, setting up datacenters, running on-site datacenters, and so on. The costs of racks of servers, round-the-clock electricity for power and cooling, and IT experts for managing the infrastructure add up fast. Shutting down a datacenter can reduce capital expense commitments. This is commonly referred to as "getting out of the datacenter business." Cost reduction is typically measured in dollars in the current budget, which could span one to five years depending on how the CFO manages finances.

- **Example #1:** A company's datacenter consumes a large percentage of the annual IT budget. IT chooses to conduct a cloud migration and transitions the assets in that datacenter to infrastructure as a service (IaaS) solutions, creating a three-year cost reduction.
- **Example #2:** A holding company recently acquired a new company. In the acquisition, the terms dictate that the new entity should be removed from the current datacenters within six months. Failure to do so will result in a fine of $1 million USD per month to the holding company. Moving the digital assets to the cloud in a cloud migration could allow for a quick decommission of the old assets.
- **Example #3:** An income tax company that caters to consumers experiences 70 percent of its annual revenue during the first three months of the year. The remainder of the year, its large IT investment is relatively dormant. A cloud migration could allow IT to deploy the compute/hosting capacity required for those three months. During the remaining nine months, the IaaS costs could be significantly reduced by shrinking the compute footprint.

Example: Coverdell

Coverdell modernizes their infrastructure to drive record cost savings with Azure. Coverdell's decision to invest in Azure, and to unite their network of websites, applications, data, and infrastructure within this environment, led to more cost savings than the company could have ever expected. The migration to an Azure-only environment eliminated $54,000 USD in monthly costs for colocation services. With the company's new united infrastructure alone, coverdell expects to save an estimated $1M USD over the next two to three years.

> "Having access to the Azure technology stack opens the door for some scalable, easy-to-implement, and highly available solutions that are cost effective. This allows our architects to be much more creative with the solutions they provide."
>
> Ryan Sorensen
>
> Director of Application Development and Enterprise Architecture
>
> Coverdell

Cost avoidance

Terminating a datacenter can also provide cost avoidance, by preventing future refresh cycles. A refresh cycle is the process of buying new hardware and software to replace aging on-premises systems. In Azure, hardware and OS are routinely maintained, patched, and refreshed at no additional cost to customers. This allows a CFO to remove planned future spend from long-term financial forecasts. Cost avoidance is measured in dollars. It differs from cost reduction, generally focusing on a future budget that has not been fully approved yet.

- **Example:** A company's datacenter is up for a lease renewal in six months. The datacenter has been in service for eight years. Four years ago, all servers were refreshed and virtualized, costing the company millions of dollars. Next year, the company plans to refresh the hardware and software again. Migrating the assets in that datacenter as part of a cloud migration would allow cost avoidance by removing the planned refresh

from next year's forecasted budget. It could also produce cost reduction by decreasing or eliminating the real estate lease costs.

Capital expenses and operating expenses

Before you discuss cost outcomes, it's important to understand the two primary cost options: capital expenses and operating expenses.

The following terms will help you understand the differences between capital expenses and operating expenses during business discussions about a transformation journey.

- **Capital** is the money and assets owned by a business to contribute to a particular purpose, such as increasing server capacity or building an application.
- **Capital expenditures** generate benefits over a long period. These expenditures are generally nonrecurring and result in the acquisition of permanent assets. Building an application could qualify as a capital expenditure.
- **Operating expenditures** are ongoing costs of doing business. Consuming cloud services in a pay-as-you-go model could qualify as an operating expenditure.
- **Assets** are economic resources that can be owned or controlled to produce value. Servers, data lakes, and applications can all be considered assets.
- **Depreciation** is a decrease in the value of an asset over time. More relevant to the capital expense versus operating expense conversation, depreciation is how the costs of an asset are allocated across the periods in which they are used. For example, if you build an application this year but it's expected to have an average shelf life of five years (like most commercial applications), the cost of the development team and the tools required to create and deploy the code base would be depreciated evenly over five years.
- **Valuation** is the process of estimating how much a company is worth. In most industries, valuation is based on the company's ability to generate revenue and profit, while respecting the operating costs required to create the goods that provide that revenue. In some industries, such as retail, or in some transaction types, such as private equity, assets and depreciation can play a large part in the company's valuation.

It is often a safe bet that various executives, including the chief investment officer (CIO), debate the best use of capital to grow the company in the desired direction. Giving the CIO a means of converting contentious capital expense conversations into clear accountability for operating expenses could be an attractive outcome by itself. In many industries, chief financial officers (CFOs) are actively seeking ways of better associating fiscal accountability to the cost of goods being sold.

However, before you associate any transformation journey with this type of capital versus operating expense conversion, it's wise to meet with members of the CFO or CIO teams to see which cost structure the business prefers. In some organizations, reducing capital expenses in favor of operating expenses is a highly undesirable outcome. As previously mentioned, this approach is sometimes seen in retail, holding, and private equity companies that place higher value on traditional asset accounting models, which place little value on IP. It's also seen in organizations that had negative experiences when they outsourced IT staff or other functions in the past.

If an operating expense model is desirable, the following example could be a viable business outcome:

- **Example:** The company's datacenter is currently depreciating at *$x USD* per year for the next three years. It is expected to require an additional *$y USD* to refresh the hardware next year. We can convert the capital expenses to an operating expense model at an even rate of *$z USD* per month, allowing for better management of and accountability for the operating costs of technology.

Next steps

Learn more about agility outcomes.

Agility outcomes

Examples of agility outcomes

3/6/2021 • 3 minutes to read • Edit Online

As discussed in the business outcomes overview, several potential business outcomes can serve as the foundation for any transformation journey conversation with the business. This article focuses on the timeliest business measure: business agility. Understanding your company's market position and competitive landscape can help you articulate the business outcomes that are the target of the business's transformation journey.

Traditionally, chief investment officers and IT teams were considered a source of stability in core mission-critical processes. This is still true. Few businesses can function well when their IT platform is unstable. However, in today's business world, much more is expected. IT can expand beyond a simple cost center by partnering with the business to provide market advantages. Many chief investment officers and executives assume that stability is simply a baseline for IT. For these leaders, business agility is the measure of IT's contribution to the business.

The Academy of Motion Picture Arts and Sciences migrated its legacy web applications to the cloud, delivering new streaming applications to members in a rich, responsive, and cross-platform experience.

Why is agility so important?

Markets change at a faster pace today than ever before. As of 2015, only 57 companies were still in the Fortune 500 61 years later, an 88.6 percent turnover rate. This represents market change at a previously unheard-of rate. IT agility or even business agilities are unlikely to affect an organization listing on the Fortune 500, but these figures help us understand the pace at which markets continue to change.

For incumbents and upstarts alike, business agility can be the difference between success or failure of a business initiative. Quickly adapting to market changes can help ring-fence existing customers or claim market share from competitors. The agility-related outcomes in the next sections can help communicate the value of the cloud during a transformation.

Time-to-market outcome

During cloud-enabled innovation efforts, time to market is a key measure of IT's ability to address market change. In many cases, a business leader might have existing budget for the creation of an application or the launch of a new product. Clearly communicating a time-to-market benefit can motivate that leader to redirect budget to IT's transformation journey.

- **Example 1:** The European division of a US-based company needs to comply with GDPR regulations by protecting customer data in a database that supports UK operations. Their existing version of SQL Server doesn't support the necessary row-level security. An in-place upgrade would be too disruptive. Using Azure SQL Database to replicate and upgrade the database, the customer adds the necessary compliance measure in a matter of weeks.
- **Example 2:** A logistics company has discovered an untapped segment of the market, but it needs a new version of their flagship application to capture this market share. Their larger competitor has made the same discovery. Through the execution of a cloud-enabled application innovation effort, the company embraces customer obsession and a DevOps-driven development approach to beat their slower, legacy competitor by *x* months. This jump on market entrance secured the customer base.

Aurora Health Care

Healthcare system transforms online services into a friendly digital experience. To transform its digital services, Aurora Health Care migrated its websites to the Microsoft Azure platform and adopted a strategy of continuous

innovation.

> "As a team, we're focused on high-quality solutions and speed. Choosing Azure was a very transformative decision for us."
>
> Jamey Shiels
>
> Vice President of Digital Experience
>
> Aurora Health Care

Provision time

When business demands new IT services or scale to existing services, acquisition and provision of new hardware or virtual resources can take weeks. After cloud migration, IT can more easily enable self-service provisioning, allowing the business to scale in hours.

- **Example:** A consumer packaged goods company requires the creation and tear-down of hundreds of database clusters per year to fulfill operational demands of the business. The on-premises virtual hosts can provision quickly, but the process of recovering virtual assets is slow and requires significant time from the team. As such, the legacy on-premises environment suffers from bloat and can seldom keep up with demand. After cloud migration, IT can more easily provide scripted self-provisioning of resources, with a chargeback approach to billing. Together, this allows the business to move as quickly as they need, but still be accountable for the cost of the resources they demand. Doing so in the cloud limits deployments to the business's budget only.

Next steps

Learn more about reach outcomes.

Reach outcomes

Examples of global reach outcomes

3/6/2021 • 3 minutes to read • Edit Online

As discussed in Business outcomes, several potential business outcomes can serve as the foundation for any transformation journey conversation with the business. This article focuses on a common business measure: reach. *Reach* is a concise term that, in this case, refers to a company's globalization strategy. Understanding the company's globalization strategy helps you better articulate the business outcomes that are the target of a business's transformation journey.

Fortune 500 and smaller enterprises have focused on the globalization of services and customers for over three decades, and most business are likely to engage in global commerce as this globalization continues to pull focus. Hosting datacenters around the world can consume more than 80 percent of an annual IT budget, and wide-area networks using private lines to connect those datacenters can cost millions of dollars per year. Therefore, supporting global operations is both challenging and costly.

Cloud solutions move the cost of globalization to the cloud provider. In Azure, customers can quickly deploy resources in the same region as customers or operations, without buying and provisioning a datacenter. Microsoft owns one of the largest wide-area networks in the world, connecting datacenters around the globe. Connectivity and global operating capacity are available to global customers on demand.

Walgreens Boots Alliance (WBA) moved on-premises applications and IT resources in a heterogeneous Linux and Windows environment to the cloud, benefiting from improved performance and data centralization - and helping the company provide better customer service.

Global access

Expanding into a new market can be one of the most valuable business outcomes during a transformation. The ability to quickly deploy resources in market without a longer-term commitment allows sales and operations leaders to explore options that wouldn't have been considered in the past.

Manufacturing example

A cosmetics manufacturer has identified a trend. Some products are being shipped to the Asia Pacific region even though no sales teams are operating in that region. The minimum systems required by a remote sales force are small, but latency prevents a remote access solution. To capitalize on this trend, the vice president of sales wants to experiment with sales teams in Japan and South Korea. Because the company has undergone a cloud migration, it was able to deploy the necessary systems in both Japan and South Korea within days. This allowed the vice president of sales to grow revenue in the region within three months. Those two markets continue to outperform other parts of the world, leading to sales operations throughout the region.

Retail example

An online retailer that ships products globally can engage with their customers across time zones and multiple languages. The retailer uses Azure Bot Service and various features in Azure Cognitive Services, such as Translator, Language Understanding (LUIS), QnA Maker, and Text Analytics. This ensures their customers are able to get the information they need when they need it, and that it's provided to them in their language. The retailer uses the Personalizer service to further customize the experience and catalog offerings for their customers, ensuring geographical tastes, preferences, and availability are reflected.

Data sovereignty

Operating in new markets introduces additional governance constraints. Azure provides compliance offerings that help customers meet compliance obligations across regulated industries and global markets. For more

information, see the overview of Microsoft Azure compliance.

Example

A US-based utilities provider was awarded a contract to provide utilities in Canada. Canadian data sovereignty law requires that Canadian data stays in Canada. This company had been working their way through a cloud-enabled application innovation effort for years. As a result, their software was deployed through fully scripted DevOps processes. With a few minor changes to the code base, they were able to deploy a working copy of the code to an Azure datacenter in Canada, meeting data sovereignty compliance and retaining the customer.

Next steps

Learn more about customer engagement outcomes.

Customer engagement outcomes

Examples of customer engagement outcomes

5/18/2021 • 2 minutes to read • Edit Online

As discussed in the business outcomes overview, several potential business outcomes can serve as the foundation for any transformation journey conversation with the business. This article focuses on a common business measure: customer engagement. Understanding the needs of customers, and the ecosystem around customers, helps you to articulate the business outcomes that are the target of a business's transformation journey.

During cloud-enabled data innovation efforts, you can assume that customers are engaged. The following functions are potentially disruptive and require a high degree of customer engagement:

- Aggregating data
- Testing theories
- Advancing insights
- Informing cultural change

Customer engagement outcomes are about meeting and exceeding customer expectations. As a baseline for customer engagements, customers assume that products and services perform and are reliable. When they're not, it's easy for an executive to understand the business value of performance and reliability outcomes. For more advanced companies, the speed of integrating learnings and observations from this process is a fundamental business outcome.

Descartes chose Microsoft Azure as its preferred platform, and successfully migrated its Descartes MacroPoint solution to Azure SQL Database to provide greater flexibility for customers, and focus internal resources on extending product value.

Cycle time

During customer-obsessed transformations such as a cloud-enabled application innovation effort, customers respond from direct engagement. They also appreciate seeing their needs met quickly by the development team. Cycle time is a Six Sigma term that refers to the duration from the start to the finish of a function. For business leaders who invest heavily in improving customer engagement, cycle time can be a strong business outcome.

Example:

A services company that provides business-to-business (B2B) services is trying to retain market share in a competitive market. Customers who have left for a competing service provider found that their overly complex technical solution interferes with their business processes, and is the primary reason for leaving. In this case, cycle time is imperative.

It currently takes 12 months for a feature to progress from request to release. If it's prioritized by the executive team, this cycle can shorten from nine to six months. The team can cut cycle time down to one month through a cloud-enabled application innovation effort, cloud-native application models, and Azure DevOps integration. This frees business and application development teams to interact more directly with customers.

Intelligent contact center

Customer satisfaction and experience are at the core of successful organizations. Freeing your employees to focus on superior customer service can strongly affect customer loyalty and retention. With the AI technology available today, many steps during a customer call can be automated, enabling the contact center agent more

time to focus on delivering superior customer service.

Example:

An insurance company has implemented digital agents to respond rapidly to customer requests. These digital agents are available through the company website and mobile app, by building an Azure Bot Service solution. Extending an enhanced customer service experience to their contact center, the insurance company implemented live call transcription, sentiment analysis, and key phrase detection. These help the contact center agent with recommended next steps and form processing. This led to reduced repetition from the customer calling the contact center, and enabled the contact center agent to focus more on providing a great customer experience.

Next steps

Learn more about performance outcomes.

Performance outcomes

Examples of performance outcomes

3/6/2021 • 3 minutes to read • Edit Online

As discussed in Business outcomes, several potential business outcomes can serve as the foundation for any transformation journey conversation with the business. This article focuses on a common business measure: performance.

In today's technological society, customers assume that applications will perform well and always be available. When this expectation isn't met, it causes reputation damage that can be costly and long-lasting.

GE Aviation's Digital Group deployed Microsoft Azure Digital Twins and other Azure resources to ingest and model data from multiple sources, building a complete digital aircraft view with built-in digital traceability of individual components that helps customers drive greater fuel efficiency, reduce maintenance costs, and boost their fleets' flight-readiness.

Performance

The biggest cloud computing services run on a worldwide network of secure datacenters, which are regularly upgraded to the latest generation of fast and efficient computing hardware. This provides several benefits over a single corporate datacenter, such as reduced network latency for applications and greater economies of scale.

Transform your business and reduce costs with an energy-efficient infrastructure that spans more than 100 highly secure facilities worldwide, linked by one of the largest networks on earth. Azure has more global regions than any other cloud provider. This translates into the scale that's required to bring applications closer to users around the world, preserve data residency, and provide comprehensive compliance and resiliency options for customers.

- **Example 1:** A services company worked a hosting provider that hosted multiple operational infrastructure assets. Those systems suffered from frequent outages and poor performance. The company migrated its assets to Azure to take advantage of the SLA and performance controls of the cloud. Any downtime would cost the company approximately $15,000 USD per minute of outage. With between four and eight hours of outage per month, it was easy to justify this organizational transformation.
- **Example 2:** A consumer investment company was in the early stages of a cloud-enabled application innovation effort. Agile processes and DevOps were maturing well, but application performance was spiky. As a more mature transformation, the company started a program to monitor and automate sizing based on usage demands. The company eliminated sizing issues by using Azure performance management tools, resulting in a surprising 5 percent increase in transactions.

Reliability

Cloud computing makes data backup, disaster recovery, and business continuity easier and less expensive, because data can be mirrored at multiple redundant sites on the cloud provider's network.

One of IT's crucial functions is ensuring that corporate data is never lost and applications stay available despite server crashes, power outages, or natural disasters. You can keep your data safe and recoverable by backing it up to Azure.

Azure Backup is a simple solution that decreases your infrastructure costs while providing enhanced security mechanisms to protect your data against ransomware. With one solution, you can protect workloads that are running in Azure and on-premises across Linux, Windows, VMware, and Hyper-V. You can ensure business

continuity by keeping your applications running in Azure.

Azure Site Recovery makes it simple to test disaster recovery by replicating applications between Azure regions. You can also replicate on-premises VMware and Hyper-V virtual machines and physical servers to Azure to stay available if the primary site goes down. And you can recover workloads to the primary site when it's up and running again.

- **Example:** An oil and gas company used Azure technologies to implement a full site recovery. The company chose not to fully embrace the cloud for day-to-day operations, but the cloud's business continuity and disaster recovery (BCDR) features still protected their datacenter. As a hurricane formed hundreds of miles away, their implementation partner started recovering the site to Azure. Before the hurricane touched down, all mission-critical assets were running in Azure, preventing any downtime.

Next steps

Learn how to use the business outcome template.

Use the business outcome template

Sustainability outcomes and benefits for business

6/22/2021 • 3 minutes to read • Edit Online

Though the impact and benefits of the cloud have been traditionally measured with financial and efficiency metrics, it's become more common for customers to seek understanding about how the cloud can help them to achieve their sustainability and environmental goals. Cloud computing can support your organization to reduce carbon emissions, use resources more efficiently, and lessen your environmental footprint.

Watch the following video to learn more about sustainability and how cloud migration opens the door to sustainable solutions that are good for the planet and good for your business.

Microsoft has been leading in many of these areas. The company has been operating as carbon-neutral since 2012 and has made a commitment to be carbon-negative by 2030. The carbon benefits of cloud computing, a study on the Microsoft cloud in partnership with WSP, supports research on how moving on-premises datacenters to the Microsoft cloud can significantly reduce carbon footprints.

Bühler Group's's IoT platform, known as Bühler Insights and powered by Azure IoT Hub, generates vital data so customers can monitor machine performance and generate accurate records for every product batch, helping food producers optimize safety, sustainability, and transparency across the supply chain.

The Microsoft sustainability journey

The Microsoft journey started over a decade ago when the company started to apply new business practices and adopt cloud technology. We lowered our carbon emissions by 30 percent in 2009. Since then, Microsoft has made large strides forward by investing in reducing the company's carbon footprint further. Microsoft has focused on these four areas:

- **Carbon:** Cutting energy consumption across corporate offices, charging carbon tax to business divisions, and using cloud-powered technology to lower emissions.
- **Ecosystem:** Making a commitment to green datacenters and purchasing of 1.1 billion kilowatt-hours of green energy.
- **Water:** Reducing use intensity and investing in technology for managing water.
- **Waste:** Practicing responsible sourcing, recycling, and disposal; using software and technology to make buildings more efficient.

Read more about how Microsoft's commitment to a planet-sized challenge has helped us plan and achieve sustainability goals.

Examples of sustainability outcomes

Focusing on sustainability and protecting limited environmental resources is key to our future, and this focus also benefits business. Today, companies can draw from a broad range of assets and resources that can help them to expand into new geographic areas and develop innovative resource management solutions.

AGL, one of Australia's leading integrated energy companies, built a solution on Azure that remotely manages networked solar batteries. Learn about how the company is growing an innovative energy partnership across Australia to help local customers give back to the grid.

Bee'ah is a sustainability pioneer in the Middle East that believes in technology and sustainability creating solutions for the future. Their services include waste management, environmental consulting, renewable energy,

and sustainable transportation. Azure has supported the company to launch the first AI platform to digitize all operations and services. Read more about how the cloud drives sustainable management and digital innovation throughout the company's sustainability journey.

These customer stories demonstrate how prioritizing sustainability and environmental solutions can help organizations to create new business opportunities.

Next steps

An intentional approach can help organizations to navigate their sustainability journey. These four steps can influence outcomes for your company:

Step 1: Record and understand your company's carbon emissions. Start by categorizing your emissions, which will help you to list of areas on which to focus. The Microsoft sustainability calculator can assist you with this task.

Step 2: Evaluate whether your vendors, partners, and providers are taking steps to reduce their emissions and if these steps align with yours.

Step 3: Create an incentive for teams to reduce carbon emissions. The Microsoft carbon fee: theory and practice guide can help your organization to drive alignment and accountability across your teams.

Step 4: Seek out teams in our business to enlist their support and generate ideas for areas for improvement. Build an innovation culture where individuals are participants with a sense of ownership.

Learn more about how your organization can measure and reach sustainability outcomes with the cloud.

Reach outcomes

How to use the business outcome template

3/6/2021 • 2 minutes to read • Edit Online

As discussed in the business outcomes overview, it can be difficult to bridge the gap between business and technical conversations. This simple template is designed to help teams uniformly capture business outcomes to be used later in the development of customer transformation journey strategies.

Download the business outcome template to begin brainstorming and tracking business outcomes. Continue reading to learn how to use the template. Review the business outcomes section for ideas on potential business outcomes that could come up in executive conversations.

Use the business outcome template

In this template, business outcomes focus on three topics:

- Aligning to stakeholders or business decision makers.
- Understanding business drivers and objectives.
- Mapping outcomes to specific solutions and technical capability.

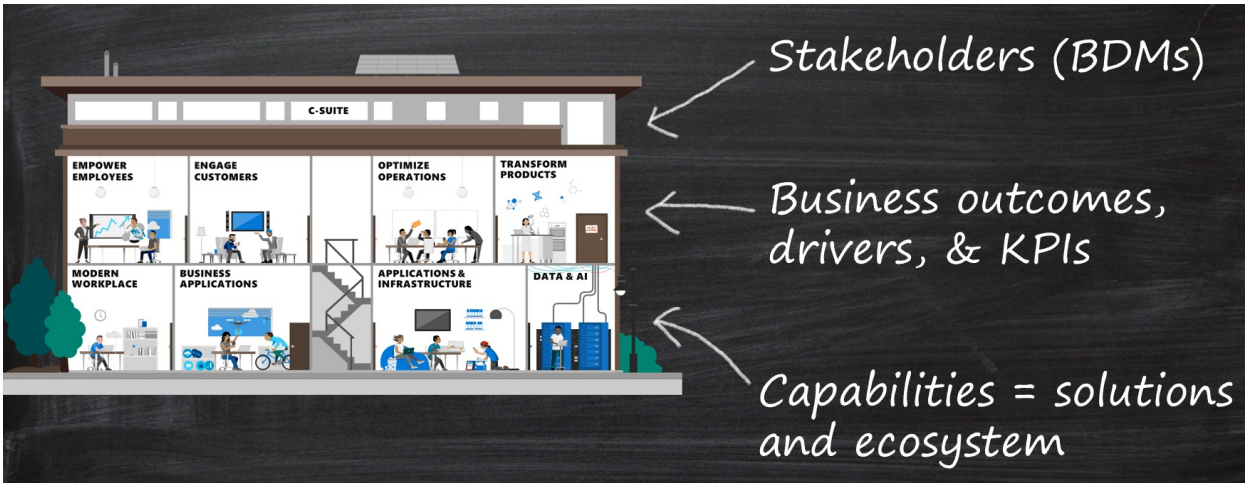


Figure 1: Business outcomes visualized as a house with stakeholders, over business outcomes, over technical capabilities.

The business outcome template focuses on simplified conversations that can quickly engage stakeholders without getting too deep into the technical solution. By rapidly understanding and aligning the key performance indicators (KPIs) and business drivers that are important to stakeholders, your team can think about high-level approaches and transformations before diving into the implementation details.

An example can be found on the "example outcome" tab of the spreadsheet, as shown below. To track multiple outcomes, add them to the "collective outcomes" tab.

XYZ Life Sciences Co. - Drug & Device Division

Stakeholder:	*Therapeutic Product Owner*	**Business Outcome:**	*Decrease Time to Market for drug/device*

Business Drivers	KPI	Capabilities
Study Design	$3M opportunity cost per day per drug	Data-Driven protocol authoring
Study Conduct		Trial Simulation
		Structured collaboration for trial approval

Figure 2: Example of a business outcome template.

Why is this template relevant?

Discovery is a fundamental tenet of enterprise architecture. If discovery is limited to technical discovery, the solution is likely to miss many opportunities to improve the business. Enterprise architects, solution architects, and other technically minded leaders can master the discovery process by using this template. In effective discovery processes, these leaders consider five key aspects of the business outcome before leading a transformation journey, as shown in the following image:

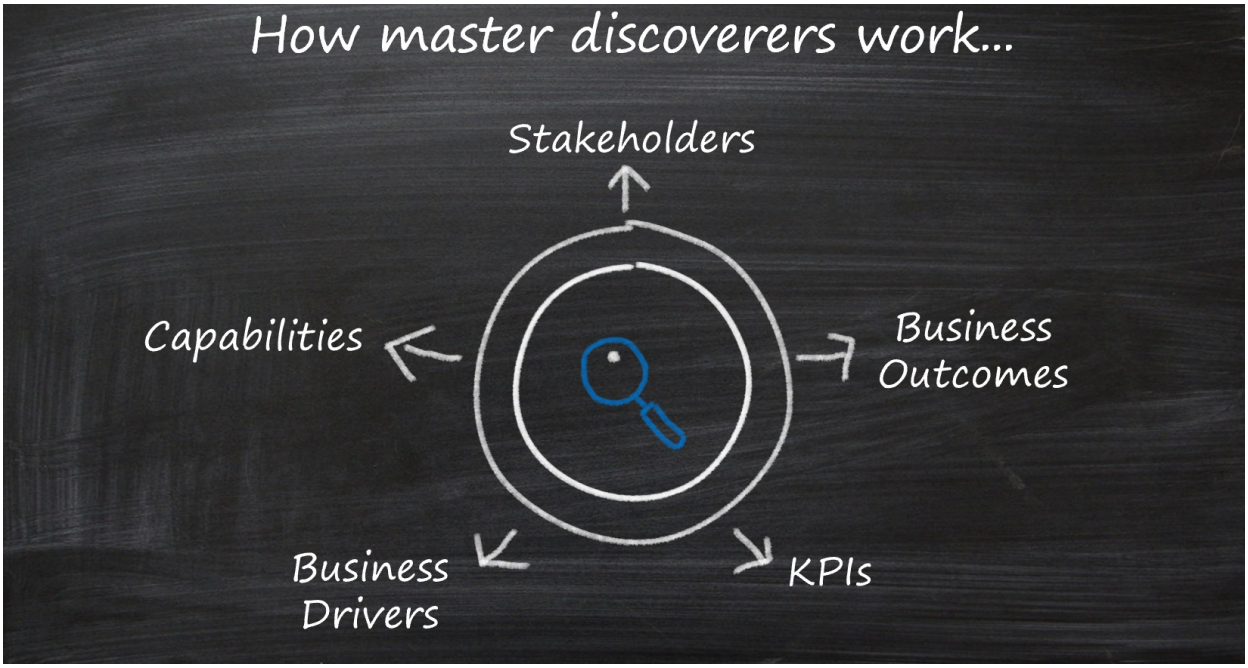


Figure 3: Five areas of focus in discovery: stakeholders, outcomes, drivers, KPIs, and capabilities.

Stakeholders: Who in the organization is likely to see the greatest value in a specific business outcome? Who is most likely to support this transformation, especially when things get tough or time consuming? Who has the greatest stake in the success of this transformation? This person is a potential stakeholder.

Business outcomes: A business outcome is a concise, defined, and observable result or change in business performance, supported by a specific measure. How does the stakeholder want to change the business? How will the business be affected? What is the value of this transformation?

Business drivers: Business drivers capture the current challenge that's preventing the company from achieving desired outcomes. They can also capture new opportunities that the business can capitalize on with the right solution. How would you describe the current challenges or future state of the business? What business functions would be changing to meet the desired outcomes?

KPIs: How will this change be measured? How does the business know whether they are successful? How frequently will this KPI be observed? Understanding each KPI helps enable incremental change and experimentation.

Capabilities: When you define any transformation journey, how will technical capabilities accelerate realization of the business outcome? What applications must be included in the transformation to achieve business objectives? How do various applications or workloads get prioritized to deliver on capabilities? How do parts of the solution need to be expanded or rearchitected to meet each of the outcomes? Can execution approaches (or timelines) be rearranged to prioritize high-impact business outcomes?

Next steps

Learn to align your technical efforts to meaningful learning metrics.

Align your technical efforts

How can we align efforts to meaningful learning metrics?

3/6/2021 • 3 minutes to read • Edit Online

The business outcomes overview discussed ways to measure and communicate the impact a transformation will have on the business. Unfortunately, it can take years for some of those outcomes to produce measurable results. The board and C-suite are unhappy with reports that show a 0% delta for long periods of time.

Learning metrics are interim, shorter-term metrics that can be tied back to longer-term business outcomes. These metrics align well with a growth mindset and help position the culture to become more resilient. Rather than highlighting the anticipated lack of progress toward a long-term business goal, learning metrics highlight early indicators of success. The metrics also highlight early indicators of failure, which are likely to produce the greatest opportunity for you to learn and adjust the plan.

As with much of the material in this framework, we assume you're familiar with the transformation journey that best aligns with your desired business outcomes. This article will outline a few learning metrics for each transformation journey to illustrate the concept.

Cloud migration

This transformation focuses on cost, complexity, and efficiency, with an emphasis on IT operations. The most easily measured data behind this transformation is the movement of assets to the cloud. In this kind of transformation, the digital estate is measured by virtual machines (VMs), racks or clusters that host those VMs, datacenter operational costs, required capital expenses to maintain systems, and depreciation of those assets over time.

As VMs are moved to the cloud, dependence on on-premises legacy assets is reduced. The cost of asset maintenance is also reduced. Unfortunately, businesses can't realize the cost reduction until clusters are deprovisioned and datacenter leases expire. In many cases, the full value of the effort isn't realized until the depreciation cycles are complete.

Always align with the CFO or finance office before making financial statements. However, IT teams can generally estimate current monetary cost and future monetary cost values for each VM based on CPU, memory, and storage consumed. You can then apply that value to each migrated VM to estimate the immediate cost savings and future monetary value of the effort.

Application innovation

Cloud-enabled application innovation focuses largely on the customer experience and the customer's willingness to consume products and services provided by the company. It takes time for increments of change to affect consumer or customer buying behaviors. But application innovation cycles tend to be much shorter than they are in the other forms of transformation. The traditional advice is that you should start with an understanding of the specific behaviors that you want to influence and use those behaviors as the learning metrics. For example, in an e-commerce application, total purchases or add-on purchases could be the target behavior. For a video company, time watching video streams could be the target.

Customer behavior metrics can easily be influenced by outside variables, so it's often important to include related statistics with the learning metrics. These related statistics can include release cadence, bugs resolved per release, code coverage of unit tests, number of page views, page throughput, page load time, and other application performance metrics. Each can show different activities and changes to the code base and the

customer experience to correlate with higher-level customer behavior patterns.

Data innovation

Changing an industry, disrupting markets, or transforming products and services can take years. In a cloud-enabled data innovation effort, experimentation is key to measuring success. Be transparent by sharing prediction metrics like percent probability, number of failed experiments, and number of models trained. Failures will accumulate faster than successes. These metrics can be discouraging, and the executive team must understand the time and investment needed to use these metrics properly.

On the other hand, some positive indicators are often associated with data-driven learning: centralization of heterogeneous data sets, data ingress, and democratization of data. While the team is learning about the customer of tomorrow, real results can be produced today. Supporting learning metrics could include:

- Number of models available.
- Number of partner data sources consumed.
- Devices producing ingress data.
- Volume of ingress data.
- Types of data.

An even more valuable metric is the number of dashboards created from combined data sources. This number reflects the current-state business processes that are affected by new data sources. By sharing new data sources openly, your business can take advantage of the data by using reporting tools like Power BI to produce incremental insights and drive business change.

Next steps

After learning metrics are aligned, you're ready to begin building the business case to deliver against those metrics.

Build the cloud business case

Measure business outcomes using objectives and key results (OKRs)

5/18/2021 • 3 minutes to read • Edit Online

Modern operations require modern ways to measure business outcomes, and cloud technology can help to increase velocity for a business. An organization's measurement platform should support a company's outcomes and plan for growth by:

- Providing insights to team members and groups.
- Supporting staff to pivot quickly when outcomes don't align with strategy and expectations.
- Offering a structured format, templates, sequences, and tools to help teams plan and visualize for increasing velocity.

An overview of objectives and key results (OKRs)

Many organizations have started to adopt objectives and key results (OKRs). OKRs have proven to drive alignment in complex work environments, foster innovation, and help individuals to focus on what matters. The two components comprising OKRs are an objective and key results for that objective. An objective is the statement of intent: what is the team trying to accomplish, and why is it important? Key results are specific outcomes that track impact on the objective.

Objective: Clarity and intent.

Key results: Measures of success within a quarter.

It's important to understand that OKRs are useful for measuring team outcomes versus individual performance. Since deadlines often motivate team performance, key results are established quarterly. OKRs help teams to focus on the most important tasks instead of the volume of work at hand.

To do this, focus on what happens in a month, a quarter, and other short-term intervals. You can have OKRs that last longer, but shorter intervals emphasize the need for OKRs that track short-term impact.

OKR key principles

WorkBoard is a company that focuses solely on OKRs and offers software solutions to help customers adopt them. According to the company, the key principles of OKRs are:

- **Aspire and inspire:** Teams establish their best possible results in a given quarter, focus efforts on great outcomes, and use retros to learn and iterate.
- **Outcome focus:** Quarterly key results provide clarity on where value is created. This helps teams and the organization to drive business impacts faster.
- **Global and local:** Teams localize OKRs into their nouns, verbs, and numbers that enrich OKRs with the team's expertise and insights.
- **Transparent:** OKRs, alignment, and progress are visible to everyone with OKR software, simplifying collaboration and making good decisions faster.

How OKRs add value to an organization

OKRs help to create alignment and accountability within organizations.

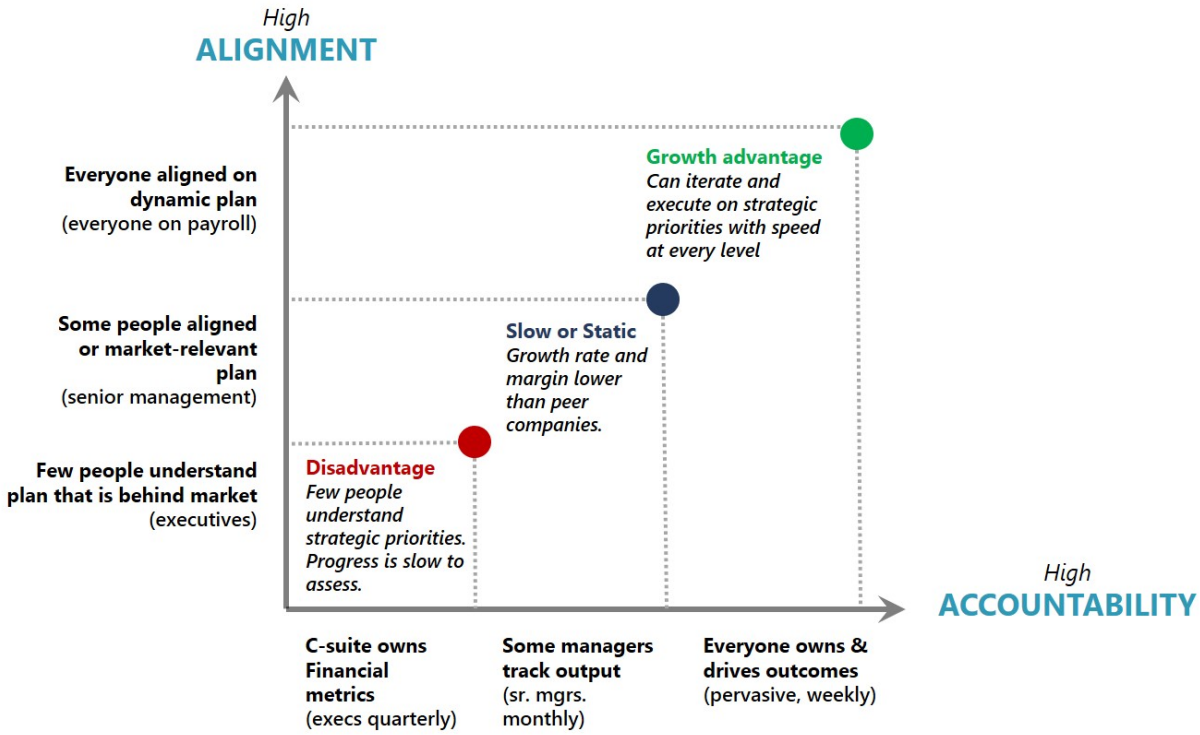


Figure 1: How OKRs increase alignment and accountability within organizations to help them meet goals faster.

Learn more about how your team can align strategy and execution during planning and execution phases.

Examples of OKRs

The principles defined by WorkBoard can help your organization to understand how useful OKRs take form. Objectives need to inspire your company and its teams to fully understand your mission. Key results need to be specific and measurable within the quarter.

Here are some example of OKRs:

Objective 1: Be the top US provider of learning platforms to schools.

Key results:

1. 45 percent of K-12 schools using our platform
2. A 12 percent increase in student engagement, as measured through internal systems
3. A 95 percent satisfaction rate from quarterly parent surveys

Objective 2: Build a technology platform that supports every person in our business to innovate and create.

Key results:

1. Five new applications developed and adopted across the organization
2. Every team with at least two members using the Microsoft Power Platform
3. Including new cloud technologies like data analytics and machine learning in all customer-facing applications

Objective 3: Transform our approach from sales-driven to data-driven.

Key results:

1. Increasing pipeline coverage from 50 percent to 200 percent
2. Increasing closing rates for sales engagements by 5 percent
3. Reducing the time to close deals by 8 percent

Next steps

Five steps can help your organization to move forward with OKRs:

Step 1: Learn: Start exploring what OKRs can do for your business, and tune in to some of your industry peers and leaders to learn about how OKRs have benefited their organizations.

Step 2: Plan: As you begin to draft your OKRs, ensure that your sponsors are contributing and involved in the process. Work with an OKR coach to refine your OKRs.

Step 3: Launch: Each organization launches initiatives differently. Maintain a strong communication plan, and build the OKR calibration and celebration process into your operating model.

Step 4: Drive: To maintain rigor and focus, make sure that you're sharing outcomes and results across the organization. This will help your teams to adopt a habit of using OKRs.

Step 5: Improve: Continue to improve, revisit, and rethink how to connect further across the organization. OKRs in spreadsheets can be useful, but an organization can benefit most from everyone participating to meet objectives and gaining insights from the aligned data.

Contact WorkBoard to get started.

Align efforts for learning metrics

Understand financial considerations

5/20/2021 • 2 minutes to read • Edit Online

A digital transformation goes hand-in-hand with a financial transformation. When you're making the shift to the cloud, there are financial considerations around how the cloud will impact your financial position, accounting KPIs, and processes that Chief Financial Officers (CFO) and finance teams need to understand. Your organization's motivations and business outcomes will help determine how you look at your financials.

This guidance will help you learn how to use the cloud to make your IT cost structure more flexible and help you build a business case to migrate to the cloud.

How does cloud pricing work?

The cloud uses a pay-for-what-you-consume model versus the up-front server infrastructure and software licensing costs that you would typically pay on-premises in your data center. The cloud allows you to take advantage of a variable cost model instead of a fixed cost model.

CAPEX to OPEX

One benefit of moving to the cloud is that it shifts how you pay for capacity, from CAPEX to OPEX. Your overall budget allocation moves from a CAPEX investment to OPEX pricing models that can fluctuate based on capacity or utilization of the cloud environment. Your organization will realize meaningful improvements in financial statements, with improved cash flow timing and a reduced need to acquire assets that result in a fixed cost structure.

Reduced data center footprint

A reduced data center footprint is another benefit of a shift to the cloud. On-premises data centers are frequently overbuilt for peaks, resulting in excess capacity and excess spend. In the cloud, you can spin up resources quickly with the flexibility of scalability, where you can scale up and down as needed. Azure operates in many different geographic regions, giving you flexibility to choose where to build your applications. Amortized facilities or co-lo or hosting agreements that you that you have in place are no longer required.

Increased productivity and service delivery

How can the cloud help your team be more productive? It's critical to consider how DevOps the cloud can improve and drive efficiency in your organization. The cloud can help unlock better DevOps processes, resulting in faster time to market, increased revenue, increased employee productivity, and the ability to deploy products more quickly.

Sustainability

Migrating to the cloud can reduce your carbon footprint and your organization immediately benefits from a sustainability perspective. Data centers that are hosting cloud services, such as Azure, are done at such scale that you benefit from overall sustainability with the least amount of environmental impact.

Next steps

The financial considerations guidance above, and subsequent guidance in the Strategy methodology help you build a business case to migrate to the cloud. Before building your business case, review common finance terms

that can help.

Glossary of common finance terms

Glossary of common finance terms

5/20/2021 • 2 minutes to read • Edit Online

Use these common finance terms when your team is creating a cloud migration business case. These terms can help when you share your business case with a finance team.

Terms

Amortization: An expense tied to a typically intangible asset, that reflects the "economic" usage of that asset in a particular time period. For example if you purchase a license worth $100, you would capitalize that on your balance sheet. If you amortized it over 5-years, you would annually recognize an expense of $20 per year that impacts your income statement.

Balance sheet: A balance sheet is a financial statement that reports a company's assets, liabilities, and shareholders' equity as of a specific date.

Capital Expense (CAPEX): The upfront investment in equipment. This equipment is capitalized as an asset and put on your balance sheet.

Cash flow statement: A cash flow statement is a financial statement that summarizes the amount of cash and cash equivalents entering and leaving a company during a given period.

Cloud economics: An understanding of the benefits and costs of the cloud, and the financial impact when you start a migration from on-premises to cloud computing.

Depreciation: An expense tied to a capitalized asset, that reflects the "economic" usage of that asset in a particular time period. For example if you purchase a server worth $100, you would capitalize that on your balance sheet. If you depreciated it over 5-years, you would annually recognize an expense of $20 per year that impacts your income statement.

Double mortgage period: A period when you have two sets of costs at the same time. For example, when you have both on-premises and cloud costs.

Earnings before interest, taxes, depreciation, and amortization (EBITDA): A performance indicator of the profitability of a business. This starts from "operating income" which is the income from your ongoing business operations (ignoring things like taxes or interest expense) and adds back depreciation and amortization. While a useful performance metric that is used for comparability, it is often viewed in conjunction metrics like Capital Expenditure to have a better all-up understanding of a companies ability to generate free cash flow.

Net Present Value (NPV): An assessment of the financial value of a business investment. This metric looks at cash flows, timing, and the required interest rate.

Operating Expense (OPEX): The ongoing expenses for a business. For example, a maintenance payment or periodic bill for Azure services.

Profit and Loss (P&L): A financial statement that summarizes the revenues, costs, and expenses incurred over a specified period, usually a fiscal quarter, or year. It is also referred to as the income statement.

Return on Investment (ROI): Return on investment (ROI) is a metric used to understand the profitability of an investment. ROI compares how much you paid for an investment to how much you earned to evaluate its efficiency.

Next steps

Learn more about the most common cloud accounting models for IT.

Understand cloud accounting

What is cloud accounting?

6/8/2021 • 4 minutes to read • Edit Online

The cloud changes how IT accounts for costs, as is described in Create a financial model for cloud transformation. Various IT accounting models are much easier to support because of how the cloud allocates costs. So it's important to understand how to account for cloud costs before you begin a cloud transformation journey. This article outlines the most common cloud accounting models for IT.

Traditional IT accounting (cost center model)

It's often accurate to consider IT a cost center. In the traditional IT accounting model, IT consolidates purchasing power for all IT assets. As we pointed out in the financial models article, that purchasing power consolidation can include software licenses, recurring charges for CRM licensing, purchase of employee desktops, and other large costs.

When IT serves as a cost center, the perceived value of IT is largely viewed through a procurement management lens. This perception makes it difficult for the board or other executives to understand the true value that IT provides. Procurement costs tend to skew the view of IT by outweighing any other value added by the organization. This view explains why IT is often lumped into the responsibilities of either the chief financial officer or the chief operating officer. This perception of IT is limited and might be shortsighted.

Central IT accounting (profit center model)

To overcome the cost center view of IT, some CIOs opted for a centralized IT model of accounting. In this type of model, IT is treated like a competing business unit and a peer to revenue-producing business units. In some cases, this model can be entirely logical. For example, some organizations have a professional IT services division that generates a revenue stream. Frequently, centralized IT models don't generate significant revenue, making it difficult to justify the model.

Regardless of the revenue model, centralized IT accounting models are unique because of how the IT unit accounts for costs. In a traditional IT model, the IT team records costs and pays those costs from shared funds like operations and maintenance (O&M) or a dedicated profit and loss (P&L) account.

In a central IT accounting model, the IT team marks up the services provided to account for overhead, management, and other estimated expenses. It then bills the competing business units for the marked-up services. In this model, the CIO is expected to manage the P&L associated with the sale of those services. This can create inflated IT costs and contention between central IT and business units, especially when IT needs to cut costs or isn't meeting agreed-upon SLAs. During times of technology or market change, any new technology would cause a disruption to central IT's P&L, making transformation difficult.

Chargeback

One of the common first steps in changing IT's reputation as a cost center is implementing a chargeback model of accounting. This model is especially common in smaller enterprises or highly efficient IT organizations. In the chargeback model, any IT costs that are associated with a specific business unit are treated like an operating expense in that business unit's budget. This practice reduces the cumulative cost effects on IT, allowing business values to show more clearly.

In a legacy on-premises model, chargeback is difficult to realize because someone still has to carry the large capital expenses and depreciation. The ongoing conversion from capital expenditures to operating expenses associated with usage is a difficult accounting exercise. This difficulty is a major reason for the creation of the

traditional IT accounting model and the central IT accounting model. The operating expenses model of cloud cost accounting is almost required if you want to efficiently deliver a chargeback model.

But you shouldn't implement this model without considering the implications. Here are a few consequences that are unique to a chargeback model:

- Chargeback results in a massive reduction of the overall IT budget. For IT organizations that are inefficient or require extensive complex technical skills in operations or maintenance, this model can expose those expenses in an unhealthy way.
- Loss of control is a common consequence. In highly political environments, chargeback can result in loss of control and staff being reallocated to the business. This could create significant inefficiencies and reduce IT's ability to consistently meet SLAs or project requirements.
- Difficulty accounting for shared services is another common consequence. If the organization has grown through acquisition and is carrying technical debt as a result, it's likely that a high percentage of shared services must be maintained to keep all systems working together effectively.

Cloud transformations include solutions to these and other consequences associated with a chargeback model. But each of those solutions includes implementation and operating expenses. The CIO and CFO should carefully weigh the pros and cons of a chargeback model before considering one.

Showback or awareness-back

For larger enterprises, a showback or awareness-back model is a safer first step in the transition from cost center to value center. This model doesn't affect financial accounting. In fact, the P&Ls of each organization don't change. The biggest shift is in mindset and awareness. In a showback or awareness-back model, IT manages the centralized, consolidated buying power as an agent for the business. In reports back to the business, IT attributes any direct costs to the relevant business unit, which reduces the perceived budget directly consumed by IT. IT also plans budgets based on the needs of the associated business units, which allows IT to more accurately account for costs associated to purely IT initiatives.

This model provides a balance between a true chargeback model and more traditional models of IT accounting.

Impact of cloud accounting models

The choice of accounting models is crucial in system design. The choice of accounting model can affect subscription strategies, naming standards, tagging standards, and policy and blueprint designs.

After you've worked with the business to make decisions about a cloud accounting model and global markets, you can learn more about how to achieve more with your investment in the cloud.

Achieve more with your investment in the cloud

Understand technical considerations

5/20/2021 • 2 minutes to read • Edit Online

When you're making the shift to the cloud, there are technical considerations around how it will help improve how you manage and maintain your cloud and workloads. This guidance will help you discover the technical flexibility, efficiencies, and capabilities that aren't possible with your on-premises IT infrastructure and help you build a business case to migrate to the cloud.

Technical benefits

Scalability

Scalability, or the ability to scale out your resources depending on usage, utilization, and demand is one of the main technical benefits of moving to the cloud.

Availability

On-premises, it's more costly to build highly available infrastructure. It's less costly to architect highly available infrastructure in the cloud.

Security and compliance

When it comes to security and compliance, Microsoft is continually expanding our security infrastructure and toolsets to keep you on par with what's transpiring with respect to security threats on global networks.

Capacity optimization

Capacity optimization, where you only for the resources you utilize over time, is another technical benefit of the cloud. The core concept to consider is how elasticity and on-demand resources help you deploy, provision, or deprovision resources more dynamically.

Get the most out of your investment

Achieve more with your investment

With a shift to the cloud, it's important to think differently about how you'll consume and manage your cloud resources. As you build your business case, it's critical to understand the key principles of cloud economics. When you plan short- and long-term cloud solutions and align them to business outcomes, you can achieve more with every dollar you invest.

Align your partner strategy

The Cloud Adoption Framework approaches cloud adoption as a self-service activity. The objective is to empower each of the teams supporting adoption through standardized approaches. In practice, you can't assume that a self-service approach will be sufficient for all adoption activities.

Successful cloud adoption programs typically involve at least one level of support. Some cloud adoption efforts may require support from multiple partners working together towards a common goal.

Next steps

Learn how you can achieve more with your Azure investment.

Achieve more with your investment in the cloud

Achieve more with your investment in the cloud

7/15/2021 • 6 minutes to read • Edit Online

When migrating to the cloud, it's important to think differently about how you'll consume and manage your cloud resources. As you build your business case, it's critical to understand the key principles of cloud economics and transform your mindset. Part of this transformation is discovering technical and financial flexibility, efficiencies, and capabilities that aren't possible with your on-premises IT infrastructure. When you plan short- and long-term cloud solutions and align them to business outcomes, you can achieve more with every dollar you invest.

How does cloud pricing work?

Cloud costs are tied to compute and storage and include the underlying software licensing fees. The cloud uses a pay-for-what-you-consume model versus the up-front server infrastructure and software licensing costs that you would typically pay on-premises in your data center. On-premises, you typically have a combination of upfront costs and operating expenditures. When you move to the cloud, you shift to the pay-as-you-consume model and mainly operating expenditures.

To take advantage of the best pricing in the cloud, you must understand how you'll consume resources for your specific workloads. Once you have a consumption plan, you can establish your fixed and variable cost models to maximize your investment.

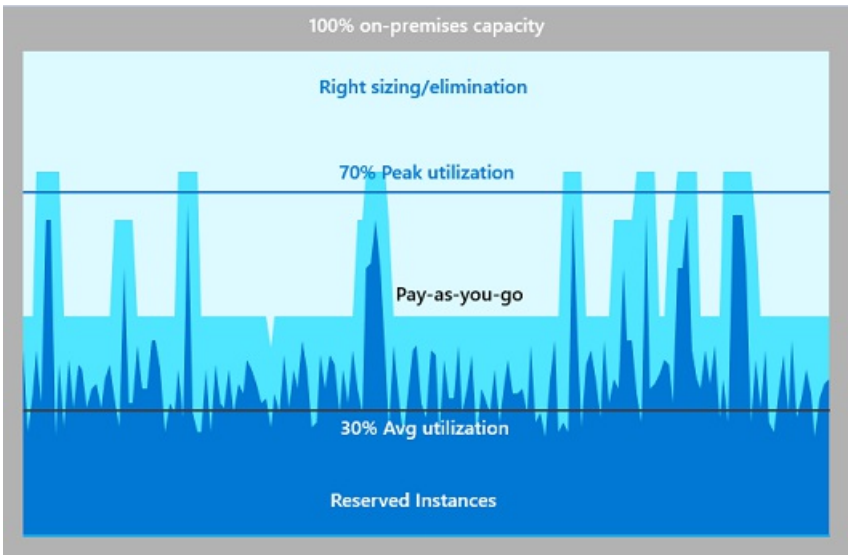
Understand your workloads

On-premises, your architecture is typically provisioned for peak capacity. Migrating from on-premises to the cloud gives you the flexibility of scalability, and you scale up and down as needed. It's critical to understand your workloads to realize the full benefits of the cloud.

Idle capacity: Azure helps eliminate the idle capacity created by overprovisioned architecture for coverage during peak usage. Rightsizing and eliminating workloads you don't need helps reduce your idle capacity when moving to the cloud. This exercise delivers immediate savings and cash flow reductions.

Unpredictable workloads: You can scale your compute resources up and down in the cloud as the demands of your business change. You're able to scale your capacity up and down and use a variable cost model as opposed to a fixed cost model. This elasticity of the cloud makes the pay-for-what-you-consume model possible and works well for your unpredictable workloads. Consider using virtual machine scale sets and snoozing VMs to only pay for the resources you need when you need them.

Predictable workloads: For your predictable workloads, you can take advantage of the cost-savings offers such as Azure Reservations.



Initial clean-up, right-sizing, and optimization: When planning to move to Azure, review which workloads are no longer needed. This process of clean-up can help you build a stronger business case and show an immediate effect on your budgets. For workloads you still want to use and bring to the cloud, you can use tools to help optimize them, for example, Azure Migrate.

Take advantage of cost-savings offers

There are many cost-savings offers the can help reduce your cloud costs:

Azure Hybrid Benefit: Reduce the costs of running your workloads in the cloud by using this licensing benefit. You can use your on-premises Software Assurance-enabled Windows Server and SQL Server licenses on Azure. This benefit applies to RedHat and SUSE Linux subscriptions, too. To learn more, see Azure Hybrid Benefit.

Spot virtual machines: You can use spot virtual machines with deep discounts for workloads that can be interrupted and don't need to complete within a specific time frame. For example, high-performance computing scenarios, batch processing jobs, visual rendering applications, dev/test environments, including continuous integration and continuous delivery workloads, or large-scale stateless applications. To learn more, see Spot virtual machines.

Reservations: Receive a discount on your workloads when you reserve your resources in advance. In return, Microsoft passes the savings onto you as discounts of up to 72 percent.1 For more information, see Azure reservations.

Azure Dev/Test pricing: Take advantage of discounted rates for your development and testing, including the Microsoft software charges on Azure Virtual Machines and special dev/test pricing on other services. For more information, see Azure Dev/Test pricing.

Extended security updates: Receive continued support for SQL Server 2008 and SQL Server 2008 R2 in the cloud, which has reached the end of their support lifecycle. You can migrate your on-premises SQL Server instances to Azure Virtual Machines, Azure SQL Database, or stay on-premises and purchase extended security updates. You'll receive free extended security patches by migrating to an Azure Virtual Machine. To learn more, see Lifecycle FAQ: Extended Security Updates.

Continually optimize your environment

Microsoft provides frameworks and tools to help you understand your costs and continually optimize your environment:

Understand and forecast your costs: Monitor your bill, set budgets, and allocate costs to teams and projects with Azure Cost Management + Billing.

Learn more:

- Optimize costs from recommendations.
- Prevent unexpected charges.

Cost optimize your workloads: Optimize your resources and architecture with Azure best practices from Azure Advisor and the Microsoft Azure Well-Architected Framework.

Learn more:

- Read about Azure Advisor.
- Get Azure Advisor cost recommendations.
- Learn about the Microsoft Azure Well-Architected Review.
- Learn about the Microsoft Azure Well-Architected Framework.

Save with Azure offers and licensing terms like the Azure Hybrid Benefit and Azure Reservations.:

Learn more:

- Learn about the Azure Hybrid Benefit.
- Learn about Azure Hybrid Benefit for Windows Server.
- Review pricing guidance for SQL Server Azure VMs.
- Learn about Azure Reservations.
- Read the reserved instances FAQ.

Control your costs: Establish spending goals and policies with guidance from the Microsoft Cloud Adoption Framework for Azure. Implement cost controls with Azure Policy so your teams can go fast while complying with policy. For more information, see enforce tagging conventions using Azure Policy.

Understand your financial stories

The core financial benefits of Azure are driven by a fundamental shift in the IT operating model. This shift benefits your organization's core financial statements and frees up cash flow for reinvestment:

Balance sheet: When you operate on-premises in datacenters, you typically have invested up front in long-term assets that limit the cash and capital required to grow your business. While in the cloud, you can shift datacenter operations costs into modernization, developing cloud applications, and other projects that drive business growth. This shift can make your balance sheet more agile.

Cash flow statement: The pay-for-what-you-consume model and the ability to apply policies and tags to your Azure resources help you improve the predictability of your cash flow statement. This model delays spend and improves your cash flow timing.

Income statement (profit and loss): You can improve profitability over time by taking advantage of Azure's flexibility, low management costs, services, and pricing models.

Achieve more with your investment

The goal of your cloud business case is to achieve more with every dollar invested. This goal can be accomplished by releasing committed cash flows and budgets that can be reinvested into further modernization. This concept is the velocity of the dollar - you accelerate value per dollar through phased reinvestment driving modernization and value.

The initial technical benefits of a cloud migration focus on the lift and shift model, where you migrate workloads to infrastructure-as-a-service (IaaS) in the cloud. The goal is to get the most out of your on-premises investment, and then move those workloads to IaaS, potentially freeing up cash flow. Historically, this process

would be considered a savings opportunity. This approach in the cloud is better viewed as a *reinvestment opportunity*.

As you free up cash flow, continue your cloud adoption, and mature your workloads, you can reinvest the savings to modernize to different service levels. Once you have your initial workloads in IaaS, you might consider shifting some workloads to platform-as-a-service (PaaS). You'll still provide the same type of service delivery however, you'll deliver it at a lower cost with more features and functionality. The next step in the iterative modernization process is moving some workflows and line-of-business applications to software-as-a-service (SaaS).

When you plan your phases of cloud maturity and the reinvestment of your cloud savings, and align with business outcomes, you can achieve more with every dollar you invest.

Next steps

Learn more about how to align your partner strategy when you migrate to the cloud.

Strategy for partner alignment

Strategy for partner alignment

7/14/2021 • 5 minutes to read • Edit Online

The Cloud Adoption Framework approaches cloud adoption as a self-service activity. The objective is to empower each of the teams supporting adoption through standardized approaches. In practice, you can't assume that a self-service approach will be sufficient for all adoption activities.

Successful cloud adoption programs typically involve at least one level of support. Some cloud adoption efforts may require support from multiple partners working together towards a common goal.

Steps to align the partnership strategy

It's important to start aligning your partnership strategy during the strategizing phase of adoption. The following steps can help remove roadblocks in later phases of the adoption lifecycle.

1. Start to understand support needs.
2. Consider partnership options that fit your culture and needs.
3. Evaluate a shortlist of partner options.
4. Begin contract and paperwork reviews with selected partners.

Completing these steps early, will ensure success of the team when the technical efforts begin. The following sections of this article provide guidance for each of these steps.

Understanding support needs

Throughout the cloud adoption lifecycle, the various teams may require support to be successful. The following are a few examples of the types of help commonly required.

- **Strategy:** Support defining the business strategy, building a business case, and supporting technology strategy.
- **Plan:** Support with discovery of the portfolio, quantitative assessment of the digital estate, development of a cloud adoption plan, creation of a skilling plan.
- **Ready:** Support deploying a landing zone or full cloud environment capable of supporting the cloud adoption plan.
- **Migrate:** Assistance migrating workloads or building a migration factory to ensure sound migration processes.
- **Innovate:** Assistance developing new solutions or rebuilding/rearchitecting existing solutions to drive innovation.
- **Govern:** Support or ongoing managed services to provide governance and controls across the cloud environment.
- **Manage:** Support or ongoing managed services to operate the cloud platform and the workloads hosted in the cloud.

Few corporations have the diversity of skills required to support strategy, planning, readiness, adoption, governance, and management. Partners and other support models are often necessary to fill in the gaps in the team's skills and responsibilities.

Various partnership options can help develop needed skills, augment staffing requirement, or completely offload-specific processes.

Partnership options

You are not alone in your cloud journey. There are several options to support your team throughout your cloud adoption journey.

- **Azure solution providers (partners):** Get connected with Azure expert-managed services providers (MSP) and other Microsoft partners who have service offerings aligned to the Cloud Adoption Framework methodologies.
- **FastTrack for Azure:** Use the Microsoft FastTrack for Azure program to accelerate migration.
- **Azure Migration and Modernization Program (AMMP):** The AMMP program aligns a mixture of partners and Microsoft employees to accelerate and support your migration.
- **Solution assessments:** Get assistance from a Microsoft solutions assessment expert or qualified partner as part of a Solution Assessment engagement.

Azure solution providers

Microsoft certified solution providers specialize in providing modern customer solutions base on Microsoft technologies across the world. Optimize your business in the cloud with help from an experienced partner.

Find a Cloud Solution Provider (CSP). A certified CSP can help take full advantage of the cloud by assessing business goals for cloud adoption, identifying the right cloud solution that meets business needs and helps the business become more agile and efficient.

Azure expert-managed services providers (MSP) have undergone a third-party audit to validate a higher tier of capability, demonstrated through certified staff headcounts, customer references, annual consumption of Azure at scale, and other criteria.

Find a managed service partner. An Azure managed service partner (MSP) helps a business transition to Azure by guiding all aspects of the cloud journey. From consulting to migrations and operations management, cloud MSPs show customers all the benefits that come with cloud adoption. They also act as a one-stop shop for common support, provisioning, and the billing experience, all with a flexible pay-as-you-go business model.

In parallel to the development of the cloud adoption strategy, the cloud strategy team should start to identify solution providers that can partner in the delivery of business objectives.

FastTrack for Azure

FastTrack for Azure provides direct assistance from Azure engineers, working hand in hand with partners, to help customers build Azure solutions quickly and confidently. FastTrack brings best practices and tools from real customer experiences to guide customers from setup, configuration, and development to production of Azure solutions, including:

During a typical FastTrack for Azure engagement, Microsoft helps to define the business vision to plan and develop Azure solutions successfully. The team assesses architectural needs and provides guidance, design principles, tools, and resources to help build, deploy, and manage Azure solutions. The team matches skilled partners for deployment services on request and periodically checks in to ensure that deployment is on track and to help remove blockers.

Azure Migration and Modernization Program (AMMP)

The Azure Migration and Modernization Program (AMMP) provides a mixture of technical skill building, step-by-step guidance, free migration tools, and potential offers to reduce migration costs.

The program uses FastTrack for Azure and Azure solution providers to improve customer success during migration.

Watch this short video to get an overview of how the Azure Migration and Modernization Program (AMMP) can help you.

Solution assessments

Get assistance from a Microsoft solutions assessment expert or qualified partner as part of a Solution Assessment engagement. Microsoft Solution Assessments provide customers with an in depth understanding of the opportunities available in their environments to improve productivity, reduce cost and optimize investments. These assessments utilize modern tools to collect the customer's data estate, analyze the deployed environment and provide insights for data-driven recommendations to help customers determine actionable steps for digital transformations, cloud migrations and process improvement.

Azure support

If you have questions or need help, create a support request. If your support request requires deep technical guidance, visit Azure support plans to align the best plan for your needs.

Shortlist of partner options

During strategy development, it's hard to define specific partnership needs. During development of the cloud adoption plan and skilling plan, those needs will come into focus.

But, based on the culture and maturity of your team, it may be possible to decide on a partnership option that is more aligned with your expected needs.

Choose one or more of the partnership options above to narrow down the options to investigate first.

Begin contract and paperwork reviews

As the shortlist of options is reviewed, there will likely be one or more partners that stand out. If there is a clear leader among the partners, start the process to review contracts and paperwork with the partner.

The contracting process can take time. Reviewing legal terms ahead of time can remove one barrier to engagement when your teams need help the most.

This is especially true if your company requires vendors to be added to an approved vendor list.

Next steps

After your partner alignment strategy is kicked off, you may want to consider your security strategy next.

Define your security strategy

Build a digital transformation timeline

6/8/2021 • 5 minutes to read • Edit Online

Digital transformation and cloud adoption might seem like technical project. But at their core, both are complex change management efforts involving multiple stakeholders moving in alignment to accomplish business and technical change. When they move in unison, the impact on the business is apparent.

Competing objectives

If you've been building out the strategy and plan template for your organization, some trends have likely begun to emerge:

- Competing motivations: Your organization is likely motivated by needs that span several of the motivation categories: Migration motivations, innovation motivations, and critical business events. Each of these categories requires slightly different behaviors during technical implementation and business adoption.
- Competing business outcomes: Your organization is also likely to see competing business outcomes. It's possible to improve customer engagements in the cloud. You could also use the cloud to achieve cost reduction outcomes. But the behaviors required to do each are different, making it difficult to succeed at both in parallel.
- Aligning financial and technical considerations: If motivations and outcomes aren't aligned, it's easy to have misaligned conceptions when evaluating key considerations. This can lead to difficulty executing your strategy or at minimum communicating success.

It's possible to achieve conflicting motivations and competing outcomes. However, your successes will be greater if you prioritize your transformation and develop a timeline that focuses on the most important objectives first.

Constrained staff

Change management of any type places a strain on existing staff. Cloud adoption is no different. Even if the financial and technical considerations look good on paper, competing or parallel outcomes will intensify the strains of change management.

Even with adequate staffing, or ample budget for implementation partners, staffing constraints are a notable risk when outcomes are expected in parallel. There's typically some overlooked shared dependency, which becomes a blocker for one or more projects. Usually, those overlooked dependencies follow Murphy's law and become an issue when they are on the critical path for multiple projects.

If you must deliver parallel outcomes, scrutinize your staffing plans and critical path for any potential shared dependencies before you commit to the strategy.

De-risk with cloud adoption horizons

It's common for organizations to have competing motivations. But competing motivations create a divide in the financial investments that the company can make in any one objective. Such a divide leads to an overall reduction in the amount of change or transformation that any effort can deliver.

Stratification is a sound principle in macroeconomics. But with the limited budget of most technology-driven change management projects, a stratified approach leads to confusing and distracting signals within the program. More noticeably, multiple, simultaneous investments in competing strategies lead to misalignment of the people, processes, and projects that are required for overall program success. For digital transformation to

succeed, organizations must prioritize motivations based on timeline expectations, organizational alignment, and capacity for investment.

To create clarity and alignment, it's suggested that complex digital transformation projects align to an organization's horizons or phased program delivery. In this type of approach, the company commits to a single motivation category for a time-bound period. All teams and organizations prioritize investments and collaborate to support the priority outcome, as needed, for the defined period of time. This approach creates unity, clarity, and drives a snowball effect, allowing the success of one horizon to accelerate the target outcomes of the next horizon.

In this model, the full program team can overcome skills gaps, together. Support one another in achieving small but critical milestones, together. Deliver on those outcomes more quickly, then move on to the next horizon together. In the end, a supportive team-based approach with loosely defined and agile horizons will progress faster than the same teams competing with one another for limited resources.

Example of a cloud adoption plan with four horizons

In this example, a customer has a high number of motivations that appear in each category of the motivations table. Diverse motivations suggest that the organization has multiple critical business events to address, a need for operational improvements from a migration or modernization, and innovation opportunities:

- **Datacenter exit:** They're delivering on a datacenter exit requiring significant focus from the central IT and cloud adoption teams. The 12- to 18-month time frame to fully plan two datacenter replacements is aggressive but realistic, if there are few distractions.
- **Operational improvements:** Accelerating innovation requires modernizing existing operational systems. It also requires modernizing processes that are dedicated to current production environments.
- **Innovation expansion:** The long-term objective is to continue to grow and lead the market through innovation. The ultimate goal is to maximize the amount of effort IT invests in innovation overall and reduce existing operational investments.

This reference customer developed horizons aligned to the following target schedule, to be evaluated and updated as part of quarterly release plan:

HORIZON	OBJECTIVE	TIME FRAME	CONSIDERATIONS
1. Migration and modernization	Prioritize the datacenter exit with a focus on modern platform as a service (PaaS) solutions over a basic lift-and-shift migration.	Months 0 - 18	The migration as priority should minimize conflicts with existing innovation commitments.
2. Operation modernization	Prioritize operational improvements built on cloud-native governance, operations management, security, and compliance capabilities.	Months 6 - 18	This effort complements and supports the primary migration effort.
3. Advanced modernization	With post-migration and operational improvements, the team will have sufficient data and cloud skills to perform deeper modernization of complex architectures.	Months 18-24	

HORIZON	OBJECTIVE	TIME FRAME	CONSIDERATIONS
4. Innovation and growth	Redirect capital reduction from datacenter exits and new skills in central IT to focus on accelerating continued innovation.	Month 24+	All prior horizons will produce a long list of new innovations as the central IT and cloud adoption teams create tighter collaborations and build out automation assets.

Next steps

Developing a similar set of cloud horizons can help refine your technical and financial considerations. This approach could also make it easier to share your priorities and objectives with stakeholders and technical staff to ensure their efforts align with the desired outcomes. With defined cloud adoption horizons and updated considerations, you're now ready to create a business case.

Create a business case

What is a business case?

7/15/2021 • 6 minutes to read • Edit Online

Your organization depends on information technology (IT) for its operations, and probably for creating and supplying its products as well. It's a significant expense. For these reasons, a move to the cloud must be carefully considered and planned.

A business case provides a view of the technical and financial timeline of your environment and can represent the opportunities for reinvestment into further modernization. Developing a business case includes building a financial plan that takes technical considerations into account and aligns with business outcomes. It helps you foster support from your Finance team and other areas of the business, helps accelerate cloud migration, and enables business agility.

Key components of a business case

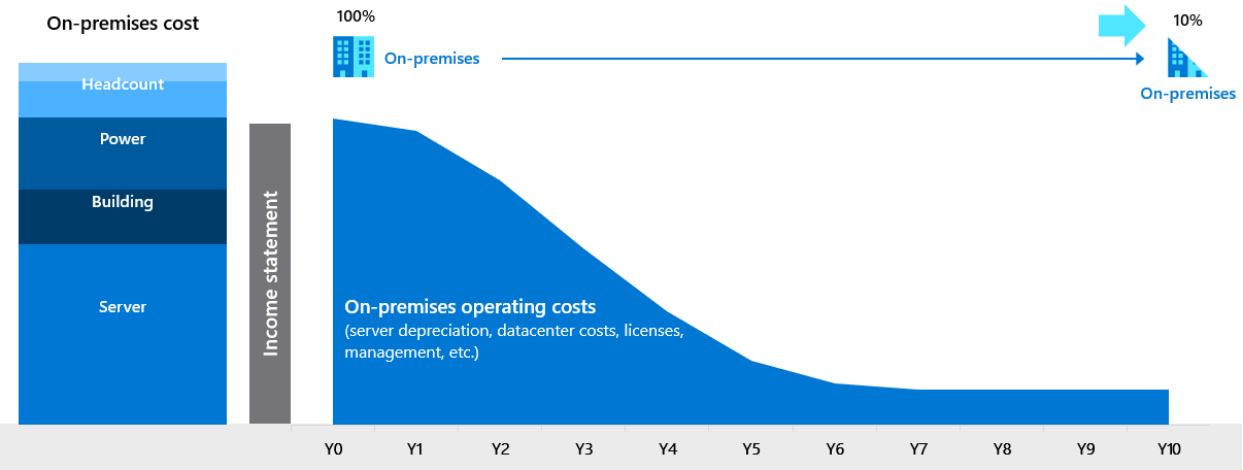
When you're planning your business case to migrate to the cloud, there are several key components to consider.

Environment scope, technical and financial: As you build out the on-premises view of your environment, think about how your environment scope, from both a technical and financial perspective, is aligned. You want to be sure the technical environment you're using for your plan matches up to the financial data.

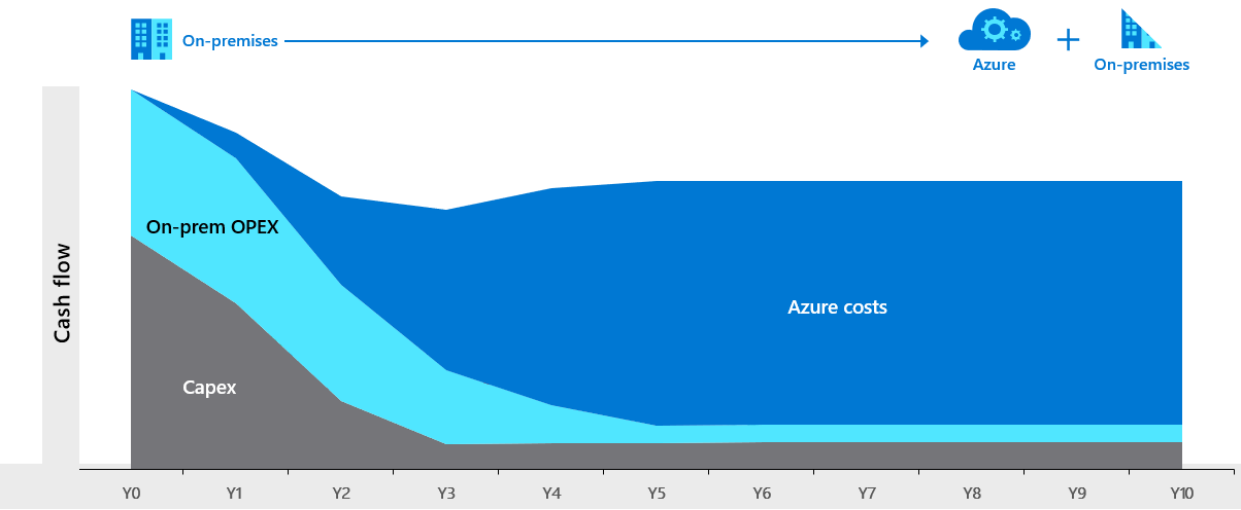
Baseline financial data: Cost to run today: When you build out your business case, it's important to pull your baseline financial data. Common questions you can ask to gather the financial data needed are:

- How much does it cost to run my environment today?
- What am I spending on servers in an average year?
- What am I spending in my data center operations categories, for example, power or lease costs?
- When is the next hardware refresh?

Projections: On-premises costs in on-premises scenario: Forecast what your on-premises costs will be if you don't migrate to the cloud.



Projections: On-premises costs in Azure scenario: Forecast what your on-premises costs will be when you migrate to the cloud in an Azure scenario. It takes resources and time to shift your environment to the cloud, so it's important to account for them in the business case. When you build out the Azure scenario, be sure to take into consideration and include all of the core benefits the cloud provides.

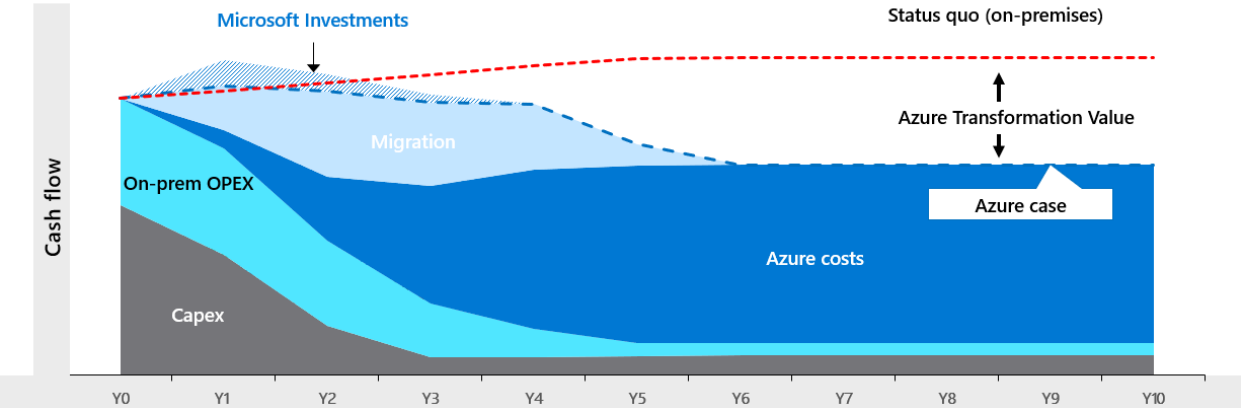


Projections: Migration timeline and Azure costs (optimized): Project what migration timeline and the Azure costs will be with a given environment. Consider how you can optimize and get the most out of your Azure investment. For example, use reserved instances, scale capacity up and down, use the Azure hybrid benefit, and right-size your resources.

A business case isn't just a point-in-time view. It's a plan for a period of time. As you shift to the cloud, you reduce your spend over-time and create a cloud migration plan. You can model out what the ramp-down in spend will be on-premises over time associated with your cloud migration plan.

Once on-premises workloads and cost structure have been identified, you can then build out your optimized Azure consumption plan.

As a final step, when you create your business case, you want to compare the cloud environment to an on-premises or status quo scenario. You can take your Azure view and compare it to your on-premises or status quo scenario with no migration so you can assess the benefit of migrating to the cloud. The Azure view will show on-premises costs that are being reduced over time, your Azure environment costs, and any migration costs associated with shifting to a cloud environment.



Cloud savings provide for future investment

Cloud billing models and offers differ from on-premises however, they can create meaningful savings opportunities you can take advantage of to reduce cloud costs. Saved costs from cloud adoption can be reinvested into iterative modernization. Cloud costs are flexible, and can be reduced with:

Azure Hybrid Benefit: Reduce the costs of running your workloads in the cloud by using this licensing benefit. You can use your on-premises Software Assurance-enabled Windows Server and SQL Server licenses on Azure.

This benefit applies to RedHat and SUSE Linux subscriptions, too. To learn more, see Azure Hybrid Benefit.

Spot virtual machines: You can use spot virtual machines with deep discounts for workloads that can be interrupted and don't need to complete within a specific time frame. For example, high-performance computing scenarios, batch processing jobs, visual rendering applications, dev/test environments including continuous integration and continuous delivery workloads, or large-scale stateless applications. To learn more, see Spot virtual machines.

Reservations: Receive a discount on your workloads when you reserve your resources in advance. In return, Microsoft passes the savings onto you as discounts of up to 72 percent. For more information, see Azure reservations.

Azure Dev/Test pricing: Take advantage of discounted rates for your development and testing, including the Microsoft software charges on Azure Virtual Machines and special dev/test pricing on other services. For more information, see Azure Dev/Test pricing.

Extended security updates: Receive continued support for SQL Server 2008 and SQL Server 2008 R2 in the cloud, which has reached the end of their support lifecycle. You can migrate your on-premises SQL Server instances to Azure Virtual Machines, Azure SQL Database, or stay on-premises and purchase extended security updates. You'll receive free extended security patches by migrating to an Azure Virtual Machine. To learn more, see Lifecycle FAQ: Extended security updates.

Tools

There are many valuable tools and calculators you can use to help prepare a business case for your cloud migration.

Azure Total Cost of Ownership (TCO) Calculator: Use the Total Cost of Ownership (TCO) Calculator online tool to estimate the cost savings you can realize when you migrate your workloads to Azure.

Enter details of your on-premises infrastructure into the tool including servers, databases, storage, and networking, licensing assumptions and costs.

The Calculator creates a match from Azure Services to create a high-level initial TCO comparison. However, the results of the TCO calculator need to be considered with care, since an on-premises server list is often complex and optimization steps can be taken when considering Azure.

Retail Rates Prices API: Use the Retail Rates Prices API to retrieve retail prices for all Azure services. Previously, the only way that you could retrieve prices for Azure services was to either use the Azure Pricing Calculator or use the Azure portal. This API gives you an unauthenticated experience to get retail rates for all Azure services. Use the API to explore prices for Azure services against different regions and different SKUs. The programmatic API can also help you create your own tools for internal analysis and price comparison across SKUs and regions. To learn more, see Retail Rates Prices API.

Azure Pricing Calculator: Use the Azure Pricing Calculator to configure and estimate the costs for Azure products.

Partner toolsets: Microsoft Partners have tools in the Azure Marketplace that can help create a migration cost analysis.

Solution assessments: Get assistance from a Microsoft solutions assessment expert or qualified partner as part of a Solution Assessment engagement.

Azure Migration and Modernization Program: Join the Azure Migration and Modernization Program to get the guidance and expert help you need at every stage of the cloud migration journey. Migrate infrastructure, databases, and apps—and move forward with confidence.

Learn about Azure with Microsoft Learn

Microsoft Learn offers many Azure learning paths that you might want to consider as you build your business case.

- Control Azure spending and manage bills with Azure Cost Management + Billing.
- Microsoft Azure Well-Architected Framework—Cost Optimization
- Plan and manage your Azure costs
- Analyze costs and create budgets with Azure Cost Management
- Save money with Azure Reserved Instances
- Optimize Azure costs with data analysis in Power BI
- Configure and manage costs as a Microsoft partner by using Azure Cost Management

Next steps

Learn more about how to share your strategy and business case to migrate to the cloud:

Share your strategy

Share your strategy

6/8/2021 • 2 minutes to read • Edit Online

After your team has decided on the strategy and roadmap for your cloud adoption plan, you'll want to share your strategy with your organization's finance team and gain their support. Your goal is to showcase the benefits and cost-saving measures of moving services to the cloud from an on-premises plan.

Get started

The information in the following guidance is a reminder of tasks to complete before sharing your strategy for cloud migration with the finance team.

Gather data

Augment your business case with as much data as possible. Personalizing it specifically to your organization gives your finance team a view of how the cloud can benefit your organization. It helps to review your most recent balance sheet, income statement, and cashflow statement.

Identify a focus

What's most important to your organization? What business outcomes do you want to achieve? Cost savings and business agility are some of the topics you might focus on. Customize and align your strategy to your organization's goals, business outcomes, and greatest needs.

Use finance terminology

Use terms and acronyms familiar to the finance team, for example, capital expenditure (CAPEX), operational expenditure (OPEX), and double mortgage period. Using these terms keeps your your business case streamlined and prevents over-explanation. See the finance glossary for detailed descriptions of the most common finance terms.

Share your business case

Show the finance team your business case that includes a financial roadmap and multi-year plan. Your plan should reiterate your organization's business and financial goals, and demonstrate how cloud economics aligns to their financial considerations.

Your business case should be tailored to your organization's goals and business outcomes. For help with building your business case, see Create your business case and Strategy for partner alignment.

Answer questions

After you've shared your strategy and it's likely that the finance team will have questions that you'll need to answer. Some of these questions might be challenging, so it's an effective practice to consider what you might be asked and prepare in advance to answer those questions.

Common goals, drivers, and expected outcomes

Consider the common goals, drivers, and expected outcomes of a Chief Technology Officer (CTO), Chief Information Officer (CIO), Chief Financial Officer (CFO), and the finance team:

	CTO/CIO VIEW	CFO VIEW

	CTO/CIO VIEW	CFO VIEW
Flexibility	Cloud enables dynamic reallocation of IT resources from lower value IT projects to high value projects.	Cloud enables flexible budgetary allocation
Resiliency	More fault-tolerant options for backup, disaster recovery, and security	Lower required cashflow to support disaster recovery and backup
Agility	Higher capability of new technology adoption	New technology adoption without new budgetary allocation
Strategic value	Change IT from a cost center to a profit center. IT personnel evolve into consultants to business units, providing means to innovation, and time to market.	IT as an asset for CFO. New reusable processes, toolsets, scripts, compute images, and ways to increase business agility and innovation
Efficiency	More rapid service delivery to the business units, with focus on driving strategic value.	More granular activity-based costing (ABC), as on-premises cost tracking per workload may be fuzzy. Easier to target the costs per workload or application in the cloud.
Financial benefits	Cloud enables doing more with less, as budgets given to CTO and CIOs are typically constant or shrinking.	Reduced cost. Enables reinvestment opportunities into higher value projects.

Next steps

Learn more about how the cloud can advance your business strategy.

Using cloud economics to advance your business strategy

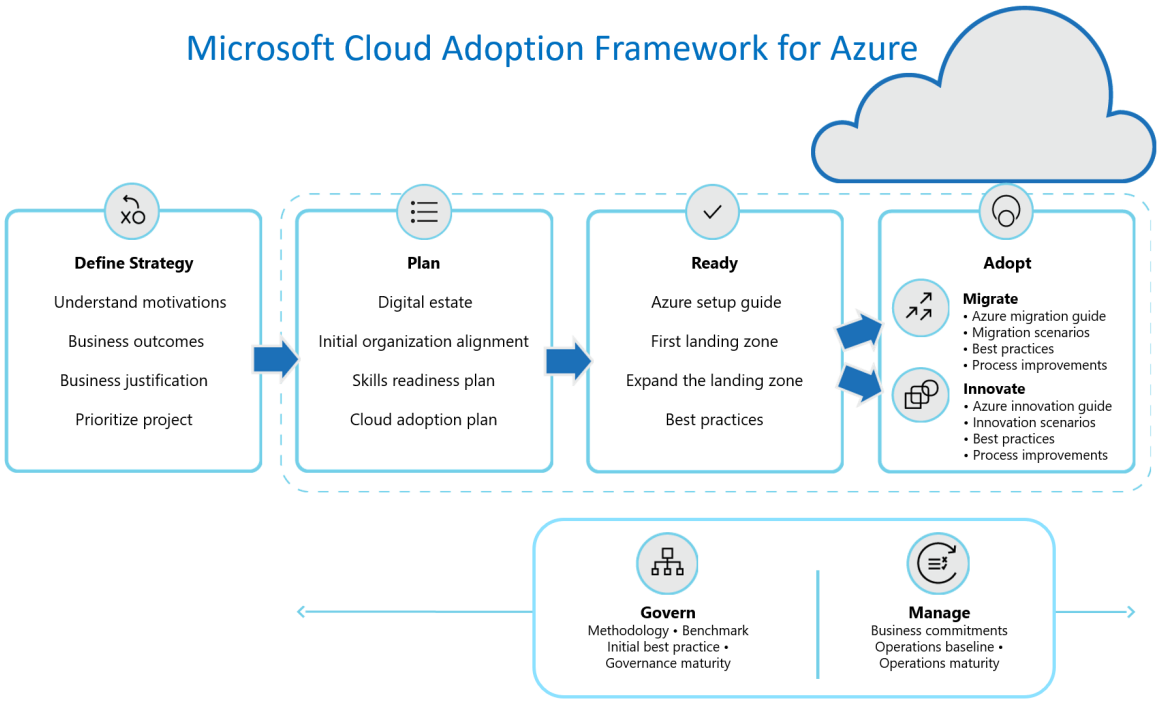
Balance competing priorities

5/18/2021 • 13 minutes to read • Edit Online

Embarking on any digital transformation journey acts like a forcing function for stakeholders, across the business and technology teams. The path to success is firmly rooted in the organization's ability to balance competing priorities.

Similar to other digital transformations, cloud adoption will expose competing priorities throughout the adoption lifecycle. Like other forms of transformation, the ability to find balance in those priorities will have a significant impact on the realization of business value. Balancing these competing priorities will require open and sometimes difficult conversations between stakeholders (and sometimes with individual contributors).

This article outlines some of the competing priorities commonly discussed during the execution of each methodology. We hope this advanced awareness will help better prepare for those discussions when developing your cloud adoption strategy.



The following sections align to the flow of the cloud adoption lifecycle visual above. However, it's important to recognize that cloud adoption is iterative (not a sequential process) and these competing priorities will emerge (and sometimes reemerge) at various points along your cloud adoption journey.

General theme of the Cloud Adoption Framework approach

Monolithic solutions and advanced planning are both built on a series of assumptions that may (or may not) prove to be accurate over time. Adopting the cloud is often a new experience for the business and the technical teams. As with most new experiences or learning opportunities, there is a high probability that those assumptions will be proven false.

Following proven agile principles of delayed technical decisions is the favored approach for most guidance within this framework. That approach follows a consistent pattern: establish a general end-state, move quickly to initial implementation, test and validate assumptions, and refactor early to address assumptions. This type of growth mindset maximizes learning and minimizes risk to business value, but it requires some comfort with ambiguity.

At times, ambiguity can be scarier (or more dangerous) than false assumptions. While this framework leans towards learning and addressing ambiguity during execution, there are a many situations that require the team to lean towards analysis-based or assumption-based approaches. The following sections will attempt to illustrate at least one "expanded scope example" in each section to illustrate times when a second deeper iteration would be valuable.

Balance during the strategizing phase

The core objective of the Strategy methodology is to develop alignment between stakeholders. Once defined, that aligned strategic position will drive behaviors throughout each of the methodologies to ensure that technical decisions align desired business outcomes. Fostering alignment between stakeholders creates a common set competing priorities: **depth of justification** versus **time to business impact**.

Competing priorities:

- **Depth of justification:** Stakeholders often want a deep financial analysis and full business justification to be comfortable aligning to a strategic direction. Unfortunately, that level of analysis may require an extended time period to allow for data collection and analysis.
- **Time to business impact:** Conversely, stakeholders are often held accountable for delivering business outcomes within defined time frames. Time consuming analysis and assessment can put those outcomes at risk before the technical work even begins.

Minimum scope: Finding this balance requires stakeholder discussions early in the process. The Strategy methodology suggests limiting the scope of alignment during this early effort. In the suggested approach, stakeholders focus on aligning around a set of core motivations, measurable outcomes, and a high-level business justification. Stakeholders should then quickly commit to a small number of initial projects or pilots to drive required learning opportunities.

Expanded scope example: If the initial business analysis indicates a high risk of negatively affecting the business, then stakeholders may need to slow down and more cautiously evaluate a deeper analysis during business justification.

Balance during the planning phase

Similar to the priorities during the strategizing phase, there's a need during the planning phase to balance the depth of initial planning versus delayed technical decisions.

Competing priorities:

- **Depth of initial planning** regarding technical implementation in the cloud often contains a high number of assumptions. Especially when the team has skill gaps, the environment suffers from discovery gaps, or the workloads don't have clearly defined architectural end-states. All of these assumptions are common in detailed cloud adoption plans. Experimentation, pilots, and qualitative analysis are required to remove these assumptions.
- **Delayed technical decisions** assume that the later a technical decision can be made, the more accurate that decision will be. Following principles of agile product planning will help delay technical decisions, allowing them to happen at the right time with sufficient information. However, this approach results in a much higher degree of ambiguity in the initial plan.

Minimum scope: Agile product development approaches are suggested to drive prompt action within manageable plans. The Plan methodology recommends the following steps to achieve this balance. Inventory the full digital estate using automated discovery tools, but use incremental rationalization to plan as far as the next 1-3 months of work. Ensure proper organizational alignment to move quickly. Create a skills readiness plan for the assigned team. Use the strategy and plan template to quickly deploy an initial backlog.

Expanded scope example: At times, delivery of a cloud adoption plan may be responding to a time-sensitive or high-impact business event. When success requires the movement of a high-number of assets in a fixed period of time, the steps above are often followed with a deeper planning effort. The key to success in these scenarios, is to plan enough to get going and then plan for the full engagement. This approach reduces the likelihood of planning blocking business outcomes.

Balance during the readiness phase

When adoption teams are preparing for their first steps into the cloud, there are often competing priorities between time to adoption and long-term operations. The team may struggle with being well suited to deliver on the task at hand versus being well managed. This struggle is necessary in traditional IT environments, where the act of developing a platform requires physical assets and acquisition cycles. However, when the entire IT platform is defined in code, traditional development tactics (like refactoring) reduce the need to be well managed from the beginning.

Competing priorities:

- **Long-term operations:** Customers often get blocked by the desire to have a cloud environment that meets feature parity with current operations management, governance, and security systems. In a current study of customers, more than 90% of the customers required support getting past this mindset. This blocker injects months of delay slowing or preventing business impact.
- **Time to adoption:** Cloud-based tools like Azure Policy, Azure Blueprints, and management groups allow for ease of refactoring across the IT platform. Additionally, predefined landing zones provide opinionated positions to accelerate deployment towards an environment that already meets many of the feature parity requirements. Together there are opportunities to accelerate time to market, with minimal impact on long-term operations.

Minimum scope: The Ready methodology outlines a direct path from rapid adoption to long-term operations. This approach starts with a basic introduction to the tools that enable environment refactoring. Based on those tools and environmental requirements, customers are guided to a selection of predefined landing zones (each delivered using infrastructure as code models). That code can then be refactored during the course of cloud adoption to improve operations, security, and management postures.

Expanded scope example: For teams whose adoption plan calls for a mid-term objective (within 24 months) to host **more than 1,000 assets (applications, infrastructure, or data assets) in the cloud**, a more robust view of landing zones is suggested. In these situations, the Govern and Manage methodologies should be considered during initial landing zone conversations. However, this deeper consideration often adds weeks or months to a cloud adoption plan. To minimize impact on business outcomes, the adoption team should pilot actual workloads in the cloud in parallel to the creation of a more mature landing zone and central architecture solution.

Balance during the migration phase

During migration efforts, it's common for adoption teams to assume that workloads will be rehosted in the cloud in their current as-is configuration. This directly competes with a forward-looking view to rearchitect every workload to better take advantage of cloud capabilities. However, the two are not mutually exclusive and can be complimentary when managed through a common process.

Competing priorities:

- **Rehost:** Customers often equate migration to a *lift and shift* motion of replicating all assets to the cloud in their current state configuration. This results in little drift within the IT portfolio. This approach is also the fastest way to retire assets in an existing datacenter.
- **Rearchitect:** Modernizing the architecture of each workload maximizes the value of the cloud across cost, performance, and operations. However, this approach is much slower and often requires access to each

applications' source code.

Minimum scope: During early-stage planning, use the rehost option for planning, with a clear understanding that this option is an initial business assumption and not a technical decision. In the Migrate methodology, the cloud adoption team would then challenge this assumption for each migrated workload. This methodology follows the assess/migrate/promote approach for each workload or group or workloads creating a migration factory. During the assessment phase, the adoption team evaluates technical fit and architecture of each workload. That assessment effort seldom results in a pure lift-and-shift approach because many of the components in the architecture tend to be selected for refactoring and modernization.

Expanded scope example: For mission-critical or high-sensitivity workloads, like a mainframe or multitier microservices application, a deeper assessment of the workload may be required during the assessment phase. In these rearchitecture situations, customers should use the Microsoft Azure Well-Architected Review and the Microsoft Azure Well-Architected Framework to refine workload requirements during the assessment.

Balance during the innovation phase

True customer-facing innovation creates common conflicting priorities between the need to deliver on a planned feature set and a customer empathy development process.

Competing priorities:

- **Feature focus:** Initial plans for innovation build on the existing digital estate and cloud capabilities to deliver a set of features that meet a customer need. It's easy to allow the plan to drive technical implementation, leading to a feature focused development effort. This approach often leads to temporary stakeholder satisfaction but reduces the likelihood of driving innovation that impacts customer behaviors.
- **Customer empathy:** Initial plans are an important part of the business side of development and should be included in regular reporting. However, learning, measuring, and building with customer empathy is a more accurate measure of success in an innovation effort. Focusing on the customer over features is more likely to result in both short-term and long-term customer satisfaction and business impact.

Minimum scope: The Innovate methodology illustrates how to integrate strategy and plans through business value consensus. The guide then introduces cloud-native tools that can accelerate each discipline of innovation, accompanying best practices for implementation. Finally the process improvements section demonstrates approaches to building customer empathy while respecting plans and strategies across the cloud adoption journey. This approach focuses on delivering innovation with the use of as little technology as is possible.

Expanded scope example: At times, an innovation may be dependent on mission-critical or high-sensitivity workloads. When the "customer" is an internal user, the development effort may be both mission-critical and high-sensitivity during the earliest of iterations. For these scenarios, adoption teams should use Microsoft Azure Well-Architected Review and Microsoft Azure Well-Architected Framework to evaluate advanced architectural design early in the process.

Balance during the governance phase

The practice of cloud governance is a constant balance between two competing priorities: speed and agility versus a well-governed environment. The cloud governance team focuses on evaluating and minimizing risks to the business through uniform controls and minimizing change. The adoption team focuses on driving business outcomes, which require new risks and inherently creates change.

Competing priorities:

- **Well-governed:** Every control designed to minimize risk blocks some aspect of change or limits design options. Control is essential to a well-governed environment. However, when controls are designed and deployed in isolation, they can be as damaging as the risks they are intended to prevent.

- **Speed and agility:** Speed and agility are fundamental business requirements in the digital economy. Both require the ability to drive change with minimal blockers to innovation or adoption. When change is driven in isolation of governance, it generates new risks that could harm the business in unintended ways.

Minimum scope: The Govern methodology suggests that neither governance nor adoption should ever happen in isolation. This methodology starts with an understanding of the governance disciplines and a conversation around business risk, policy, and process. As an active member throughout the cloud adoption journey, the governance team can implement a minimum set of guardrails to address the tangible risks within the cloud adoption plan. Over time the governance team can refactor and expand those guardrails to meet new risks. This approach maximizes learning and innovation, while minimizing risk.

Expanded scope example: When the business risk is high, especially early in adoption, the cloud governance team may be required to accelerate the expansion of governance implementations. The same guidance and exercises can be used to add this higher level of governance, but timing may have to be accelerated. In some scenarios, an advanced state of governance may even be required during the deployment of the first landing zones.

Balance during the management phase

The IT business model regarding operations management has been continuously evolving over the last decade. As hardware maintenance moves further from IT's core value proposition, the view on operations management has shifted as well. As IT increases a focus on delivering business value, operations management teams are conflicted with balancing no-ops/low-ops versus broad investments.

Competing priorities:

- **Broad investments:** Investing equally in outage avoidance, rapid recovery, and monitoring across the environment is the traditional approach to operations management. This approach can be costly and sometimes duplicates the supporting products made available by the cloud vendor.
- **No-ops and low-ops:** Use cloud-native operations tools to minimize repetitive and recurring tasks previously delivered by full-time employees. Reducing these operational dependencies in the operations management model frees those employees to drive more value. Alone, this approach can lead to subpar operations support.

Minimum scope: The Manage methodology suggests establishing a cloud-native, no-ops baseline. Acknowledging that the no-ops baseline will not meet all business needs, work with the business to define commitments and better align investments. Expand the baseline to meet common needs for all workloads. Then enable platform teams or specific workload teams to maintain well-managed solutions within a well-managed environment.

Expanded scope example: In most environments, a small percentage of workloads whose business value justifies deep investments in operations from IT. In those scenarios, the IT team might want to use Microsoft Azure Well-Architected Review and Microsoft Azure Well-Architected Framework to guide deeper operations.

Balance during the organization phase

The competing priorities throughout this article are reflective of IT's drive to deliver on business demands for speed and agility. This same shift is showing up in changes to org charts (or virtual team structures) to empower greater support for business outcomes. As IT leaders reflect on team structures, two competing priorities are commonly addressed: centralized control versus delegated control.

Competing priorities:

- **Centralized control:** This operating model focuses on centralization of all controls required to enforce rigid policies. In this model, IT serves as a blocker to innovation, speed, and agility. However, IT can ensure a higher degree of stability, compliance, and security.

- **Delegated control:** In this distributed operating model, it is assumed that each DevOps team or business application team will provide their own set of controls, based on the solutions required to deliver on business objectives. In this model, IT provides guardrails to help keep the teams on the road, but minimizes the number of forced technical constraints whenever possible.

Minimum scope: Most organizations will go through a natural set of evolutions over time. The Organize methodology outlines the most common series of evolutions. The suggested guidance is for teams to strive to move towards a cloud center of excellence (CCoE) structure to deliver delegated control approaches.

Expanded scope example: There are many situations that would trigger a need for centralized control. Third-party compliance requirements and temporary security exposure are two examples of triggers for centralized control. In these situations, there is commonly a need to establish limiting policies and rigid, fixed controls. However, to enable innovation and adoption to continue, it is encouraged that central IT teams deliver those controls based on criticality and sensitivity of each workload. Providing environments with less control but a reduced scope or risk profile, allows for flexibility even when control is required.

Next steps

Learn to balance migration, innovation, and experimentation to maximize the value your cloud migration efforts.

Balance the portfolio

Balance the portfolio

3/6/2021 • 9 minutes to read • Edit Online

Cloud adoption is a portfolio-management effort, cleverly disguised as technical implementation. Like any portfolio management exercise, balancing the portfolio is critical. At a strategic level, this means balancing migration, innovation, and experimentation to get the most out of the cloud. When the cloud adoption effort leans too far in one direction, complexity finds its way into the adoption efforts. This article will guide the reader through approaches to achieve balance in the portfolio.

General scope expansion

Balancing the portfolio is strategic in nature. As such, the approach taken in this article is equally strategic. To ground the strategy in data-driven decisions, this article assumes the reader has evaluated the existing digital estate or has begun that process. The objective of this approach is to aid in evaluating workloads to ensure proper balance across the portfolio through qualitative questions and portfolio refinement.

Document business outcomes

Before balancing the portfolio, it is important to document and share the business outcomes driving the cloud-migration effort. The following table can help document and share desired business outcomes. It's important to note that most businesses are pursuing several outcomes at a time. The importance of this exercise is to clarify the outcomes that are most directly related to the cloud migration effort:

OUTCOME	MEASURED BY	GOAL	TIME FRAME	PRIORITY FOR THIS EFFORT
Reduce IT costs	Datacenter budget	Reduce by $2M USD	12 months	#1
Datacenter exit	Exit from datacenters	2 datacenters	6 months	#2
Increase business agility	Improve time to market	Reduce deployment time by six months	2 years	#3
Improve customer experience	Customer satisfaction (CSAT)	10% improvement	12 months	#4

> **IMPORTANT**
> The above table is a fictional example and should not used to set priorities. In many cases, this table could be considered an antipattern by placing cost savings above customer experiences.

The above table could accurately represent the priorities of the cloud strategy team and the cloud adoption team. Due to short-term constraints, this team is placing a higher emphasis on IT cost reduction and prioritizing a datacenter exit as a means to achieve the desired IT cost reductions. However, by documenting the competing priorities in this table, the cloud adoption team is empowered to help the cloud strategy team identify opportunities to better align implementation of the overarching portfolio strategy.

Move fast while maintaining balance

The guidance regarding incremental rationalization of the digital estate suggests an approach in which the rationalization starts with an unbalanced position. The cloud strategy team should evaluate every workload for compatibility with a rehost approach. Such an approach is suggested because it allows for the rapid evaluation

of a complex digital estate based on quantitative data. Making such an initial assumption allows the cloud adoption team to engage quickly, reducing time to business outcomes. However, as stated in that article, qualitative questions will provide the necessary balance in the portfolio. This article documents the process for creating the promised balance.

Importance of sunset and retire decisions

The table in the documenting business outcomes section above misses a key outcome that would support the number one objective of reducing IT costs. When IT costs reductions rank anywhere in the list of business outcomes, it is important to consider the potential to sunset or retire workloads. In some scenarios, cost savings can come from not migrating workloads that don't warrant a short-term investment. Some customers have reported cost savings in excess of 20% total cost reductions by retiring underutilized workloads.

To balance the portfolio, better reflecting sunset and retire decisions, the cloud strategy team and the cloud adoption team are encouraged to ask the following questions of each workload during assessment and migration phases:

- Has the workload been used by end users in the past six months?
- Is end-user traffic consistent or growing?
- Will this workload be required by the business 12 months from now?

If the answer to any of these questions is "no", then the workload could be a candidate for retirement. If retirement potential is confirmed with the application owner, then it may not make sense to migrate the workload. This prompts for a few qualification questions:

- Can a retirement plan or sunset plan be established for this workload?
- Can this workload be retired prior to the datacenter exit?

If the answer to both of these questions is "yes", then it would be wise to consider **not** migrating the workload. This approach would help meet the objectives of reducing costs and exiting the datacenter.

If the answer to either question is "no", it may be wise to establish a plan for hosting the workload until it can be retired. This plan could include moving the assets to a lower-cost datacenter or alternative datacenter, which would also accomplish the objectives of reducing costs and exiting one datacenter.

Adopt process changes

Balancing the portfolio requires additional qualitative analysis during the Adopt phase, which will help drive simple portfolio rationalization.

Based on the data from the table in the documenting business outcomes section above, there is a likely risk of the portfolio leaning too far into a migration-focused execution model. If customer experience was top priority, an innovation heavy portfolio would be more likely. Neither is right or wrong, but leaning too far in one direction commonly results in diminishing returns, adds unnecessary complexity, and increases execution time related to cloud adoption efforts.

To reduce complexity, you should follow a traditional approach to portfolio rationalization, but in an iterative model. The following steps outline a qualitative model to such an approach:

- The cloud strategy team maintains a prioritized backlog of workloads to be migrated.
- The cloud strategy team and the cloud adoption team host a release planning meeting prior to the completion of each release.
- In the release planning meeting, the teams agree on the top 5 to 10 workloads in the prioritized backlog.
- Outside of the release planning meeting, the cloud adoption team asks the following questions of application owners and subject matter experts:
  - Could this application be replaced with a platform as a service (PaaS) equivalent?

    - Is this application a third-party application?
    - Has budget been approved to invest in ongoing development of the application in the next 12 months?
    - Would additional development of this application improve the customer experience? Create a competitive differentiator? Drive additional revenue for the business?
    - Will the data within this workload contribute to a downstream innovation related to BI, machine learning, IoT, or related technologies?
    - Is the workload compatible with modern application platforms like Azure App Service?
- The answers to the above questions and any other required qualitative analysis would then influence adjustments to the prioritized backlog. These adjustments may include:
    - If a workload could be replaced with a PaaS solution, it may be removed from the migration backlog entirely. At a minimum, additional due diligence to decide between rehost and replace would be added as a task, temporarily reducing that workload's priority from the migration backlog.
    - If a workload is (or should be) undergoing development advancement, then it may best fit into a refactor-rearchitect-rebuild model. Since innovation and migration require different technical skills, applications that align to a refactor-rearchitect-rebuild approach should be managed through an innovation backlog rather than a migration backlog.
    - If a workload is part of a downstream innovation, then it may make sense to refactor the data platform, but leave the application layers as a rehost candidate. Minor refactoring of a workload's data platform can often be addressed in a migration or an innovation backlog. This rationalization outcome may result in more detailed work items in the backlog, but otherwise no change to priorities.
    - If a workload isn't strategic but is compatible with modern, cloud-based application hosting platforms, then it may be wise to perform minor refactoring on the application to deploy it as a modern application. This can contribute to the overall savings by reducing the overall IaaS and OS licensing requirements of the cloud migration.
    - If a workload is a third-party application and that workload's data isn't planned for use in a downstream innovation, then it may be best to leave as a rehost option on the backlog.

These questions shouldn't be the extent of the qualitative analysis completed for each workload, but they help guide a conversation about addressing the complexity of an imbalanced portfolio.

Migration process changes

During migration, portfolio balancing activities can have a negative impact on migration velocity (the speed at which assets are migrated). The following guidance will expand on why and how to align work to avoid interruptions to the migration effort.

Portfolio rationalization requires diversity of technical effort. It is tempting for cloud adoption teams to match that portfolio diversity within migration efforts. Business stakeholders of ask for a single cloud adoption team to address the entire migration backlog. This is seldom an advisable approach, in many cases this can be counter productive.

These diverse efforts should be segmented across two or more cloud adoption teams. Using a two-team model as an example mode of execution, team 1 is the migration team and team 2 is the innovation team. For larger efforts, these teams could be further segmented to address other approaches like replace/PaaS efforts or minor refactoring. The following outlines the skills and roles needed to rehost, refactor, or minor refactoring:

Rehost: Rehost requires team members to implement infrastructure focused changes. Generally using a tool like Azure Site Recovery to migrate VMs or other assets to Azure. This work aligns well to datacenter admins or IT implementors. The cloud migration team is well structured to deliver this work at high scale. This is the fastest approach to migrate existing assets in most scenarios.

Refactor: Refactor requires team members to modify source code, change the architecture of an application, or

adopt new cloud services. Generally this effort would use development tools like Visual Studio and deployment pipeline tools like Azure DevOps to redeploy modernized applications to Azure. This work aligns well to application development roles or DevOps pipeline development roles. The cloud innovation team is best structured to deliver this work. It can take longer to replace existing assets with cloud assets in this approach, but the applications can take advantage of cloud-native features.

Minor refactoring: Some applications can be modernized with minor refactoring at the data or application level. This work requires team members to deploy data to cloud-based data platforms or to make minor configuration changes to the application. This may require limited support for data or application development subject matter experts. However, this work is similar to the work conducted by IT implementors when deploying third-party applications. This work could easily align with the cloud migration team or the cloud strategy team. While this effort is not nearly as fast as a rehost migration, it takes less time to execute than refactor efforts.

During migration, efforts should be segmented in the three ways listed above and executed by the appropriate team in the appropriate iteration. While you should diversify the portfolio, also ensure that efforts stay very focused and segregated.

Next steps

Understand how global market decisions can affect your transformation journey.

Understand global markets

How will global market decisions affect the transformation journey?

3/6/2021 • 2 minutes to read • Edit Online

The cloud opens new opportunities to perform on a global scale. Barriers to global operations are significantly reduced, by empowering companies to deploy assets in market, without the need to invest heavily in new datacenters. Unfortunately, this also adds a great deal of complexity from technical and legal perspectives.

Data sovereignty

Many geopolitical regions have established data sovereignty regulations. Those regulations restrict where data can be stored, what data can leave the country of origin, and what data can be collected about citizens of that region. Before operating any cloud-based solution in a foreign geography, you should understand how that cloud provider handles data sovereignty. For more information about Azure's approach for each geography, see Azure geographies. For more information about compliance in Azure, see Privacy at Microsoft in the Microsoft Trust Center.

The remainder of this article assumes legal counsel has reviewed and approved operations in a foreign country.

Business units

It is important to understand which business units operate in foreign countries, and which countries are affected. This information will be used to design solutions for hosting, billing, and deployments to the cloud provider.

Employee usage patterns

It is important to understand how global users access applications that are not hosted in the same country as the user. Global wide-area networks (WANs) route users based on existing networking agreements. In a traditional on-premises world, some constraints limit WAN design. Those constraints can lead to poor user experiences if not properly understood before cloud adoption.

In a cloud model, commodity internet opens up many new options as well. Communicating the spread of employees across multiple geographies can help the cloud adoption team design WAN solutions that create positive user experiences **and** potential reduce networking costs.

External user usage patterns

It is equally important to understand the usage patterns of external users, like customers or partners. Much like employee usage patterns, external user usage patterns can negatively affect performance of cloud deployments. When a large or mission-critical user base resides in a foreign country, it could be wise to include a global deployment strategy into the overall solution design.

Next steps

Learn about the skills needed during the strategizing phase of your cloud adoption journey.

Skills needed during the strategizing phase

Define a security strategy

5/24/2021 • 23 minutes to read • Edit Online

The ultimate objectives for a security organization don't change with adoption of cloud services, but how those objectives are achieved will change. Security teams must still focus on reducing business risk from attacks and work to get confidentiality, integrity, and availability assurances built into all information systems and data.

Modernize your security strategy

Security teams need to modernize strategies, architectures, and technology as the organization adopts cloud and operates it over time. While the size and number of changes can initially seem daunting, the modernization of the security program allows security to shed some painful burdens associated with legacy approaches. An organization may temporarily operate with legacy strategy and tooling, but this approach is difficult to sustain with the pace of change in cloud and the threat environment:

- Security teams are likely to be left out of cloud adoption decision making if they take a legacy mindset of "arms-length" security where the answer always starts with "no" (instead of working together with IT and business teams to reduce risk while enabling the business).
- Security teams will have a difficult time detecting and defending against cloud attacks if they use only legacy on-premises tooling and exclusively adhere to network perimeter only doctrine for all defenses and monitoring. Defending at cloud scale mandates the use of cloud native detection and automation capabilities and the introduction of an identity perimeter to help monitor and protect cloud and mobile assets. For more information, consult these references:
  - The Microsoft identity platform helps you incorporate modern authentication and authorization mechanisms into your applications.
  - Azure Sentinel provides cloud native security analytics and threat intelligence across your organization.

Because this transformation can be significant, we recommend that security teams take an agile approach to modernizing security that rapidly modernizes the most critical aspects of the strategy and then continuously improve incrementally afterwards.

Security of the cloud and from the cloud

As your organization adopts cloud services, security teams will work toward two main objectives:

- **Security *of* the cloud (securing cloud resources):** Security should be integrated into the planning and operation of cloud services to ensure that those core security assurances are consistently applied across all resources.
- **Security *from* the cloud (using the cloud to transform security):** Security should immediately start planning and thinking about how to use cloud technologies to modernize security tools and processes, particularly natively integrated security tools. Increasingly, security tools are being hosted in the cloud and providing capabilities that are difficult or impossible to do in an on-premises environment.

Many organizations start by treating cloud resources as another *virtual datacenter*, an effective starting point for security of the cloud. As organizations modernize using security from the cloud, most will find themselves quickly outgrowing this model of thinking. Securing a software-defined datacenter using cloud-hosted tools enables capabilities beyond what on-premises models can offer:

- Rapid enablement and scaling of security capabilities.
- Highly effective asset inventory and security configuration hygiene discovery.

- Continuous assessment of the organization's security posture and controls, by deploying for example Azure Security Center.
- Vastly improved threat detection that uses vast repositories of threat intelligence and the nearly unlimited processing and storage capabilities of the cloud, such as that enabled by Azure Sentinel.

The right level of security friction

Security naturally creates friction that slows down processes, it's critical to identifying which elements are healthy in your DevOps and IT processes and which are not:

- **Healthy friction:** Much like the resistance in exercise makes a muscle stronger, integrating the right level of security friction strengthens the system or application by forcing critical thinking at the right time. This typically takes the form of considering how and why an attacker may try to compromise an application or system during design (known as threat modeling), and reviewing, identifying, and ideally fixing potential vulnerabilities an attacker can exploit in software code, configurations, or operational practices.
- **Unhealthy friction:** Impedes more value than it protects. This often happens when security bugs generated by tools have a high false positive rate (such as false alarms) or when the effort to discover or fix security issues far exceeds the potential impact of an attack.

Standalone and integrated responsibilities

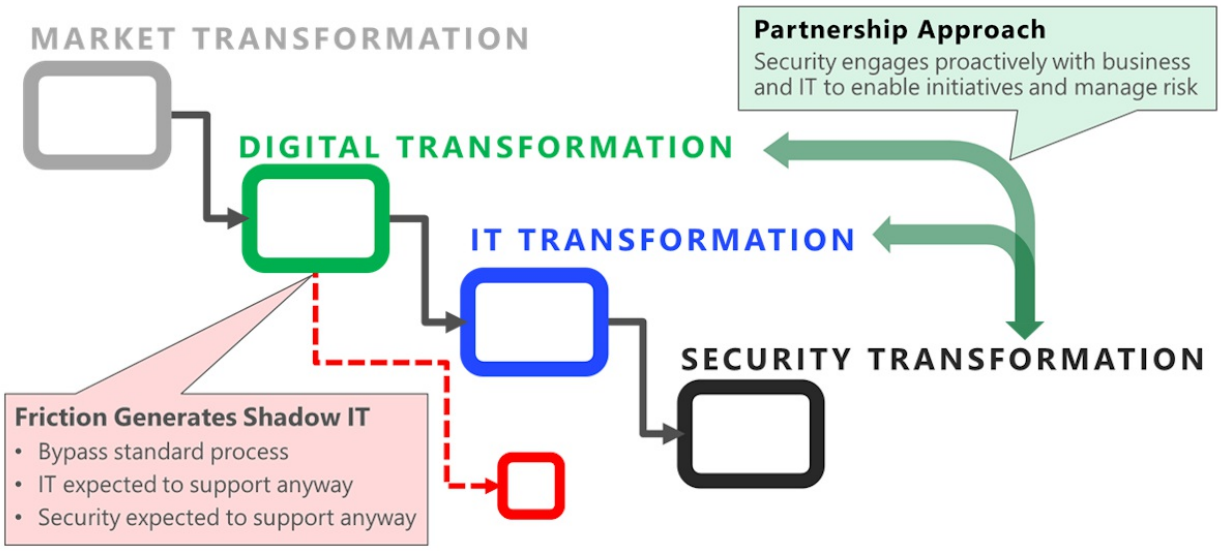
Providing confidentiality, integrity, and availability assurances requires security experts to operate dedicated security functions and work closely with other teams in the organization:

- **Unique security functions:** Security teams perform independent functions that are not found elsewhere in the organization, such as security operations, vulnerability management (such as for virtual machines, containers), and other functions.
- **Integrating security into other functions:** Security teams also serve as subject matter experts to other teams and functions in the organization who are driving business initiatives, assessing risk, designing or developing applications, and operating IT systems. Security teams advise these teams with expertise and context on attackers, attack methods and trends, vulnerabilities that could allow unauthorized access, and options for mitigation steps or workarounds and their potential benefits or pitfalls. This function of security resembles that of a quality function as it will be woven into many places large and small in support of a single outcome.

Executing on these responsibilities while keeping up with the rapid pace of change in the cloud and the transformation of business requires security teams to modernize their tools, technologies, and processes.

Transformations, mindsets, and expectations

Many organizations are managing a chain of multiple simultaneous transformations in the organization. These internal transformations typically start because nearly all external markets are transforming to meet new customer preferences for mobile and cloud technologies. Organizations often face the competitive threat of new startups and the digital transformation of traditional competitors who can disrupt the market.



The internal transformation process typically includes:

- **Digital transformation** of the business to capture new opportunities and stay competitive against digital native startups.
- **Technology transformation** of the IT organization to support the initiative with cloud services, modernized development practices, and related changes.
- **Security transformation** to both adapt to the cloud and simultaneously address an increasingly sophisticated threat environment.

Internal conflict can be costly

Change creates stress and conflict, which can grind decision making to a halt. This is especially true in security where accountability for security risk is often misplaced on the subject matter experts (security teams), rather than on the owners of the assets (business owners) that are accountable for business outcomes and all other risk types. This misplaced accountability often happens because all stakeholders incorrectly view security as a technical or absolute problem to be solved, rather than a dynamic ongoing risk like corporate espionage and other traditional criminal activities.

During this time of transformation, leadership of all teams must work actively to reduce conflict that can both derail critical projects and incentivize teams to bypass security risk mitigation. Internecine conflict between teams can result in:

- **Increased security risk** such as avoidable security incidents or increased business damage from attacks (particularly when teams get frustrated by security and bypass normal processes or when outdated security approaches are easily bypassed by attackers).
- **Negative impact on the business or mission** such as when business processes aren't enabled or updated fast enough to meet market needs (often when security processes hold up key business initiatives).

It's critical to stay aware of relationship health within and between teams to help them navigate the shifting landscape that could leave valuable team members insecure and unsettled. Patience, empathy, and education on these mindsets and the positive potential of the future will help your teams better navigate this period, driving good security outcomes for the organization.

Leaders can help drive culture changes with concrete proactive steps like:

- Publicly modeling the behavior they expect of their teams.
- Being transparent about the challenges of the changes, including highlighting their own struggles to adapt.
- Regularly reminding teams of the urgency and importance of modernizing and integrating security.

Cybersecurity resilience

Many classic security strategies have been focused solely on preventing attacks, an approach that is insufficient

for modern threats. Security teams must ensure their strategy goes beyond this and also enables rapid attack detection, response, and recovery to increase resilience. Organizations must assume that attackers will compromise some resources (sometimes called *assume breach*) and work to ensure that resources and technical designs are balanced between attack prevention and attack management (rather than the typical default approach of only attempting to prevent attacks).

Many organizations are already on this journey because they have been managing the steady rise in volume and sophistication of attacks in recent years. This journey often starts with the first major incident, which can be an emotional event where people lose their prior sense of invulnerability and safety. While not as severe as a loss of life, this event can trigger similar emotions starting with denial and ultimately ending in acceptance. This assumption of "failure" may be difficult for some to accept at first, but it has strong parallels to the well-established "fail-safe" engineering principle and the assumption allows your teams to focus on a better definition of success: resilience.

The functions of the NIST cybersecurity framework serve as a useful guide on how to balance investments between the complementary activities of identify, protect, detect, respond, and recover in a resilient strategy.

More on cybersecurity resilience and the ultimate goals of cybersecurity controls is discussed in How do you keep your organization's risk down.

How the cloud is changing security

Shifting to the cloud for security is more than a simple technology change, it's a generational shift in technology akin to moving from mainframes to desktops and onto enterprise servers. Successfully navigating this change requires fundamental shifts in expectations and mindset by security teams. Adopting the right mindsets and expectations reduces conflict within your organization and increases the effectiveness of security teams.

While these could be part of any security modernization plan, the rapid pace of change in the cloud makes adopting them an urgent priority.

- **Partnership with shared goals.** In this age of fast paced decisions and constant process evolution, security can no longer adopt an "arms-length" approach to approving or denying changes to the environment. Security teams must partner closely with business and IT teams to establish shared goals around productivity, reliability, and security and work collectively with those partners to achieve them.

  This partnership is the ultimate form of "shift left"—the principle of integrating security earlier in the processes to make fixing security issues easier and more effective. This requires a culture change by all involved (security, business, and IT), requiring each to learn the culture and norms of other groups while simultaneously teaching others about their own.

  Security teams must:

  - **Learn** the business and IT objectives and why each is important and how they are thinking about achieving them as they transform.
  - **Share** why security is important in the context of those business goals and risks, what other teams can do to meet security goals, and how they should do it.

  While not an easy task, it's essential for sustainably securing the organization and its assets. This partnership will likely result in healthy compromises where only the minimum security, business, and reliability goals may be met initially, but incrementally improve steadily over time.

- **Security is an ongoing risk, not a problem.** You can't "solve" crime. At its core, security is just a risk management discipline, which happens to be focused on malicious actions by humans rather than natural events. Like all risks, security is not a problem that can be fixed by a solution, it's a combination of the likelihood and impact of damage from a negative event, an attack. It's most comparable to traditional corporate espionage and criminal activities where organizations face motivated human attackers who have financial incentive to successfully attack the organization.

- **Success in either productivity or security requires both.** An organization must focus on both security and productivity in today's "innovation or become irrelevant" environment. If the organization is not productive and driving new innovation, it may lose competitiveness in the marketplace that causes it to weaken financially or eventually fail. If the organization is not secure and loses control of assets to attackers, it may lose competitiveness in the marketplace that causes it to weaken financially and eventually fail.

- **Nobody's perfect.** No organization is perfect at adopting the cloud, not even Microsoft. Microsoft's IT and security teams grapple with many of the same challenges that our customers do such as figuring out how to structure programs well, balancing supporting legacy software with supporting cutting edge innovation, and even technology gaps in cloud services. As these teams learn how to better operate and secure the cloud, they are actively sharing their lessons learned via documents like this along with others on the IT showcase site, while continuously providing feedback to our engineering teams and third-party vendors to improve their offerings.

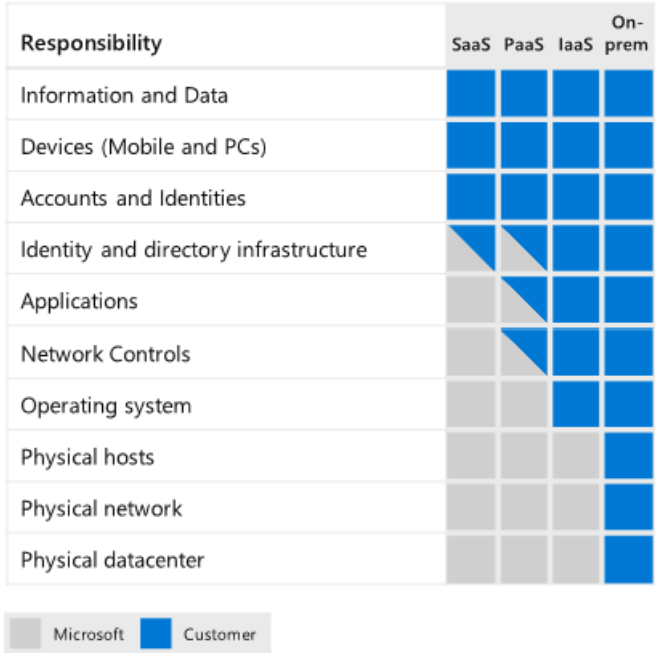
  Based on our experience, we recommend that teams are held to a standard of continuous learning and improvement rather than a standard of perfection.

- **Opportunity in transformation.** It's important to view digital transformation as a positive opportunity for security. While it's easy to see the potential downsides and risk of this change, it's easy to miss the massive opportunity to reinvent the role of security and earn a seat at the table where decisions are made. Partnering with the business can result in increased security funding, reduce wasteful repetitive efforts in security, and make working in security more enjoyable as they will be more connected to the organization's mission.

Adopting the shared responsibility model

Hosting IT services in the cloud splits the operational and security responsibilities for workloads between the cloud provider and the customer tenant, creating a de facto partnership with shared responsibilities. All security teams must study and understand this shared responsibility model to adapt their processes, tools, and skill sets to the new world. This will help avoid inadvertently creating gaps or overlaps in your security posture resulting in security risks or wasted resources.

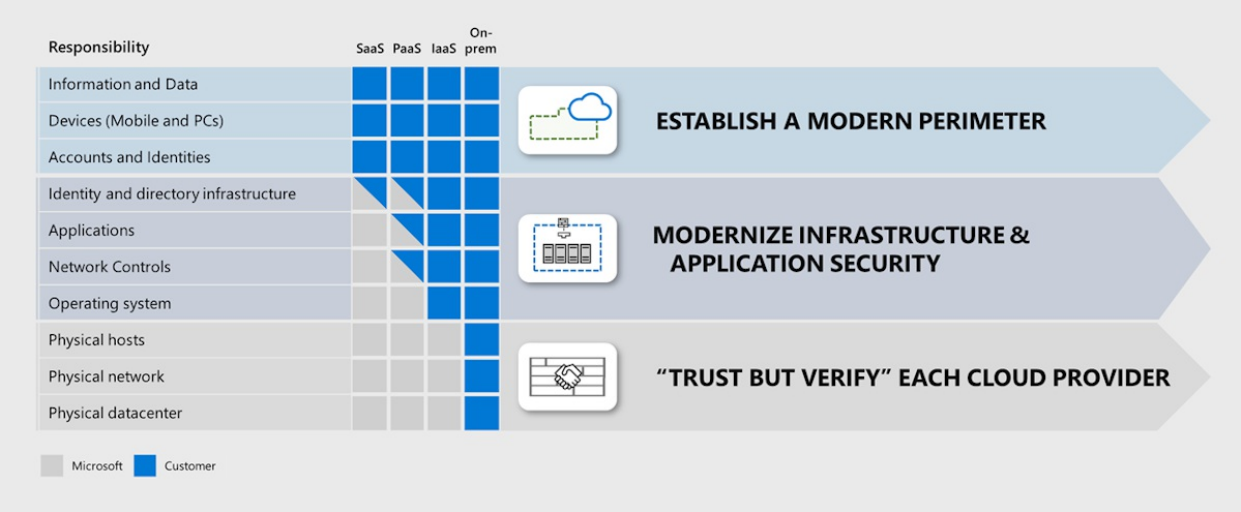
This diagram illustrates how security responsibilities will be distributed between cloud vendors and cloud customer organizations in a de facto partnership:



As there are different models of cloud services, the responsibilities for each workload will vary depending on whether it's hosted on software as a service (SaaS), platform as a service (PaaS), infrastructure as a service (IaaS), or in an on-premises datacenter.

Building security initiatives

This diagram illustrates the three primary security initiatives that most security programs should follow to adjust their security strategy and security program goals for the cloud:



Building a resilient security posture in the cloud requires several parallel complementary approaches:

- **Trust but verify:** For responsibilities performed by the cloud provider, organizations should take a "trust but verify" approach. Organizations should evaluate the security practices of their cloud providers and the security controls they offer to ensure the cloud provider meets the security needs of the organization.
- **Modernize infrastructure and application security:** For technical elements under the organization's control, prioritize modernizing security tooling and associated skill sets to minimize coverage gaps for securing resources in the cloud. This is composed of two different complementary efforts:
  - **Infrastructure security:** Organizations should use the cloud to modernize their approach to protecting and monitoring the common components used by many applications, such as operating systems, networks, and container infrastructure. These cloud capabilities can often including managing infrastructure components across both IaaS and on-premises environments. Optimizing this strategy is important because this infrastructure is a dependency of the applications and data that run on it, which often enable critical business processes and store critical business data.
  - **Application security:** Organizations should also modernize the way they secure the unique applications and technology that is developed by or for their organization. This discipline is changing rapidly with the adoption of agile DevOps processes, the increasing use of open-source components, and introduction of cloud APIs and cloud services to replace application components or interconnect applications.

    Getting this right is critical because these applications often enable critical business processes and store critical business data.
  - **Modern perimeter:** Organizations should have a comprehensive approach for protecting data across all workloads, organizations should establish a modern perimeter of consistent, centrally managed identity controls to protect their data, devices, and accounts. This is heavily influenced by a zero trust strategy discussed in detail in Module 3 of the CISO workshop.

Security and trust

The use of the word *trust* in security can be confusing. This documentation refers to it in two ways that illustrate useful applications of this concept:

- Zero trust is a common industry term for a strategic approach to security that assumes a corporate or intranet network is hostile (worthy of *zero trust*) and designs security accordingly.
- Trust but verify is an expression that captures the essence of two different organizations working together toward a common goal despite having some other potentially divergent interests. This concisely captures many of the nuances of the early stages of partnering with a commercial cloud provider for organizations.

A cloud provider and its practices and processes can be accountable to meet contractual and regulatory requirements and could earn or lose trust. A network is a nonliving connection that cannot face consequences if it's used by attackers (much like you cannot hold a road or a car accountable for criminals using them).

How cloud is changing security relationships and responsibilities

As with previous transitions to a new generation of technology like desktop computing and enterprise server computing, the shift to cloud computing is disrupting long-established relationships, roles, responsibilities, and skill sets. The job descriptions we have become accustomed to over the last few decades do not cleanly map to an enterprise that now includes cloud capabilities. As the industry collectively works to normalize a new model, organizations will have to focus on providing as much clarify as possible to help manage the uncertainty of ambiguity during this period of change.

Security teams are affected by these changes in the business and technology they support as well as their own internal modernization efforts to better orient to threat actors. Attackers are actively evolving to constantly search for the easiest weak points to exploit in the people, process, and technology of the organization and security must develop capabilities and skills to address these angles.

This section describes the key relationships that frequently change on the journey to the cloud, including lessons learned on minimizing risk and embracing the opportunities to improve:

- **Between security and business stakeholders:** Security leadership will need to increasingly partner with business leaders to enable organizations to reduce risk. Security leaders should support business decision making as security subject matter expert (SMEs) and should strive to grow into trusted advisors to these business leaders. This relationship will help ensure business leaders consider security risks while making business decisions, inform security of business priorities, and help ensure security investments are prioritized appropriately alongside other investments.

- **Between security leadership and team members:** Security leadership should take these insights from business leadership back to their teams to guide their investment priorities.

  By setting a tone of cooperation with business leaders and their teams rather than a classic "arms-length" relationship, security leaders can avoid an adversarial dynamic that impedes both security and productivity goals.

  Security leaders should strive to provide clarity to their team on how to manage their daily decisions on productivity and security tradeoffs as this may be new to many on their teams.

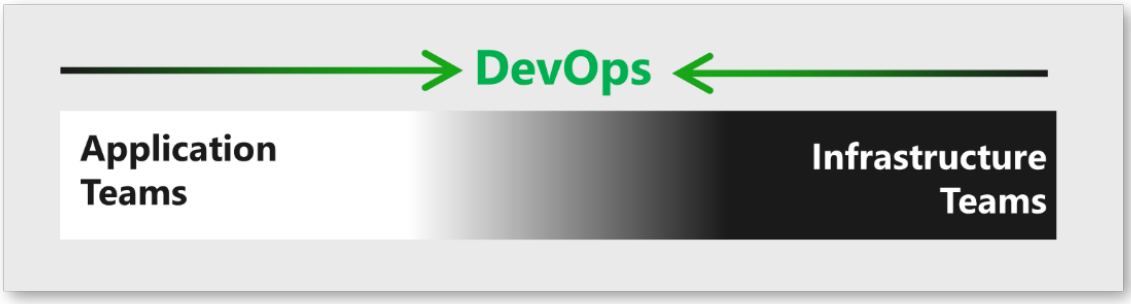
- **Between application and infrastructure teams (and cloud providers):** This relationship is undergoing significant changes because of multiple trends in the IT and security industry aimed at increasing innovation speed and developer productivity.

  The old norms and organizational functions have been disrupted, but new norms and functions are still emerging, so we recommend accepting the ambiguity, keeping up with current thinking, and experiment with what works for your organizations until it does. We don't recommend adopting a wait-and-see approach in this space because it might put your organization at a major competitive disadvantage.

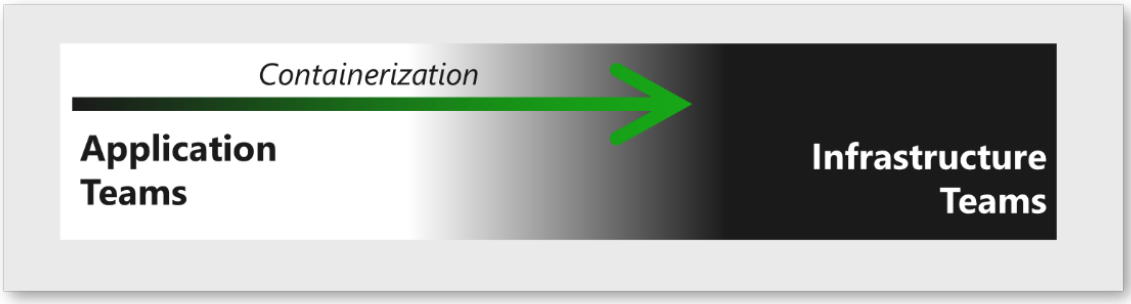
  These trends are challenging the traditional norms for roles and relationships of applications and

infrastructure:

- **DevOps-fusing disciplines:** In its ideal state, this effectively creates a single highly functional team that combines both sets of subject matter expertise together to rapidly innovate, release updates, and resolve issues (security and otherwise). While this ideal state will take some time to achieve and the responsibilities in the middle are still ambiguous, organizations are already reaping some benefits of rapid releases because of this cooperative approach. Microsoft recommends integrating security into this cycle to help learn those cultures, share security learnings, and work toward a common goal of rapidly releasing secure and reliable applications.



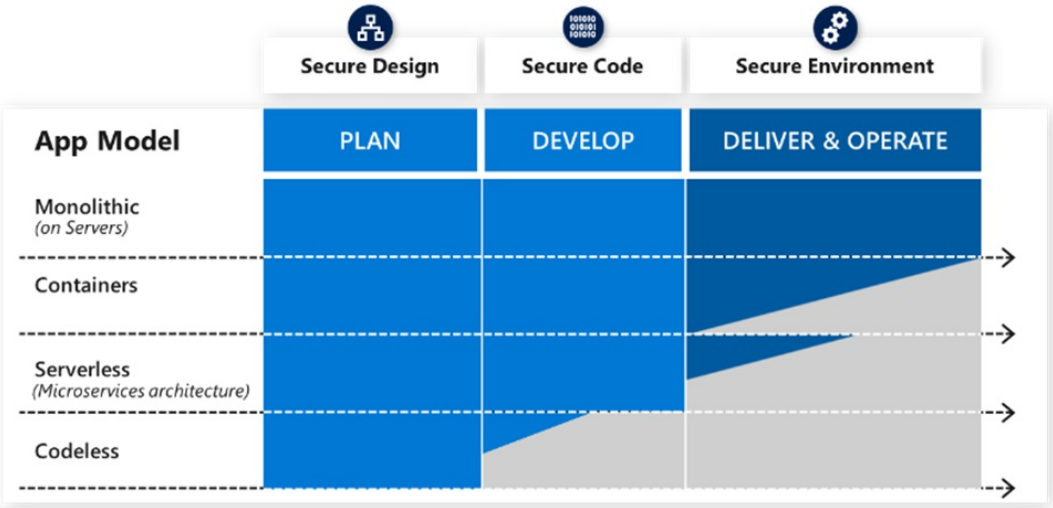
- **Containerization becoming a common infrastructure component:** Applications are increasingly being hosted and orchestrated by technologies like Docker, Kubernetes, and similar technologies. These technologies simplify development and release by abstracting many elements of the setup and configuration of the underlying operating system.



While containers began as an application development technology managed by development teams, it's becoming a common infrastructure component that is increasingly shifting to infrastructure teams. This transition is still in progress at many organizations, but it's a natural and positive direction many of the current challenges will be best solved with traditional infrastructure skill sets like networking, storage, and capacity management.

Infrastructure teams and security team members that support them should be provided with training, processes, and tooling to help manage, monitor, and secure this technology.

- **Serverless and cloud application services:** One of the dominant trends in industry right now is reducing the amount of time and development work required to build or update applications.



Developers are also increasingly using cloud services to:

- Run code instead of hosting applications on virtual machines (VMs) and servers.
- Provide application functions instead of developing their own components. This has led to a *serverless* model that uses existing cloud services for common functions. The number and variety of cloud services (and their pace of innovation) has also exceeded the ability of security teams to evaluate and approve the use of those services, leaving them to choose between allowing developers to use any service, attempting to prevent the development teams from using unapproved services, or trying to find a better way.
- **Codeless applications and Power Apps:** Another emerging trend is the use of codeless technologies like Microsoft Power Apps. This technology enables people without coding skills to create applications that achieve business outcomes. Because of this low friction and high value potential, this trend has the potential to rise in popularity quickly and security professionals would be wise to rapidly understand its implications. Security efforts should be focused on the areas where a human could make a mistake in the application, namely the design of the application and asset permissions via threat modeling the application components, interactions/relationships, and role permissions.

- **Between developers and open-source component authors:** Developers are also increasing efficiency by using open-source components and libraries instead of developing their own components. This brings value through efficiency, but also introduces security risks by creating an external dependency and a requirement to properly maintain and patch those components. Developers are effectively assuming the risk of security and other bugs when they use these components and have to ensure there is a plan to mitigate them at the same standards as code they would develop.

- **Between applications and data:** The line between security of data and applications is becoming blurred in places and new regulations are creating a need for closer cooperation between data/privacy teams and security teams:

  - **Machine learning algorithms:** Machine learning algorithms are similar to applications in that they are designed to process data to create an outcome. The key differences are:

    - **High-value machine learning:** Machine learning often confers a significant competitive advantage and is often considered sensitive intellectual property and a trade secret.

    - **Sensitivity imprint:** Supervised machine learning is tuned using data sets, which imprints characteristics of the dataset on the algorithm. Because of this, the tuned algorithm may be considered sensitive because of the dataset used to train it. For example, training a machine learning algorithm to find secret army bases on a map using a dataset of secret army bases would make it a sensitive asset.

> **NOTE**
>
> Not all examples are obvious, so it's critical to bring a team together with the right stakeholders from data science teams, business stakeholders, security teams, privacy teams, and others. These teams should have a responsibility to meet common goals of innovation and responsibility. They should address common issues such as how and where to store copies of data in insecure configurations, how to classify algorithms, as well as any concerns of your organizations.
>
> Microsoft has published our principles of responsible AI to guide our own teams and our customers.

    - **Data ownership and privacy:** Regulations like GDPR have increased the visibility of data issues and applications. Application teams now have the ability to control, protect, and track sensitive data at a level comparable to tracking financial data by banks and financial institutions. Data owners and applications teams need to build a rich understanding of what data applications store and what controls are required.

- **Between organizations and cloud providers:** As organizations host workloads in the cloud, they are entering into a business relationship with each of those cloud providers. The use of cloud services often brings business value such as:

    - **Accelerating digital transformation initiatives** by reducing time to market for new capabilities.

    - **Increasing value of IT and security activities** by freeing teams to focus on higher value (business-aligned) activities rather than lower-level commodity tasks that are provided more efficiently by cloud services on their behalf.

    - **Increased reliability and responsiveness:** Most modern clouds also have high uptime compared to traditional on-premises datacenters and have shown they can scale rapidly (such as during the COVID-19 pandemic) and provide resiliency following natural events like lightning strikes (which would have kept many on-premises equivalents down for much longer).

      While beneficial, this shift to the cloud is not without risk. As organizations adopt cloud services, they should consider potential risk areas including:

    - **Business continuity and disaster recovery:** Is the cloud provider financially healthy with a business model that's likely to survive and thrive during your organization's use of the service? Has the cloud provider made provisions to allow customer continuity if the provider experiences financial or other failure, such as providing their source code to customers or open-sourcing it?

      For more information and documents regarding Microsoft's financial health, see Microsoft investor relations.

    - **Security:** Does the cloud provider follow industry best practices for security? Has this been validated by independent regulatory bodies?

        - Microsoft Cloud App Security allows you to discover usage of over 16,000 cloud applications, which are ranked and scored based on more than 70 risk factors to provide you with ongoing visibility into cloud use, shadow IT, and the risk that shadow IT poses to your organization.
        - The Microsoft Service Trust Portal makes regulatory compliance certifications, audit reports, pen tests, and more available to customers. These documents include many details of internal security practices (notably the SOC 2 Type 2 report and FedRAMP Moderate system security plan). Azure Security Center allows the management of security policies and can indicate the level of compliance with predefined industry and regulatory standards.

    - **Business competitor:** Is the cloud provider a significant business competitor in your industry? Do you have sufficient protections in the cloud services contract or other means to protect your

business against potentially hostile actions?

Review this article for commentary on how Microsoft avoids competing with cloud customers.

- **Multicloud:** Many organizations have a de facto or intentional multicloud strategy. This could be an intentional objective to reduce reliance on a single supplier or to access unique best of breed capabilities, but can also happen because developers chose preferred or familiar cloud services, or your organization acquired another business. Regardless of the reason, this strategy can introduce potential risks and costs that have to be managed including:
  - **Downtime from multiple dependencies:** Systems architected to rely on multiple clouds are exposed to more sources of downtime risk as disruptions in the cloud providers (or your team's use of them) could cause an outage/disruption of your business. This increased system complexity would also increase the likelihood of disruption events as team members are less likely to fully understand a more complex system.
  - **Negotiating power:** Larger organizations also should consider whether a single-cloud (mutual commitment/partnership) or multicloud strategy (ability to shift business) will achieve greater influence over their cloud providers to get their organization's feature requests prioritized.
  - **Increased maintenance overhead:** IT and security resources already are overburdened from their existing workloads and keeping up with the changes of a single cloud platform. Each additional platform further increases this overhead and takes team members away from higher value activities like streamlining technical process to speed business innovation, consulting with business groups on more effective use of technologies, and so on.
  - **Staffing and training:** Organizations often do not consider the staffing requirements necessary to support multiple platforms and the training required to maintain knowledge and currency of new features which are released in a rapid pace.

Cloud monitoring guide: Formulate a monitoring strategy

8/9/2021 • 20 minutes to read • Edit Online

As you undergo your digital transformation to the cloud, it's important that you plan and develop an effective cloud monitoring strategy with participation of developers, operations staff, and infrastructure engineers. The strategy should be growth-oriented, defined minimally, then refined iteratively; always aligned with business needs. Its outcome delivers an agile operations modality centered around the ability of the organization to proactively monitor complex distributed applications the business depends on.

Where to start?

To ease your journey to the cloud, use the strategizing and the planning phase of the Cloud Adoption Framework. Monitoring influences and justifies the motivations, business outcomes, and initiatives. Include monitoring during strategizing and planning phases, your initiatives, and projects. For example, examine how the first adoption project establishes early operations management in Azure. Imagine what the cloud operating model needs to look like, including the role of monitoring. Monitoring is best served with a service-based approach, as an operations function, where monitoring is an advisory service and a provider of expertise to business and IT consumers.

The following are important areas that strongly influence a sound monitoring strategy:

- Monitor the health of your applications, based on its components and their relationship with other dependencies. Start with the cloud service platform, resources, the network, and lastly the application by collecting metrics and logs where applicable. For the hybrid cloud model, include on-premises infrastructure and other systems the application relies on.
- Include measuring the end user's experience in your applications performance monitoring plan by mimicking your customer's typical interactions with the application.
- Ensure security requirements correspond with your organizations security compliance policy.
- Align alerts with what is considered a relevant and practical incident (such as warnings and exceptions) and align severity with its significance following your incident priority and urgency escalation matrix.
- Collect only the metrics and logs that are useful, measurable, and identifiable to the business and IT organization.
- Define an integration plan with existing ITSM solutions such as remedy or ServiceNow for incident generation or upstream monitoring. Determine which alerts should be forwarded, whether alert enrichment is required to support specific filtering requirements, and how to configure.
- Understand who needs visibility, what they need to see, and how it should be visualized based on their roles and responsibilities.

At the heart of operations management, your IT organization needs to establish centralized governance and strict delegation over approaches to build, operate, and manage IT services.

Initial strategy goals

As an architect or strategic planner, you may need to formulate an early strategy for operations management, in which monitoring plays a major role. Consider these four outcomes:

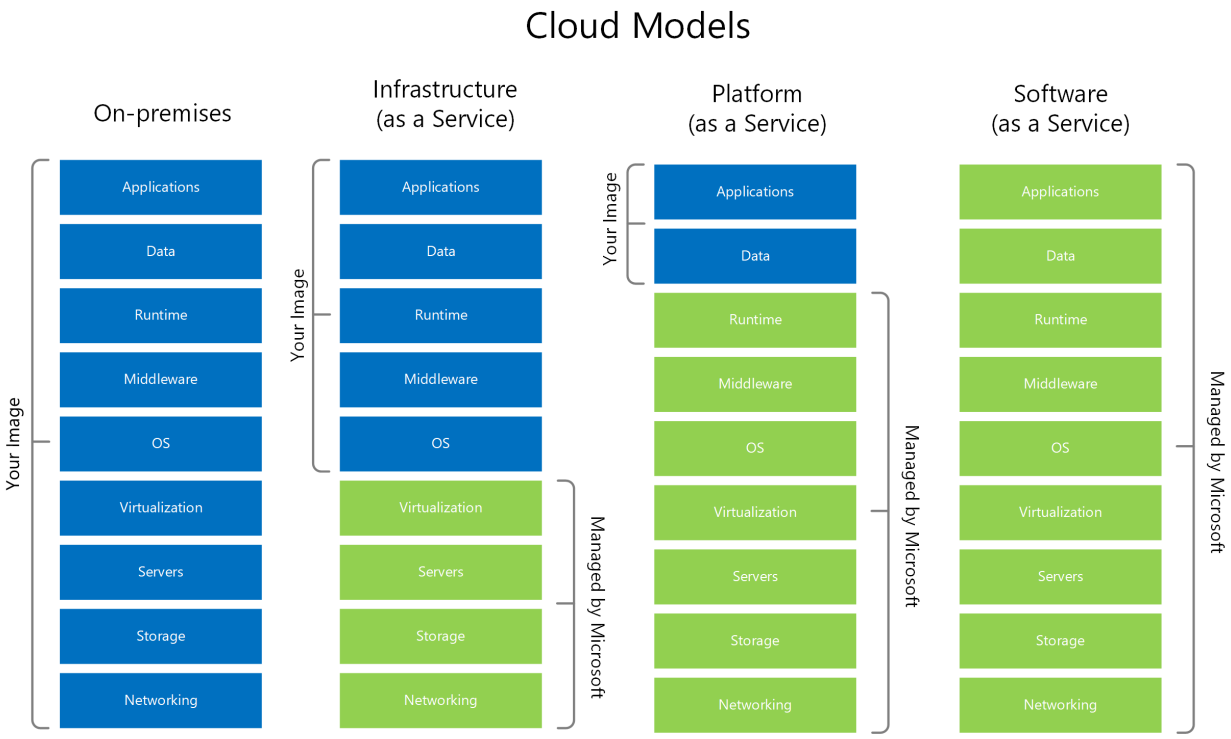
1. Manage cloud production services when they go live into production, such as networking, applications, security and virtual infrastructure.
2. Apply limited resources to rationalize your existing monitoring tools, skills and expertise, and use cloud monitoring to reduce complexity.
3. Make your monitoring solution processes more efficient, work faster and smoother, at scale and be able to change quickly too.
4. Account for how your organization will plan for and host monitoring based on cloud models. Work toward the goal of reducing your requirements as the organization transitions from IaaS to PaaS, and then to SaaS.

Determine what you have

As a manageability expert, you may be working closely with a steering committee, an architect, and strategic planners. You might be working to formulate your monitoring strategy by assessing the current state of your systems management, including: the people, partners, outsourcing, tooling, complexity, gaps, and risks. An assessment will help you prioritize the set of found problems and select the key opportunities that improve the current situation. Determine also, the services, systems, and data that are likely to remain on-premises as one important outcome. Ideally, management wants a roadmap of initiatives, but in direct proportion to the known planning horizon. Discussing unknowns, are as important.

High-level modeling

As the business determines what services to move, you need to invest your resources carefully. On-premises, you own all responsibilities for monitoring and are heavily invested. The moves made toward SaaS services, for example, do not eliminate your monitoring responsibility. You'll decide who needs access, who gets alerts, and who needs access to analytics at a minimum. Azure Monitor and Azure Arc are Azure services with the flexibility of addressing monitoring scenarios across all four cloud models, not just resources inside Azure. Look beyond the common cloud models as shown below. If you're using Microsoft Office applications delivered by Microsoft 365 services in your organization, you'll need to include security and compliance monitoring with Microsoft 365 in addition to Azure Security Center. This includes identities, endpoint management, and device monitoring outside of your corporate network.



Monitoring informs strategy

Consider where early monitoring capability *informs strategy*. Many decisions depend on early monitoring data in order to build a capability roadmap that guides limited resources and adds confidence. Strategies also need real-world input from monitoring of service enablement.

Consider the role monitoring plays in strategies to incrementally protect and secure the digital estate:

- Activity logs and security monitoring are needed to measure directory usage and external sharing of sensitive content, to inform in an incremental approach to layer on protective features and achieve the right balance with privacy monitoring.
- Policies and baselines will inform the rationalization objective (migrate, lift and shift, or rearchitect) and improve confidence that data and information can be migrated from on-premises to cloud services.

Later in this guide, discover some common monitoring scenarios or use cases that will help accelerate adoption.

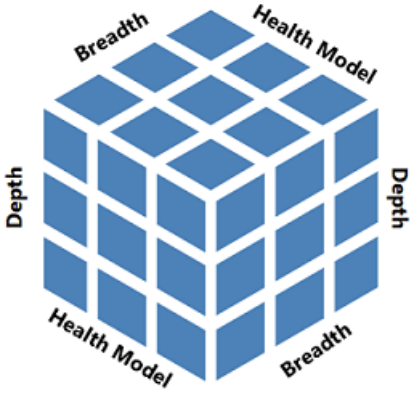
Formulate a monitoring architecture

Define your current and future architecture of systems management that includes monitoring, to:

- Apply limited resources in consolidating your monitoring investment.
- Decide how monitoring will help enable the future services your business needs: cloud monitoring of highly scalable, resilient, and globally aware cloud services.
- Align monitoring to the future services and resources that you will be monitoring in the cloud.
- Identify monitoring gaps across the three dimensions (depth, breadth, and across) of the health model.
- Model the financial aspects, costs, and support factors that support a cost-benefit analysis.
- Guide the hybrid decisions that you need to make.

One principle of monitoring is service visibility. For a service, asset or component to be fully visible, you need to balance the three sides of this principle, which are:

1. Monitoring in-depth by collecting meaningful and relevant signals.
2. Monitor end-to-end or breadth from the lowest layer of the stack up to the application.
3. East to west with a focus on its aspects of health (availability, performance, security, and continuity).



Some key questions include:

- How will you shape security logs and secure their access back to security and new privacy controls?
- Which services will be globally available and as such, can be globally monitored at the service edge?

- What about the network points between your network infrastructure, and network connectivity to service and application endpoints that tells us when it's us or the cloud provider?
- What are the boundaries of security operations versus health and performance? How can we provide summaries of health and status to security operations, and the converse, back to service owners?

To assemble this architecture, here are several considerations:

- A dataflow approach starting from service assets and going up the stack: metrics and log data emitted by infrastructure, IoT devices, mobile devices, and others. Are all of the items under management-to-monitoring tools (mid-tier)? Move upward and outward (ITSM tools, global monitoring, security information and event management (SIEM), custom alert enrichment, and others).
- Whether to continue with System Center Operations Manager or other monitoring tools.
- The economic cost.
- How the business will use logs and metrics. Azure Monitor brings a significant volume of log and time-series data to the performance and health side of monitoring, similar to what security operations experiences. Logs and metrics are two major data components of the Azure Monitor architecture. The reasons why this is important:

  1. Since you can build large-scale complex cloud services, your problem management costs are reduced to analyze, correlate, and determine causes of problems in one place reducing the need to access resources directly, thereby improving security.
  2. Similar to a SIEM, Azure Monitor is consolidating machine data directly from on-premises assets and Azure resources (including activity logs, tenant and subscription data, and any log data from a REST client), and provides a simple query language to provide data analysis far beyond what was possible before.

Consider your data flows and tools:

- Sources and types (telemetric, traces, stateful, time series).
- Tools and suites (rows): (Columns: availability, capacity, security, continuity, and compliance).
- The role of global monitoring or the top-tier.
- The role of IT service management (ITSM) integration to trigger on significant events.

Consider a single policy in your governance plan for event significance, throughout your enterprise, to drive alerting and notifications. It's one of the key policies in your monitoring strategy. The following table is an example of incident management priority model to standardize events, significance, and alerting used for notifications.

Priority		**Impact**			
		Extensive	Significant	Moderate	Minor
Urgency	Sev 0 - Critical	**Critical**	**Critical**	**Error**	**Error**
	Sev 1 = Error	**Critical**	**Error**	**Error**	Warning
	Sev 2= Warning	**Error**	**Error**	Warning	Warning
	Sev 3 = Informational	Low	Low	Low	Low

Formulate initiatives

As a monitoring expert or systems administrator, you've discovered that cloud monitoring is faster and easier to establish, leading to inexpensive demos or proofs-of-value. To overcome the tendency to stay in demo mode, you need to stay in constant touch with strategy and be able to execute on production-focused monitoring plans. Because strategy has plenty of uncertainty and unknowns, you won't know all of the monitoring requirements in advance. Therefore, decide on the first set of adoption plans, based on what is minimally viable to the business and IT management. You can refer to this capability as *that which is needed to begin the journey*. Here are two example initiatives that help declare forward motion:

- **Initiative 1:** To reduce the diversity and complexity of our current monitoring investment, we will invest in establishing a core capability using Azure Monitor first, given the same skills and readiness applies to other areas of cloud monitoring.
- **Initiative 2:** To decide on how we use our license plans for identity, access, and overall information protection, we will help the security and privacy offices establish early activity monitoring of users and content as they migrate to the cloud, to clarify questions on classification labels, data loss prevention, encryption, and retention policies.

Consider scale

Consider scale in your strategy and who will be defining and standardizing *monitoring as code*. Your organization should plan to build standardized solutions using a combination of tools such as:

- Azure Resource Manager templates.
- Azure Policy monitoring initiative definitions and policies.
- GitHub to establish a source control for the scripts, code, and documentation.

Consider privacy and security

In Azure, you'll need to secure certain monitoring data emitted by resources and the control plane actions that are logged in Azure, known as activity logs. Additionally, specialized logs that record user activity such as the Azure Active Directory sign-in and audit logs, and if integrated, the Microsoft 365 unified audit log, as they contain sensitive data that may need to be protected under privacy laws.

Your monitoring strategy should include these components:

- Separate non-monitoring data from monitoring data
- Restrict access to resources

Consider business continuity

Azure Monitor collects, indexes, and analyzes real-time machine and resource-generated data to support your operations and help drive business decisions. Under rare circumstances, it's possible that facilities in an entire region can become inaccessible, for example due to network failures. Or facilities can be lost entirely, for example due to a natural disaster. By relying on these services in the cloud, your planning isn't focused around infrastructure resiliency and high availability, rather its planning for:

- Availability for data ingestion from all your dependent services and resources in Azure, resources in other clouds, and from on-premises.
- Data availability for insights, solutions, workbooks and other visualizations, alerting, integration with ITSM, and other control plane services in Azure supporting your operational requirements.

Create a recovery plan, and make sure that it covers data restoration, network outages, dependent service failures, and region-wide service disruptions.

Consider maturity

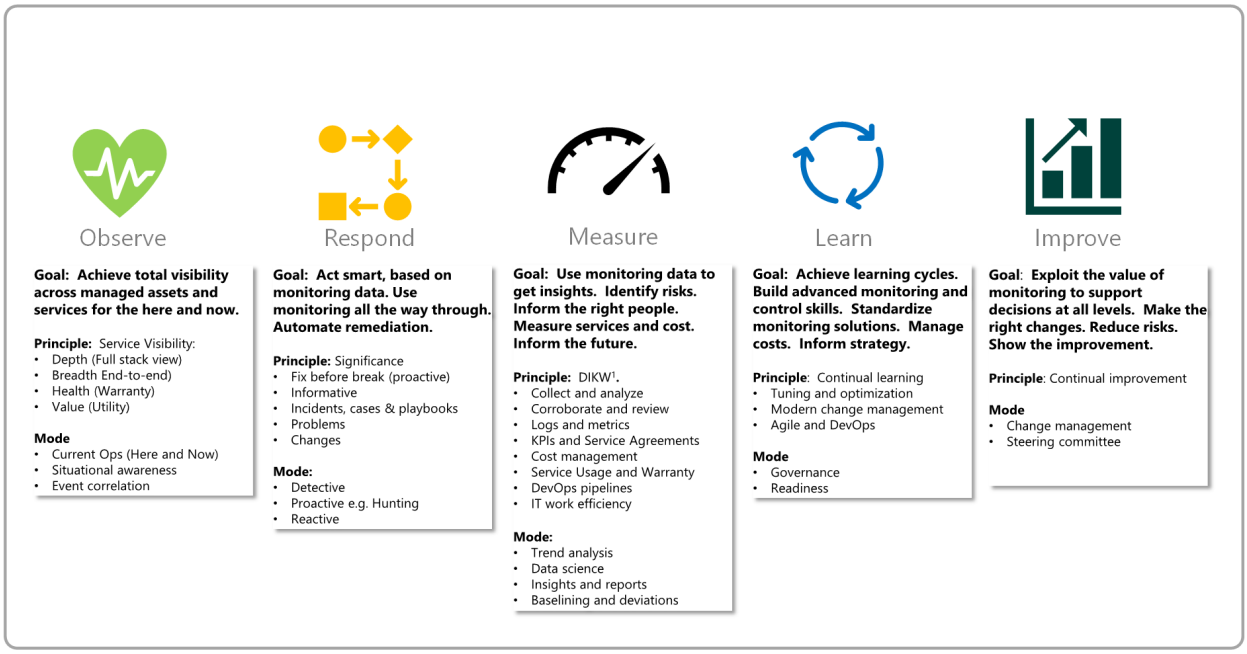
Maturity is an important consideration in your monitoring strategy. We recommend you start minimally, gather data and with this information, determine the strategy. The first monitoring solutions you'll want are those that ensure observability, to include responsive processes, such as incident and problem management. Here, you

will:

- Create one or more Log Analytics workspaces
- Enable agents
- Enable resource diagnostic settings
- Enable initial alert rules

Over time, you gain confidence in Azure Monitor capabilities with the need to measure health indicators, so this involves expanding the focus on the collection of logs, enabling and using insights and metrics, and defining log search queries that drive the measurement and calculation of what is healthy or unhealthy.

Learning cycles includes getting monitoring data and insights into the hands of managers, and ensuring the right consumers have monitoring data they need. Learning cycles include continual tuning and optimizing of your initial monitoring plans to adapt, to improve service, and inform adoption plans.



[1] The dikw model is an often used method, with roots in knowledge management, to explain the ways we move from Data to Information, Knowledge, and Wisdom with a component of actions and decisions.

Monitoring is foundational for services you build in Azure. Your strategy can address these four disciplines of modern monitoring, to help you define minimum viable monitoring, and gain confidence in steps. Moving your capability from reactive to proactive and scaling its reach to end users is but one goal.

- **Observe:** First, you should focus on establishing monitoring to observe the health and status of Azure services and resources. Configure basic monitoring and then automate with Azure Policy and Azure Resource Manager templates, to establish initial visibility of services and their warranty: availability, performance or capacity, security, and configuration compliance. For example, based on minimum viable setup of Azure Monitor, configure resources for monitoring and diagnostics, set up alerts, and insights. Include knowledge and readiness of monitoring consumers, defining and triggering from events, for service work such as incidents and problems. One indicator of maturity is how much can be automated to reduce unnecessary human costs to manually observe health and status. Knowing which services are healthy is as important as being alerted on services that are unhealthy.
- **Measure:** Configure collection of metrics and logs from all resources to monitor for symptoms/conditions that are issues, which indicate potential or actual impact to the availability of the service, or impact of the consumers of the service/application. For example:
    - When using a feature in the application, is it showing response time latency, returning an error

when I selected something, or unresponsive?

    - Ensure services are meeting service agreements by measuring the utility of the service or application.

- **Respond:** Based on the context of known issues to observe and measure, evaluate what qualifies as a bug, autoremediation, or requires manual response based on what is classified as an incident, problem, or change.
- **Learn and improve:** Providers and consumers participating in learning cycles implies consuming actual monitoring data through insights, reports and workbooks, to continually improve the target service and to enact tuning and optimization of the monitoring configuration. Change is important too, that the monitoring configuration is changing in tandem with changes to the service (such as new, modified, or retired) and continues to match the actual service warranty.

To help you align monitoring plans to strategy, use the following table to categorize the different monitoring scenarios that occur in more detail. This works with the five Rs of rationalization introduced earlier during the planning phase. If you're using System Center Operations Manager, you have hybrid and cloud options available to rationalize your investment.

TYPE	MONITORING OBJECTIVE	EXAMPLE OBJECTIVE
1	Only on-premises	System Center Operations Manager. Continue to monitor services, infrastructure, networking up the application layer in owned datacenters with no cloud considerations.
2	On-premises to the cloud	Continue using System Center Operations Manager, and apply the Microsoft 365 and Azure management packs.
3	On-premises to/with cloud (cooperative) where services run in both the cloud and on-premises	Establish initial monitoring with Azure Monitor. Connect Azure Monitor to System Center Operations Manager and alert sources, such as Zabbix or Nagios. Deploy Azure Monitor monitoring agents, multihoming with System Center Operations Manager where they monitor cooperatively.
4	Hybrid migration	Monitor the migration, for example, Microsoft Exchange Server to Microsoft 365 Exchange Online. Exchange Online service health and service usage, security and compliance, all from Microsoft 365. Gradually decommission monitoring exchange on-premises with System Center Operations Manager until migration is complete.
5	Hybrid forever	System Center Operations Manager, Azure AD, Azure Monitor, Azure Security Center, Intune, and others; a range of tools for a mixture of digital assets.

TYPE	MONITORING OBJECTIVE	EXAMPLE OBJECTIVE
6	Cloud native	Azure Monitor, Azure Policy, Azure Security Center, Microsoft 365, Azure Service Health, Azure Resource Health, and others.
7	Multicloud owned tenants (consolidate)	Centralize the monitoring of many tenants. Azure Lighthouse, Azure Policy, Azure Monitor, and Azure Sentinel.
8	Multicloud ecosystem	Centralize the monitoring of different cloud providers: Microsoft, Amazon, Google, and others.
9	Provider > consumer	Monitoring solutions and services as a cloud provider.

Formulate monitoring requirements

As you progress through this process, your strategy reveals there may be much to do in the end. Ultimately your mindset extends outside the corporate network into the workplace, to devices and endpoints, and further outward to the identity-as-security boundary. The new edge defined with cloud monitoring is a strong motivator in contrast with a datacenter and workplace mindset.

You can use Azure now to gradually begin managing all or some aspects of your on-premises resources, even for services you'll keep on-premises. You also want strategy to define your monitoring boundaries of responsibility in alignment with the business' cloud adoption strategy, based on the cloud service model your business adopts. Even for services based on IaaS, you'll get metrics, logs, views, and alerting capabilities through Azure Service Health and here, you'll configure alerts from availability monitoring of your Azure resources with resource health. With SaaS services, such as Microsoft 365, much is already provided, and you need to configure appropriate access to portals, dashboards, analytics, and alerts. From a service perspective, a large service with distributed components such as Microsoft 365 Exchange Online have many objectives, not just the need to observe its health and status.

PRIMARY OBJECTIVE	GOAL AND OUTCOME
Health and status monitoring	Holistically observe, measure, learn, and improve the long-term warranty of the service or component, including service levels, in these aspects taken together: availability, capacity, performance, security, and compliance. A healthy system, service or component is online, performing well, secure and compliant. Health monitoring includes logs and is stateful with real-time health states and metrics. It also includes trending reports, insights, and trends focused on service usage.
Utility monitoring	Observe, measure, learn, and improve the quality or qualitative aspects of how a system delivers value. User experience is one type of monitoring use case.

PRIMARY OBJECTIVE	GOAL AND OUTCOME
Security monitoring	Observe, measure, learn, and improve protection in support of cybersecurity strategy and functions such as security operations, identity and access, information protection, privacy, threat management and compliance. Monitor using Azure Security Center and Azure Sentinel, and Microsoft 365.
Cost monitoring	Monitor usage and estimate costs using Azure Monitor and Azure Cost Management + Billing as a new primary objective. The Azure Cost Management + Billing APIs allow you to explore cost and usage data using multidimensional analysis.

TERTIARY OBJECTIVES	GOAL AND OUTCOME
Activity monitoring	Observe, measure, learn, and improve usage, security, and compliance from sources such as Azure activity logs, audit logs, and the Microsoft 365 unified audit log for subscription level events, actions on resources, user and administrator activity, content, data, and for your security and compliance needs in Azure and Microsoft 365.
Service usage	Service owners want analytics and insights to measure, learn, and improve the usage of Azure and Microsoft 365 services (IaaS, PaaS, SaaS) with service usage reports, analytics, and insights. Ensure plans include who will need access to the admin portals, dashboards, insights, and reports.
Service and resource health	Observe the health of your cloud resources, and service outages and advisories from Microsoft, to stay informed about incidents and maintenance. Include resource health in monitoring of the availability of your resources and alert on changes in availability.
Capacity and performance monitoring	In support of health monitoring, your needs may require more depth and specialization.
Change and compliance monitoring	Observe, measure, learn, and improve configuration management of resources, which should now include security in the formulation, influenced by good use of Azure Policy to standardize monitoring configurations and enforce security hardening. Log data to filter on key changes being made on resources.
Identity and access monitoring	Observe, measure, learn, and improve both the usage and security of Active Directory, Azure Active Directory, and identity management that integrates users, applications, devices, and other resources no matter where they are.
Information protection	Not only Azure Monitor, but Azure Information Protection depending on the plan, includes usage analytics critical to your development of a robust information protection strategy across Azure and Microsoft.

TERTIARY OBJECTIVES	GOAL AND OUTCOME
Privacy monitoring	Organizations face expanding privacy needs to include information protection of the digital estate, data classification, and data loss prevention to mitigate risks to privacy breaches and infractions. Microsoft 365 information protection includes monitoring capabilities that can also be integrated with Azure Monitor.
Threat management and integrated threat protection	The cloud brings together the separate, traditional roles of security monitoring with health monitoring. Integrated threat protection, for example, involves monitoring to accelerate an optimal state of zero trust. Integrating Microsoft Defender for Identity allows a migration from using System Center Operations Manager to monitor Active Directory, and integrate your Active Directory security-related signals to detect advanced attacks in hybrid environments.

Agile solution releases

Ultimately, you'll deliver monitoring configurations or solutions into production. As an IT Operations Manager or monitoring team lead, consider a standard, simple taxonomy to improve communication with consumers, managers, and IT operations. An agile DevOps approach ensures monitoring is embedded within the teams who will be building and operating cloud services. While traditional project management works, it's not fast enough nor typically accepted as a standard practice by operations teams.

Include in your strategy and operating model how you communicate monitoring plans, objectives, and configurations (the solutions). For example, how you might use Azure Boards:

AGILE TERM	WHAT TO INCLUDE	EXAMPLES
Epics	Broad monitoring Initiatives of the monitoring strategy	Consolidate Azure cloud monitoring Hybrid cloud monitoring Private cloud monitoring Establish core monitoring service
Features	Individual monitoring Plans and projects	Monitoring requirements Monitoring consumers and providers Objectives Tools Schedule
User stories and tasks	The end result is a monitoring configuration or solution	Network monitoring (for example, ExpressRoute) Standardized IaaS VM monitoring (for example Azure Monitor for VMs, Application Insights, Azure Policy, settings, policies, reports, workspaces.)

Establish minimum governance

As early as possible, establish how you intend to govern your cloud monitoring investment. Remember that Azure Monitor is a *tenant* service with visibility across management groups and subscriptions, and users can be scoped to limit their actions with Azure role-based access control.

Define who will have what level of access in Azure to support their role and responsibility. We recommend you

to set Reader role access for monitoring consumers as early as possible and then start controlling who are granted the Contributor role.

First, identify the roles who will own and manage resource groups in Azure as part of your governance framework:

- Whether a monitoring team or one or more administrators of resources and resource groups will have privileged access to the Monitoring Contributor role.
- The consumers who should be granted the Monitoring Reader role, which enables access to features in Azure Monitor, as well as investigate issues within the monitoring section that is included with each Azure resource.
- Which managers require access to other Azure reader roles, such as the Reports Reader role.

In summary, your monitoring consumer roles probably need broad access, versus your developers and system administrators who only need role-based access to certain Azure resources. As another restriction, ensure you exempt readers from access to sensitive monitoring data such as security, sign-in, and user activity logs.

Establish readiness

Early on, formulate a readiness plan to help your IT staff adopt new skills, practices, and techniques for cloud monitoring in Azure. Consider the skills readiness guidance for monitoring that includes foundational needs, as well as those specific to monitoring.

Next steps

Observability

Responsible AI

6/15/2021 • 2 minutes to read • Edit Online

Driven by ethical principles that put people first, Microsoft is committed to advancing AI. We want to partner with you to support this endeavor.

Responsible AI principles

As you implement AI solutions, consider the following principles in your solution:

- **Fairness:** AI systems should treat all people fairly.
- **Reliability and safety:** AI systems should perform reliably and safely.
- **Privacy and security:** AI systems should be secure and respect privacy.
- **Inclusiveness:** AI systems should empower everyone and engage people.
- **Transparency:** AI systems should be understandable.
- **Accountability:** People should be accountable for AI systems.

Establish a responsible AI strategy

Learn how to develop your own responsible AI strategy and principles based on the values of your organization.

- Get started at AI Business School

Guidelines to develop AI responsibly

Put responsible AI into practice with these guidelines, designed to help you anticipate and address potential issues throughout the software development lifecycle.

- Human-AI interaction guidelines
- Conversational AI guidelines
- Inclusive design guidelines
- AI fairness checklist
- Datasheets for datasets template
- AI security engineering guidance

Tools for responsible AI

Tools are available to help developers and data scientists understand, protect, and control AI systems. These tools can come from a variety of sources, including Azure Machine Learning, open-source projects, and research.

- **Understand:** AI systems can behave unexpectedly for a variety of reasons. Software tools can help you understand the behavior of your AI systems so that you can better tailor them to your needs. Examples of this type of tool include InterpretML, Error Analysis and Fairlearn.
- **Protect:** AI systems rely on data. Software tools can help you protect that data by preserving privacy and ensuring confidentiality. Examples of this type of tool include Confidential Computing for Machine Learning, SmartNoise differential privacy toolkit, SEAL Homomorphic Encryption toolkit, and the Presidio data de-identification toolkit.
- **Control:** Responsible AI needs governance and control through the development cycle. Azure Machine Learning enables an audit trail for better traceability, lineage, and control to meet regulatory requirements. Examples include audit trail and traceability.

Next steps

For more information about responsible solution development, visit:

- Responsible AI overview
- Responsible AI resources
- Responsible bots: 10 guidelines for developers of conversational AI

How GitHub accelerates cloud adoption

5/24/2021 • 9 minutes to read • Edit Online

Overview

Innovation is the new currency in today's competitive landscape. Ride sharing, streaming content, self-driving cars, and other services have fundamentally changed people's daily rhythms while turning markets upside down and showing how the competitive landscape has moved from physical assets to digital experiences.

These types of superior digital experience are leading a disruption where well-established businesses face stiff competition from companies that can innovate and deliver value to their customers faster. To compete and avoid disruption, businesses need to build a culture of innovation and use the best and most fitting tools and cloud services.

GitHub provides a range of features that can help companies to:

- Take advantage of Azure services and capabilities.
- Modernize their practices.
- Become more agile and innovative during this cultural shift.

Companies can take advantage of GitHub's connectedness to the open-source community and find thousands of reiterated, enhanced, and ready-to-deploy cloud solution examples from organizations that have successfully adopted Azure services. They can easily borrow from and iterate on these solutions to tailor them to their business needs.

GitHub makes it easy for organizations to share within their teams, which makes it faster to modernize and deploy the next application or workload. Companies can look to *innersource*, a key tenet of innovation, to borrow best practices like sharing and reuse, collaboration and communication, and more from the open-source community and apply them within their organization.

From securing the open-source packages to the intellectual property that's written daily, securing the entire software supply chain should be a main priority for to every company. This goal requires advanced security technology that can be incorporated and automated throughout the entire lifecycle, and native GitHub capabilities like GitHub advanced security and GitHub Actions offer this type flexibility.

Take advantage of open-source assets

Highly effective organizations recognize open-source software (OSS) as essential versus optional for modern software development. They engage with the developer communities on which they depend and use a secure platform to strategically invest in OSS. As a result, these organizations experience innovate quickly, outpace competitors, and cut costs while minimizing risk.

While OSS could be interpreted as the packages, libraries, scripts, and dependencies incorporated into applications, there are thousands of open-source assets in the form of infrastructure as code (IaC), documentation, guidance, and blueprints for well-defined Azure architectures. These blueprints have been contributed to the OSS community by Microsoft, partners, vendors, customers, and individuals, and they're readily available in GitHub. They can be easily modified, reused, and deployed to a specific Azure environment.

Infrastructure as code

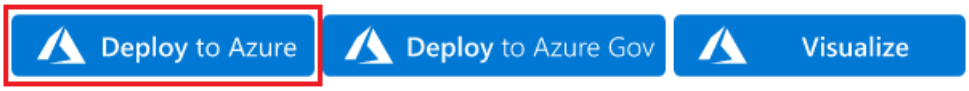
Infrastructure as code (IaC) is the management of infrastructure (networks, virtual machines, load balancers, and connection topology) in a descriptive model, using the same versioning system that the DevOps team uses for source code. Like the principle that the same source code generates the same binary, an IaC model generates

the same environment every time that it's applied. IaC is a key DevOps practice that's used with continuous delivery (CD).

IaC evolved to solve the problem of environment drift in the release pipeline. Without it, teams must maintain the settings of individual deployment environments, and inconsistencies between environments lead to issues during deployments. Every environment eventually becomes a snowflake, a unique configuration that can't be reproduced automatically. With snowflakes, infrastructure administration and maintenance requires manual processes that contribute to errors and are hard to track. Infrastructure deployments with IaC are repeatable and prevent runtime issues caused by configuration drift or missing dependencies.

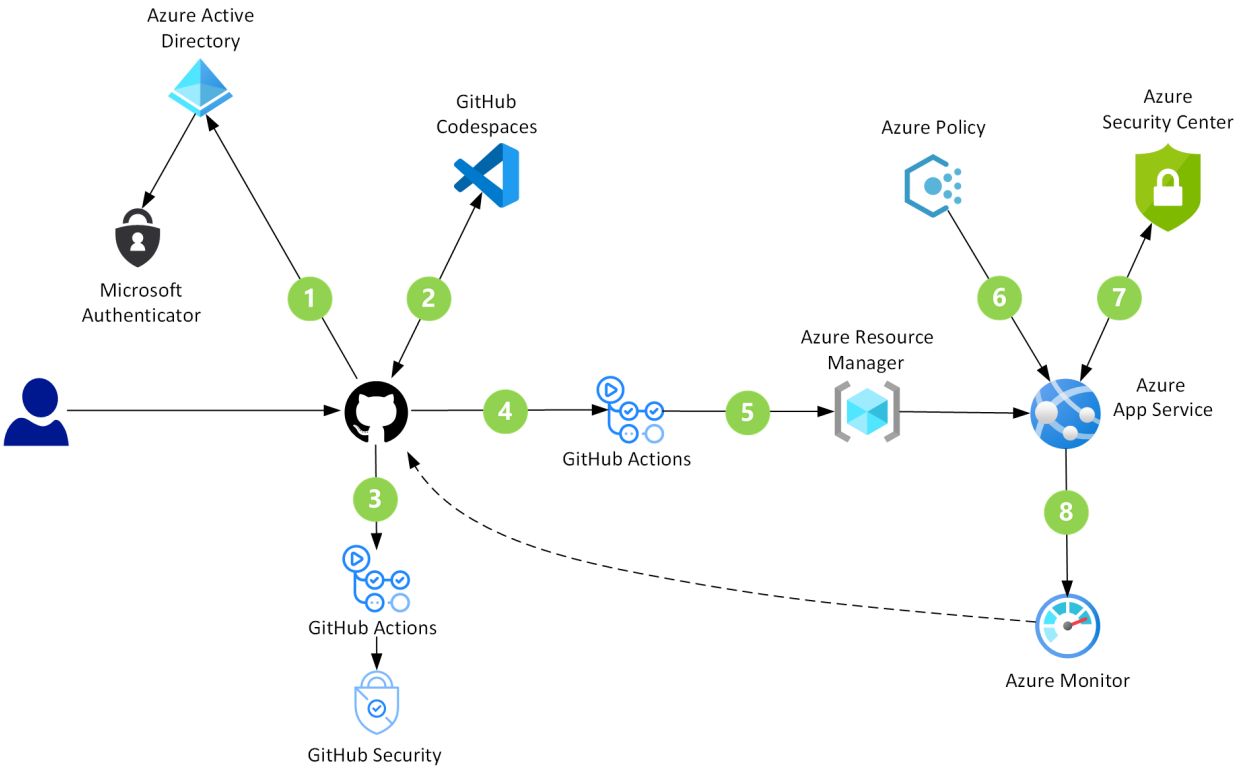
With IaC, teams make changes to the environment description and version the configuration model, which is typically in well-documented code formats like JSON; see Azure Resource Manager templates for more information. Developers can simplify their workflows by hosting IaC code in the same GitHub repo as their application source code and adopt the same continuous integration (CI) /CD practices for IaC powered by GitHub Actions.

See the AzOps GitHub action for an example of how to deploy custom Resource Manager templates at various Azure scopes. If you're new to Resource Manager templates or IaC, you can also browse the `azure-quickstart-templates` repo on GitHub, find the template that you wish to deploy, and select the **Deploy to Azure** button to test how it works.



Cloud pattern components and best practices

The following architecture diagram highlights the security checks that run in the GitHub and Azure components of a GitHub DevSecOps environment:



- GitHub provides a code-hosting platform that developers can use for collaborating on open-source and innersource projects.
- Codespaces is an online development environment. Hosted by GitHub and powered by Microsoft Visual Studio Code, this tool provides a complete development solution in the cloud.

- GitHub security works to eliminate threats in multiple ways. Agents and services identify vulnerabilities in repositories and dependent packages. They also upgrade dependencies to current and secure versions.
- GitHub Actions are custom workflows that provide CI/CD capabilities directly in repositories. Computers named runners host these CI/CD jobs.
- Azure Active Directory is a multitenant, cloud-based identity service that controls access to Azure and other cloud applications like Microsoft 365 and GitHub.
- Azure App Service provides a framework for building, deploying, and scaling web apps. This platform offers built-in infrastructure maintenance, security patching, and scaling.
- Azure Policy helps teams manage and prevent IT issues through policy definitions that can enforce rules for cloud resources. For instance, if a project is about to deploy a virtual machine with an unrecognized SKU, Azure Policy sends alerts about the problem and stops the deployment.
- Azure Security Center provides unified security management and advanced threat protection across hybrid cloud workloads.
- Azure Monitor collects and analyzes performance metrics, activity logs, and other application telemetry. This service alerts applications and personnel when it identifies irregular conditions.

Innersource

Innersource overview

Many companies use the term *innersource* to describe how their engineering teams work together on code. Innersource is a development methodology where engineers build proprietary software with best practices from large-scale open-source projects like Kubernetes or Visual Studio Code.

Large-scale open-source projects require coordination and teamwork across thousands of contributors. The most successful projects are driven by a vision for their future and daily user needs: speed, reliability, and functionality. The scale at which these projects operate provides some lessons and can help companies build better software more quickly with innersource.

With GitHub's pull requests and issues, collaboration and code review are built into the development process. Internal and outsourced teams can share work, discuss changes, and get feedback all in one place. This helps organizations share expertise internally and avoid reinventing field-tested solutions developed for other projects.

The anatomy of an innersource project

The right mix of individuals, teams, and resources can ensure a project's success. Many open-source projects follow a similar organizational structure that can help organizations to set up cross-functional teams to manage innersource projects. A typical open-source project has the following types of people:

- **Maintainers:** These contributors are responsible for driving the vision and managing the organizational aspects of the project. They might not be the original owners or authors of the code.
- **Contributors:** These people are everyone who has contributed something to the project.
- **Community members:** These are people who use the project. They might be active in conversations or express their opinion on the project's direction.

Bigger projects could also have subcommittees or working groups focused on different tasks like tooling, triage, and community moderation. Innersource projects are likely to follow a similar structure. Many engineering organizations sort developers into teams like application engineering, platform engineering, and web development. Structuring organizations this way can leave blind spots that exclude qualified people. Organizing a core decision-making group supported by teams across an organization can help to rally the expertise

necessary to solve problems faster.

Within an enterprise, contributors are developers across the company, and maintainers are a project's leaders and key decision-makers.

- **Maintainers:** Developers, product managers, and other key decision-makers within a company responsible for driving a project's vision and for managing day-to-day contributions.
- **Contributors:** Developers, data scientists, product managers, marketers, and other roles within a company that help drive software forward. Contributors might not be part of the direct project team but help build software by contributing code, submitting bug fixes, and more.

For more information, see the white paper An introduction to innersource.

Automation

GitHub Actions allows users to create custom workflows directly in their GitHub repositories. Users can discover, create, and share actions to perform any job, including CI/CD, and combine actions in a completely customized workflow. They can also create CI workflows that build and test projects written in different programming languages. Examples are available in the guides for GitHub Actions.

GitHub Actions can be used to combine IaC concepts and CI/CD practices to automate the entire end-to-end deployment lifecycle, including provisioning or updating the target environment in a repeatable manner and packaging and deploying the application itself.

Example:

GitHub Actions for Azure are built to simplify how you automate your deployment processes to target Azure services such as Azure App Service, Azure Kubernetes Service, Azure Functions, and more. The Azure starter action workflows repository includes end-to-end workflows to build and deploy web apps of any language and any ecosystem to Azure. Visit GitHub marketplace to see all of the actions that are available.

Security

GitHub's shift-left security features

Starting with the first steps of development, DevSecOps adheres to security best practices. By using a shift-left strategy, DevSecOps redirects the security focus. Instead of pointing toward auditing at the end, it shifts to development in the beginning. Besides producing robust code, this fail-fast approach helps to resolve problems early when they're easy to fix.

With many security capabilities, GitHub offers tools that support every part of a DevSecOps workflow:

- Browser-based IDEs with built-in security extensions
- Agents that continuously monitor security advisories and replace vulnerable and out-of-date dependencies
- Search capabilities that scan source code for vulnerabilities
- Action-based workflows that automate every step of development, testing, and deployment
- Spaces that provide a way to privately discuss and resolve security threats and then publish the information
- Combined with the monitoring and evaluation power of Azure, these features provide a superb service for building secure cloud solutions

Example:

GitHub DevSecOps installations cover many security scenarios. Possibilities include the following cases:

- Developers who want to take advantage of preconfigured environments that offer security capabilities.
- Administrators who rely on having up-to-date, prioritized security reports at their fingertips, along with details on affected code and suggested fixes.

- Streamlined organizations that need systems to automatically acquire new and uncompromised security devices when secrets are left exposed in code.
- Development teams that could benefit from automatic upgrades when newer or more secure versions of external packages become available.

For more information, see:

- DevSecOps in GitHub: Azure solution ideas
- Code scanning a GitHub repository using GitHub advanced security within an Azure DevOps pipeline
- Applying DevSecOps to your software supply chain

Next steps

- Choose your implementation team (typically a developer manager and a few developers defined as admins), and deploy GitHub.
- Learn common and advanced Git workflows to enhance how you use GitHub.

The following links provide more information about GitHub.

- GitHub modules in Microsoft Learn
- GitHub Learning Lab
- GitHub Docs
- Tips for getting started with GitHub DevSecOps

Skills readiness path during the planning phase of a migration journey

3/6/2021 • 4 minutes to read • Edit Online

The objective during the planning phase of a migration journey is to develop the plans necessary to guide migration implementation. This phase requires a few critical skills, including:

- Establishing the vision.
- Building the business justification.
- Rationalizing the digital estate.
- Creating a migration backlog (technical plan).

The following sections provide learning paths to develop each of these skills.

Establish the vision

The success of any cloud adoption effort is defined by the business vision. When the technical team doesn't understand the motives and desired outcomes, it's hard for them to guide their efforts toward business success. See these articles for information about documenting and articulating the business vision for the technical team:

- Adoption motivations. Document and articulate the reasons behind the technical effort.
- Business outcomes. Clearly articulate what's expected of the technical team in terms of business changes.
- Learning metrics. Establish short-term metrics that can show progress toward longer-term business outcomes.

Build the business justification

Justifying the investment to adopt the cloud can require deeper analysis and an understanding of your organization's accounting practices. The articles on business justification can help you develop these skills:

- Cloud migration business case. Establish a business case for cloud migration.

Rationalize the digital estate

You can refine your business case by aligning the desired business case with current and future digital estate inventory. These articles can guide the development of a digital estate rationalization:

- Incremental rationalization: An agile approach to rationalization that properly aligns late-bound technical decisions.
- The five Rs of rationalization: Understand the various rationalization options.

Create a migration backlog (technical plan)

Convert the business case and rationalized digital estate into an actionable migration plan to guide the technical activities required to achieve the desired business outcomes.

Business planning skills

During the readiness phase, technical staff creates a migration landing zone capable of hosting, operating, and governing workloads that have been migrated to the cloud. These learning paths can help you develop the

necessary skills:

- Create an Azure account. The first step to using Azure is to create an account. Your account holds the Azure services you provision and handles your personal settings, like identity, billing, and preferences.
- Azure portal. Tour the Azure portal features and services, and customize the portal.
- Introduction to Azure. Get started with Azure by creating and configuring your first virtual machine in the cloud.
- Introduction to security in Azure. Learn the basic concepts for protecting your infrastructure and data when you work in the cloud. Understand what responsibilities are yours and what Azure takes care of for you.
- Manage resources in Azure. Learn how to work with the Azure command line and web portal to create, manage, and control cloud-based resources.
- Create a VM. Create a virtual machine by using the Azure portal.
- Azure networking. Learn the basics of Azure networking and how Azure networking helps you improve resiliency and reduce latency.
- Azure compute options. Learn about the Azure compute services.
- Secure resources with Azure RBAC. Use Azure RBAC to secure resources.
- Data storage options. Learn about the benefits of Azure data storage.

Organizational skills

Depending on the motivations and desired business outcomes of a cloud adoption effort, leaders might need to establish new organizational structures or virtual teams to facilitate various functions. These articles will help you develop the skills necessary to structure those teams to meet desired outcomes:

- Initial organizational alignment. Overview of organizational alignment and various team structures to facilitate specific goals.
- Breaking down silos and fiefdoms. Understanding two common organizational antipatterns and ways to guide a team to productive collaboration.

Deeper skills exploration

Various learning options beyond these initial options are available for developing skills.

Typical mappings of cloud IT roles

Microsoft and partners offer various options for all audiences to develop skills with Azure services.

- Map roles and skills: A resource for mapping your cloud career path. Learn about your cloud role and suggested skills. Follow a learning curriculum at your own pace to build the skills that you need most to stay relevant.
- Explore Azure certification training and exams to gain official recognition for your Azure knowledge.

Microsoft Learn

Microsoft Learn is a new approach to learning. Readiness for the new skills and responsibilities that come with cloud adoption doesn't come easily. Microsoft Learn provides a more rewarding approach to hands-on learning that helps you achieve your goals faster. Earn points and levels, and achieve more!

Here is an example of a tailored learning path that aligns to the Strategy methodology of the Cloud Adoption Framework.

Learn the business value of Microsoft Azure: This learning experience will take you on a journey that will begin by showing you how digital transformation and the power of the cloud can transform your business. We will cover how Microsoft Azure Cloud Services can power your organization on a trusted cloud platform. Finally, we will wrap up by illustrating how to make this journey real for your organization.

Learn more

To discover more learning paths, browse the Microsoft Learn catalog. Use the **Roles** filter to align learning paths with your role.

Cloud strategy antipatterns

3/29/2021 • 2 minutes to read • Edit Online

Customers often experience antipatterns during the Strategy phase of cloud adoption. These antipatterns can make it hard to align IT and business strategies. They also make it hard to measure the success of cloud projects.

Antipattern: Adopt the cloud without establishing goals

Many companies announce cloud-first or cloud-only strategies. But, they don't clearly define what they want to achieve with those strategies. Few cloud projects can succeed without concrete KPIs and goals. It's impossible to measure project performance without indicators or specified targets.

Example: Migrate to the cloud without defining goals

A corporation's closest competitor launches a cloud-only strategy. The competitor's goal is to accelerate business by having all systems in the cloud within a year. The corporation doesn't want to trail behind. The corporation's directors begin strategic discussions on how to adopt the cloud quickly. But, they don't define any concrete success criteria like reducing costs or improving system performance.

The corporation's first system migrates to the cloud. The directors can't check whether their cloud strategy is successful because they never defined what they wanted to achieve.

Preferred outcome: Define goals and KPIs

Define concrete KPIs when discussing your reasons for adopting the cloud. Then you'll be able to measure how successful your strategy is. You'll also know whether you can use the same strategy for other projects. See Why are we moving to the cloud? to learn more about motivations for cloud adoption.

Antipattern: Fail to communicate motivations

Cloud adoption journeys can fail when motivations or cloud adoption triggers are misaligned within a company. A business might see major benefits in adopting the cloud but fail to communicate these adoption triggers to IT. This problem even comes up when these motivations influence the company's IT strategy, not just its business strategy. Without alignment or documented motivations, cloud journeys often fail.

Example: Fail to communicate benefits

A corporation starts using the cloud when its managing board announces a cloud-first IT strategy. But, the board doesn't explain how the strategy benefits the company. The IT and business departments aren't certain why they're adopting the cloud. This uncertainty leads to a lack of focus, meaning the departments fail to work toward this common goal. Hesitancy about adopting the cloud increases, especially within IT. Because the IT strategy has poorly defined goals, that strategy comes under scrutiny, leading to more questions than answers.

Preferred outcome: Define and communicate reasons for cloud adoption

Decide why you want to adopt the cloud. Then clearly define your reasons and communicate them throughout the company. Your IT and business departments will then accept cloud adoption more readily. You can also use these motivations to influence technical decisions in later stages of the cloud adoption journey. Before justifying a move to the cloud, review cloud migration myths.

Next steps

- Why are we moving to the cloud?
- Build a business justification for cloud migration

Develop a cloud adoption plan

3/6/2021 • 2 minutes to read • Edit Online

Cloud adoption plans convert the aspirational goals of a cloud adoption strategy into an actionable plan. The collective cloud teams can use the cloud adoption plan to guide their technical efforts and align them with the business strategy.

The following exercises will help you document your technology strategy. This approach captures prioritized tasks to drive adoption efforts. The cloud adoption plan then maps to the metrics and motivations defined in the cloud adoption strategy.

	Digital estate: Inventory and rationalize your digital estate based on assumptions that align with motivations and business outcomes.
	Initial organizational alignment: Establish a plan for initial organizational alignment to support the adoption plan.
	Skills readiness plan: Create a plan for addressing skills readiness gaps.
	Cloud adoption plan: Develop a cloud adoption plan to manage change across the digital estate, skills, and organization.

Download the strategy and plan template to track the outputs of each exercise as you build out your cloud adoption strategy. Also, learn about the five Rs of cloud rationalization to help build your cloud adoption plan.

Cloud rationalization

3/6/2021 • 4 minutes to read • Edit Online

Cloud rationalization is the process of evaluating assets to determine the best way to migrate or modernize each asset in the cloud. For more information about the process of rationalization, see What is a digital estate?.

Rationalization context

The *five Rs of rationalization* listed in this article are a great way to label a potential future state for any workload that's being considered as a cloud candidate. This labeling process should be put into the correct context before you attempt to rationalize an environment. Review the following myths to provide that context:

Myth: It's easy to make rationalization decisions early in the process

Accurate rationalization requires a deep knowledge of the workload and associated assets (applications, infrastructure, and data). Most importantly, accurate rationalization decisions take time. We recommend using an incremental rationalization process.

Myth: Cloud adoption has to wait for all workloads to be rationalized

Rationalizing an entire IT portfolio or even a single datacenter can delay the realization of business value by months or even years. Full rationalization should be avoided when possible. Instead, use the Power of 10 approach to release planning to make wise decisions about the next 10 workloads that are slated for cloud adoption.

Myth: Business justification has to wait for all workloads to be rationalized

To develop a business justification for a cloud adoption effort, make a few basic assumptions at the portfolio level. When motivations are aligned to innovation, assume rearchitecture. When motivations are aligned to migration, assume rehost. These assumptions can accelerate the business justification process. Assumptions are then challenged and budgets refined during the assessment phase of each workload's adoption cycle.

Now review the following five Rs of rationalization to familiarize yourself with the long-term process. While developing your cloud adoption plan, choose the option that best aligns with your motivations, business outcomes, and current state environment. The goal in digital estate rationalization is to set a baseline, not to rationalize every workload.

The five Rs of rationalization

The five Rs of rationalization that are listed here describe the most common options for rationalization.

Rehost

Also known as a *lift and shift* migration, a rehost effort moves a current state asset to the chosen cloud provider, with minimal change to overall architecture.

Common drivers might include:

- Reducing capital expense.
- Freeing up datacenter space.
- Achieving rapid return on investment in the cloud.

Quantitative analysis factors:

- VM size (CPU, memory, storage).

- Dependencies (network traffic).
- Asset compatibility.

Qualitative analysis factors:

- Tolerance for change.
- Business priorities.
- Critical business events.
- Process dependencies.

Refactor

Platform as a service (PaaS) options can reduce the operational costs that are associated with many applications. It's a good idea to slightly refactor an application to fit a PaaS-based model.

"Refactor" also refers to the application development process of refactoring code to enable an application to deliver on new business opportunities.

Common drivers might include:

- Faster and shorter updates.
- Code portability.
- Greater cloud efficiency (resources, speed, cost, managed operations).

Quantitative analysis factors:

- Application asset size (CPU, memory, storage).
- Dependencies (network traffic).
- User traffic (page views, time on page, load time).
- Development platform (languages, data platform, middle-tier services).
- Database (CPU, memory, storage, version).

Qualitative analysis factors:

- Continued business investments.
- Bursting options or timelines.
- Business process dependencies.

Rearchitect

Some aging applications aren't compatible with cloud providers because of the architectural decisions that were made when the application was built. In these cases, the application might need to be rearchitected before transformation.

In other cases, applications that are cloud-compatible, but not cloud-native, might create cost efficiencies and operational efficiencies by rearchitecting the solution into a cloud-native application.

Common drivers might include:

- Application scale and agility.
- Easier adoption of new cloud capabilities.
- Mix of technology stacks.

Quantitative analysis factors:

- Application asset size (CPU, memory, storage).

- Dependencies (network traffic).
- User traffic (page views, time on page, load time).
- Development platform (languages, data platform, middle tier services).
- Database (CPU, memory, storage, version).

Qualitative analysis factors:

- Growing business investments.
- Operational costs.
- Potential feedback loops and DevOps investments.

Rebuild

In some scenarios, the delta that must be overcome to carry an application forward can be too large to justify further investment. This is especially true for applications that previously met the needs of a business but are now unsupported or misaligned with the current business processes. In this case, a new code base is created to align with a cloud-native approach.

Common drivers might include:

- Accelerating innovation.
- Building applications faster.
- Reducing operational cost.

Quantitative analysis factors:

- Application asset size (CPU, memory, storage).
- Dependencies (network traffic).
- User traffic (page views, time on page, load time).
- Development platform (languages, data platform, middle tier services).
- Database (CPU, memory, storage, version).

Qualitative analysis factors:

- Declining end-user satisfaction.
- Business processes limited by functionality.
- Potential cost, experience, or revenue gains.

Replace

Solutions are typically implemented by using the best technology and approach available at the time. Sometimes software as a service (SaaS) applications can provide all the necessary functionality for the hosted application. In these scenarios, a workload can be scheduled for future replacement, effectively removing it from the transformation effort.

Common drivers might include:

- Standardizing around industry best practices.
- Accelerating adoption of business-process-driven approaches.
- Reallocating development investments into applications that create competitive differentiation or advantages.

Quantitative analysis factors:

- General operating-cost reductions.

- VM size (CPU, memory, storage).
- Dependencies (network traffic).
- Assets to be retired.
- Database (CPU, memory, storage, version).

Qualitative analysis factors:

- Cost benefit analysis of the current architecture versus a SaaS solution.
- Business process maps.
- Data schemas.
- Custom or automated processes.

Next steps

Collectively, you can apply these five Rs of rationalization to a digital estate to help you make rationalization decisions about the future state of each application.

What is a digital estate?

What is a digital estate?

3/6/2021 • 2 minutes to read • Edit Online

Every modern company has some form of digital estate. Much like a physical estate, a digital estate is an abstract reference to a collection of tangible owned assets. In a digital estate, those assets include virtual machines (VMs), servers, applications, data, and so on. Essentially, a digital estate is the collection of IT assets that power business processes and supporting operations.

The importance of a digital estate is most obvious during the planning and execution of digital transformation efforts. During transformation journeys, the cloud strategy teams use the digital estate to map the business outcomes to release plans and technical efforts. That all starts with an inventory and measurement of the digital assets that the organization owns today.

How can a digital estate be measured?

The measurement of a digital estate changes depending on the desired business outcomes.

- **Infrastructure migrations:** When an organization is inward-facing and seeks to optimize costs, operational processes, agility, or other aspects of their operations, the digital estate focuses on VMs, servers, and workloads.
- **Application innovation:** For customer-focused transformations, the lens is a bit different. The focus should be placed on the applications, APIs, and transactional data that supports the customers. VMs and network appliances often receive less focus.
- **Data-driven innovation:** In today's digitally driven market, it's difficult to launch a new product or service without a strong foundation in data. During cloud-enabled data innovation efforts, the focus is more on the silos of data across the organization.
- **Operational stability:** Businesses are dependent on stable technologies to operate effectively. Near-zero downtime and service reliability are crucial in competitive markets. When operational stability is a priority, the digital estate should be measured on positive or negative impact to stable operations. Business continuity, disaster recovery, and reliability of workloads and each asset are required measures when operational stability is a priority.

After an organization understands the most important form of transformation, digital estate planning becomes much easier to manage.

> Each type of transformation can be measured with any of the above views. Companies commonly complete all these transformations in parallel. We strongly recommend that company leadership and the cloud strategy team agree regarding the transformation that is most important for business success. That understanding serves as the basis for common language and metrics across multiple initiatives.

How can a financial model be updated to reflect the digital estate?

An analysis of the digital estate drives cloud adoption activities. It also informs financial models by providing cloud costing models, which in turn drive return on investment (ROI).

To complete the digital estate analysis, take the following steps:

1. Determine analysis approach.
2. Collect current state inventory.

3. Rationalize the assets in the digital estate.
4. Align assets to cloud offerings to calculate pricing.

Financial models and migration backlogs can be modified to reflect the rationalized and priced estate.

Next steps

Before digital estate planning begins, determine which approach to use.

Approaches to digital estate planning

Approaches to digital estate planning

3/6/2021 • 3 minutes to read • Edit Online

Digital estate planning can take several forms depending on the desired outcomes and size of the existing estate. There are various approaches that you can take. It's important to set expectations regarding the approach early in planning cycles. Unclear expectations often lead to delays associated with additional inventory-gathering exercises. This article outlines three approaches to analysis.

Workload-driven approach

The top-down assessment approach evaluates security aspects. Security includes the categorization of data (high, medium, or low business impact), compliance, sovereignty, and security risk requirements. This approach assesses high-level architectural complexity. It evaluates aspects such as authentication, data structure, latency requirements, dependencies, and application life expectancy.

The top-down approach also measures the operational requirements of the application, such as service levels, integration, maintenance windows, monitoring, and insight. When these aspects have been analyzed and considered, the resulting score reflects the relative difficulty of migrating this application to each cloud platform: IaaS, PaaS, and SaaS.

In addition, the top-down assessment evaluates the financial benefits of the application, such as operational efficiencies, TCO, return on investment, and other appropriate financial metrics. The assessment also examines the seasonality of the application (such as whether there are certain times of the year when demand spikes) and overall compute load.

It also looks at the types of users it supports (casual/expert, always/occasionally logged on), and the required scalability and elasticity. Finally, the assessment concludes by examining business continuity and resiliency requirements, as well as dependencies for running the application if a disruption of service should occur.

> **TIP**
> This approach requires interviews and anecdotal feedback from business and technical stakeholders. Availability of key individuals is the biggest risk to timing. The anecdotal nature of the data sources makes it more difficult to produce accurate cost or timing estimates. Plan schedules in advance and validate any data that's collected.

Asset-driven approach

The asset-driven approach provides a plan based on the assets that support an application for migration. In this approach, you pull statistical usage data from a configuration management database (CMDB) or other infrastructure assessment tools.

This approach usually assumes an IaaS model of deployment as a baseline. In this process, the analysis evaluates the attributes of each asset: memory, number of processors (CPU cores), operating system storage space, data drives, network interface cards (NICs), IPv6, network load balancing, clustering, operating system version, database version (if necessary), supported domains, and third-party components or software packages, among others. The assets that you inventory in this approach are then aligned with workloads or applications for grouping and dependency mapping purposes.

> **TIP**
> This approach requires a rich source of statistical usage data. The time that's needed to scan the inventory and collect data is the biggest risk to timing. The low-level data sources can miss dependencies between assets or applications. Plan for at least one month to scan the inventory. Validate dependencies before deployment.

Incremental approach

We strongly suggest an incremental approach, as we do for many processes in the Cloud Adoption Framework. In the case of digital estate planning, that equates to a multiphase process:

- **Initial cost analysis:** If financial validation is required, start with an asset-driven approach, described earlier, to get an initial cost calculation for the entire digital estate, with no rationalization. This establishes a worst-case scenario benchmark.
- **Migration planning:** After you have assembled a cloud strategy team, build an initial migration backlog using a workload-driven approach that's based on their collective knowledge and limited stakeholder interviews. This approach quickly builds a lightweight workload assessment to foster collaboration.
- **Release planning:** At each release, the migration backlog is pruned and reprioritized to focus on the most relevant business impact. During this process, the next five to ten workloads are selected as prioritized releases. At this point, the cloud strategy team invests the time in completing an exhaustive workload-driven approach. Delaying this assessment until a release is aligned better respects the time of stakeholders. It also delays the investment in full analysis until the business starts to see results from earlier efforts.
- **Execution analysis:** Before migrating, modernizing, or replicating any asset, assess it both individually and as part of a collective release. At this point, the data from the initial asset-driven approach can be scrutinized to ensure accurate sizing and operational constraints.

> **TIP**
> This incremental approach enables streamlined planning and accelerated results. It's important that all parties involved understand the approach to delayed decision making. It's equally important that assumptions made at each stage be documented to avoid loss of details.

Next steps

After an approach is selected, the inventory can be collected.

Gather inventory data

Gather inventory data for a digital estate

5/24/2021 • 2 minutes to read • Edit Online

Developing an inventory is the first step for digital estate planning. In this process, a list of IT assets that support specific business functions are collected for later analysis and rationalization. This article assumes that a bottom-up approach to analysis is most appropriate for planning. For more information, see Approaches to digital estate planning.

Take inventory of a digital estate

The inventory that supports a digital estate changes depending on the desired digital transformation and corresponding transformation journey.

- **Cloud migration:** We often recommend that during a cloud migration, you collect the inventory from scanning tools that create a centralized list of all virtual machines and servers. Some tools can also create network mappings and dependencies, which help define workload alignment.
- **Application innovation:** Inventory during a cloud-enabled application innovation effort begins with the customer. Mapping the customer experience from start to finish is a good place to begin. Aligning that map to applications, APIs, data, and other assets creates a detailed inventory for analysis.
- **Data innovation:** Cloud-enabled data innovation efforts focus on the product or service. An inventory also includes a mapping of the opportunities for disrupting the market, as well as the capabilities needed.
- **Security:** Inventory provides security the understanding to help assess, protect, and monitor the organization's assets.

Accuracy and completeness of an inventory

An inventory is rarely complete in its first iteration. We strongly recommend the cloud strategy team aligns stakeholders and power users to validate the inventory. When possible, use additional tools like network and dependency analysis to identify assets that are being sent traffic, but that are not in the inventory.

Next steps

After an inventory is compiled and validated, it can be rationalized. Inventory rationalization is the next step to digital estate planning.

Rationalize the digital estate

Rationalize the digital estate

6/22/2021 • 10 minutes to read • Edit Online

Cloud rationalization is the process of evaluating assets to determine the best approach to hosting them in the cloud. After you've determined an approach and aggregated an inventory, cloud rationalization can begin. Cloud rationalization discusses the most common rationalization options.

Watch the following video to get a quick overview about completing a comprehensive assessment that will help you plan and prioritize your migration efforts.

Traditional view of rationalization

It's easy to understand rationalization when you visualize the traditional process of rationalization as a complex decision tree. Each asset in the digital estate is fed through a process that results in one of five answers (the five Rs of rationalization). For small estates, this process works well. For larger estates, it's inefficient and can lead to significant delays. Let's examine the process to see why. Then we'll present a more efficient model.

Inventory: A thorough inventory of assets, including applications, software, hardware, operating systems, and system performance metrics, is required for completing a full rationalization by using traditional models.

Quantitative analysis: In the decision tree, quantitative questions drive the first layer of decisions. Common questions include the following:

- Is the asset in use today?
- If so, is it optimized and sized properly?
- What dependencies exist between assets? These questions are vital to the classification of the inventory.

Qualitative analysis: The next set of decisions requires human intelligence in the form of qualitative analysis. Often, the questions that come up here are unique to the solution and can be answered only by business stakeholders and power users. These decisions typically delay the process, slowing things down considerably. This analysis generally consumes 40 to 80 FTE hours per application.

For guidance about building a list of qualitative analysis questions, see Approaches to digital estate planning.

Rationalization decision: In the hands of an experienced rationalization team, the qualitative and quantitative data creates clear decisions. Unfortunately, teams with a high degree of rationalization experience are expensive to hire or take months to train.

Rationalization at enterprise-scale

If this effort is time consuming and daunting for a 50-VM digital estate, imagine the effort that's required to drive business transformation in an environment with thousands of VMs and hundreds of applications. The human effort required can easily exceed 1,500 FTE hours and nine months of planning.

While full rationalization is the end state and a great direction to move in, it seldom produces a high ROI (return on investment) relative to the time and energy that's required.

When rationalization is essential to financial decisions, it's worth considering a professional services organization that specializes in cloud rationalization to accelerate the process. Even then, full rationalization can be a costly and time-consuming effort that delays transformation or business outcomes.

The rest of this article describes an alternative approach, known as incremental rationalization.

Incremental rationalization

The complete rationalization of a large digital estate is prone to risk and can suffer delays because of its complexity. The assumption behind the incremental approach is that delayed decisions stagger the load on the business to reduce the risk of roadblocks. Over time, this approach creates an organic model for developing the processes and experience required to make qualified rationalization decisions more efficiently.

Inventory: Reduce discovery data points

Few organizations invest the time, energy, and expense in maintaining an accurate real-time inventory of the full digital estate. Loss, theft, refresh cycles, and employee onboarding often justify detailed asset tracking of end-user devices. The ROI of maintaining an accurate server and application inventory in a traditional, on-premises datacenter is often low. Most IT organizations have more urgent issues to address than tracking the usage of fixed assets in a datacenter.

In a cloud transformation, inventory directly correlates to operating costs. Accurate inventory data is required for proper planning. Unfortunately, current environmental scanning options can delay decisions by weeks or months. Fortunately, a few tricks can accelerate data collection.

Agent-based scanning is the most frequently cited delay. The robust data that's required for a traditional rationalization can often only be collected with an agent running on each asset. This dependency on agents often slows progress, because it can require feedback from security, operations, and administration functions.

In an incremental rationalization process, an agentless solution could be used for an initial discovery to accelerate early decisions. Depending on the level of complexity in the environment, an agent-based solution might still be required, but it can be removed from the critical path to business change.

Quantitative analysis: Streamline decisions

Regardless of the approach to inventory discovery, quantitative analysis can drive initial decisions and assumptions. This is especially true when trying to identify the first workload or when the goal of rationalization is a high-level cost comparison. In an incremental rationalization process, the cloud strategy team and the cloud adoption teams limit the five Rs of rationalization to two concise decisions and only apply those quantitative factors. This streamlines the analysis and reduces the amount of initial data that's required to drive change.

For example, if an organization is in the midst of an IaaS migration to the cloud, you can assume that most workloads will either be retired or rehosted.

Qualitative analysis: Temporary assumptions

By reducing the number of potential outcomes, it's easier to reach an initial decision about the future state of an asset. When you reduce the options, you also reduce the number of questions asked of the business at this early stage.

For example, if the options are limited to rehosting or retiring, the business needs to answer only one question during initial rationalization, which is whether to retire the asset.

"Analysis suggests that no users are actively using this asset. Is that accurate, or have we overlooked something?" Such a binary question is typically much easier to run through qualitative analysis.

This streamlined approach produces baselines, financial plans, strategy, and direction. In later activities, each asset goes through further rationalization and qualitative analysis to evaluate other options. All assumptions that you make in this initial rationalization are tested before migrating individual workloads.

Challenge assumptions

The outcome of the prior section is a rough rationalization that's full of assumptions. Next, it's time to challenge

some of those assumptions.

Retire assets

In a traditional on-premises environment, hosting small, unused assets seldom causes a significant impact on annual costs. With a few exceptions, FTE effort that's required to analyze and retire the actual asset outweighs the cost savings from pruning and retiring those assets.

When you move to a cloud accounting model, retiring assets can produce significant savings in annual operating costs and up-front migration efforts.

It's not uncommon for organizations to retire 20% or more of their digital estate after completing a quantitative analysis. We recommend conducting further qualitative analysis before taking action. After it's confirmed, retiring those assets can produce the first ROI victory of the cloud migration. This is often one of the biggest cost-saving factors. Therefore, the cloud strategy team should oversee the validation and retirement of assets, in parallel with execution of the Migrate methodology, to achieve an early financial win.

Program adjustments

A company seldom embarks on just one transformation journey. The choice between cost reduction, market growth, and new revenue streams is rarely a binary decision. As such, we recommend that the cloud strategy team work with IT to identify assets on parallel transformation efforts that are outside of the scope of the primary transformation journey.

In the IaaS migration example given in this article:

- Ask the DevOps team to identify assets that are already part of a deployment automation and remove those assets from the core migration plan.
- Ask the data and R&D teams to identify assets that are powering new revenue streams and remove them from the core migration plan.

This program-focused qualitative analysis can be executed quickly and creates alignment across multiple migration backlogs.

You might still need to consider some assets as rehost assets for a while. You can phase in later rationalization after the initial migration.

Select the first workload

Implementing the first workload is key to testing and learning. It's the first opportunity to demonstrate and build a growth mindset.

Business criteria

To ensure business transparency, identify a workload that is supported by a member of the cloud strategy team's business unit. Preferably choose one in which the team has a vested stake and strong motivation to move to the cloud.

Technical criteria

Select a workload that has minimum dependencies and can be moved as a small group of assets. We recommend that you select a workload with a defined testing path to make validation easier.

The first workload is often deployed in an experimental environment with no operational or governance capacity. It's important to select a workload that doesn't interact with secure data.

Qualitative analysis

The cloud adoption teams and the cloud strategy team can work together to analyze this small workload. This collaboration creates a controlled opportunity to create and test qualitative analysis criteria. The smaller population creates an opportunity to survey the affected users, and to complete a detailed qualitative analysis in

a week or less. For common qualitative analysis factors, see the specific rationalization target in the five Rs of rationalization.

Migration

In parallel with continued rationalization, the cloud adoption team can begin migrating the small workload to expand learning in the following key areas:

- Strengthen skills with the cloud provider's platform.
- Define the core services and Azure standards needed to fit the long-term vision.
- Better understand how operations might need to change later in the transformation.
- Understand any inherent business risks and the business's tolerance for those risks.
- Establish a baseline or minimum viable product (MVP) for governance based on the business's risk tolerance.

Release planning

While the cloud adoption team is executing the migration or implementation of the first workload, the cloud strategy team can begin prioritizing the remaining applications and workloads.

Power of 10

The traditional approach to rationalization attempts to meet all foreseeable needs. Fortunately, a plan for every application is often not required to start a transformation journey. In an incremental model, the Power of 10 approach provides a good starting point. In this model, the cloud strategy team selects the first 10 applications to be migrated. Those ten workloads should contain a mixture of simple and complex workloads.

Build the first backlogs

The cloud adoption teams and the cloud strategy team can work together on the qualitative analysis for the first 10 workloads. This effort creates the first prioritized migration backlog and the first prioritized release backlog. This method enables the teams to iterate on the approach and provides sufficient time to create an adequate process for qualitative analysis.

Mature the process

After the two teams agree on the qualitative analysis criteria, assessment can become a task within each iteration. Reaching consensus on assessment criteria usually requires two to three releases.

After the assessment has moved into the incremental execution process of migration, the cloud adoption team can iterate faster on assessment and architecture. At this stage, the cloud strategy team is also abstracted, reducing the drain on their time. This also enables the cloud strategy team to focus on prioritizing the applications that are not yet in a specific release, ensuring tight alignment with changing market conditions.

Not all of the prioritized applications will be ready for migration. Sequencing is likely to change as the team does deeper qualitative analysis and discovers business events and dependencies that might prompt reprioritization of the backlog. Some releases might group together a small number of workloads. Others might just contain a single workload.

The cloud adoption team is likely to run iterations that don't produce a complete workload migration. The smaller the workload, and the fewer dependencies, the more likely a workload is to fit into a single sprint or iteration. For this reason, we recommend that the first few applications in the release backlog be small and contain few external dependencies.

End state

Over time, the cloud adoption team and the cloud strategy team together complete a full rationalization of the inventory. This incremental approach enables the teams to get continually faster at the rationalization process. It also helps the transformation journey to yield tangible business results sooner, without as much upfront analysis effort.

In some cases, the financial model might be too tight to make a decision without additional rationalization. In such cases, you might need a more traditional approach to rationalization.

Next steps

The output of a rationalization effort is a prioritized backlog of all assets that are affected by the chosen transformation. This backlog is now ready to serve as the foundation for costing models of cloud services.

Align cost models with the digital estate

Align cost models with the digital estate to forecast cloud costs

3/6/2021 • 2 minutes to read • Edit Online

After you've rationalized a digital estate, you can align it to equivalent costing models with the chosen cloud provider. Discussing cost models is difficult without focusing on a specific cloud provider. To provide tangible examples in this article, Azure is the assumed cloud provider.

Azure pricing tools help you manage cloud spend with transparency and accuracy, so you can make the most of Azure and other clouds. Providing the tools to monitor, allocate, and optimize cloud costs, empowers customers to accelerate future investments with confidence.

- Azure Migrate: Azure Migrate is perhaps the most cost effective approach to cost model alignment. This tool allows for digital estate inventory, limited rationalization, and cost calculations in one tool.
- Total cost of ownership (TCO) calculator: Lower the total cost of ownership of your on-premises infrastructure with the Azure cloud platform. Use the Azure TCO calculator to estimate the cost savings you can realize by migrating your application workloads to Azure. Provide a brief description of your on-premises environment to get an instant report.
- Azure pricing calculator: Estimate your expected monthly bill by using our pricing calculator. Track your actual account usage and bill at any time using the billing portal. Set up automatic email billing alerts to notify you if your spend goes above an amount you configure.
- Azure Cost Management + Billing: Azure Cost Management + Billing is a cost management solution that helps you use and manage Azure and other cloud resources effectively. Collect cloud usage and billing data through application program interfaces (APIs) from Azure, Amazon Web Services, and Google Cloud Platform. With that data, gain full visibility into resource consumption and costs across cloud platforms in a single, unified view. Continuously monitor cloud consumption and cost trends. Track actual cloud spending against your budget to avoid overspending. Detect spending anomalies and usage inefficiencies. Use historical data to improve your forecasting accuracy for cloud usage and expenditures.

Measure business outcomes with AppDynamics

5/24/2021 • 4 minutes to read • Edit Online

Measuring and quantifying successful business outcomes is a crucial part of any cloud adoption strategy. Understanding an application's performance and user experience is key to measuring those business outcomes. However, accurately measuring the correlation between application performance, user experience, and business impact is often difficult, inaccurate, and time consuming.

AppDynamics can provide business insights for most use cases, and many organizations start a comprehensive cloud adoption strategy with the following use cases:

- A pre-migration and post-migration comparison
- Business health
- Release validation
- Segment health
- User journeys
- Business journeys
- Conversion funnels

This guidance will focus on how to measure business outcomes of a pre-migration and post-migration comparison and how to accelerate and reduce risk for a migration during your cloud adoption journey.

How AppDynamics works

Prior to migration, a small, lightweight agent is deployed alongside your applications. Agents are purpose-built for various languages such as .NET, Java, and Node.js. The agent collects performance and diagnostic data during the migration and sends it to a controller to correlate and analyze the information. Controllers can reside in a fully managed AppDynamics environment, or the customer can choose to manage them in Azure. Key user experiences are identified as *business transactions*, which help you to discover the baseline for normal application or business performance. Whether they're traditional server infrastructure, database, middleware components, on-premises, or in the cloud, all application components and dependencies are identified in real time for the entire application and each business transaction.

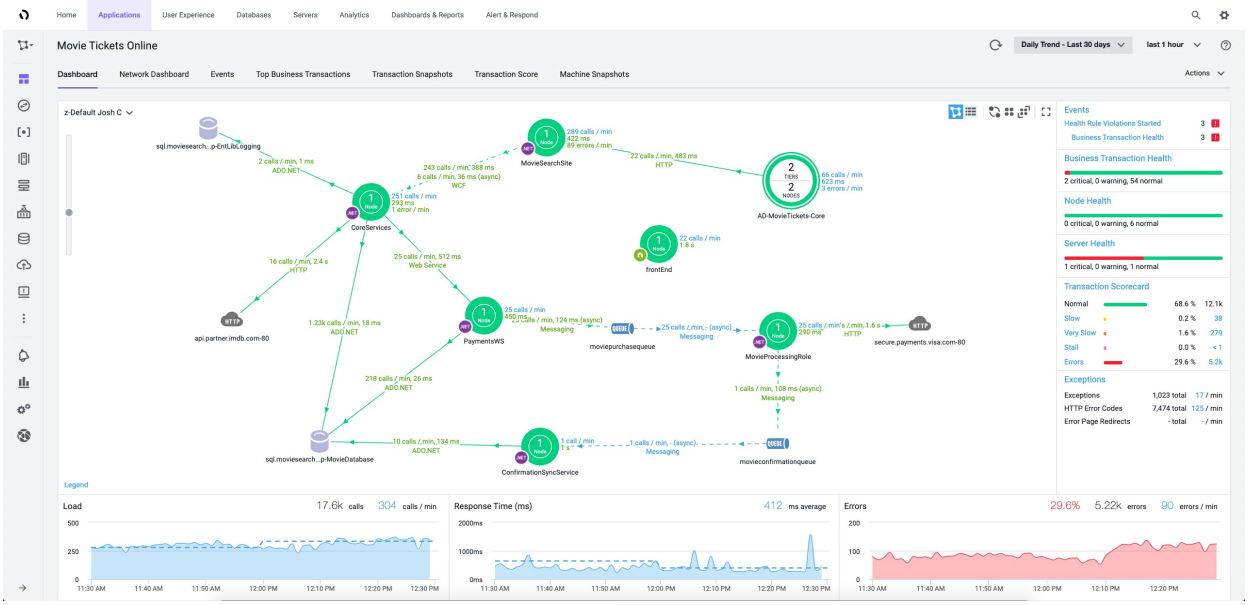


Figure 1: An AppDynamics flow map.

AppDynamics identifies business metrics

AppDynamics helps you to define business value for your applications, identify the key metrics that they should meet to retain their value, and verify if they're fulfilling their target business outcomes. AppDynamics agents collect these data points and traditional application performance metrics like response time and memory utilization in real time, directly from the application, and without any changes to code.

Business metrics are closely related to business outcomes. Many organizations have complex metrics that measure unique business outcomes, and these outcomes can range from fiscal and agility-related to performance and customer engagement goals. AppDynamics collects the metrics that are specific and useful to your organization, and those metrics can contribute to current business operations before and after migrating workloads to Azure.

Example:

A company that sells widgets from an online marketplace has identified the following key business transactions within their web application:

- Landing page
- Add to cart
- Shipping
- Billing
- Confirm order

These types of business transactions are common to e-commerce applications. A conversion funnel is the journey that a user takes through these pages, and it directly leads to sales revenue on the company's platform. When users abandon the journey because of poor page performance or errors, this directly impacts the company's underlying profit.

Additionally, the company has identified the following key business metrics:

- Cart totals
- Customer segments
- Customer locations

Combining application and business performance metrics helps to clearly demonstrate how their application's performance relates to their underlying profit. This level of visibility and types of insights will be vital during migrations.

Configurable dashboards are one of many AppDynamics tools that visualize these insights. In this real-time example, we see the overall conversion funnel and the impact on individual page performance against abandoners alongside shopping cart totals, customer segment, location, and general revenue details.

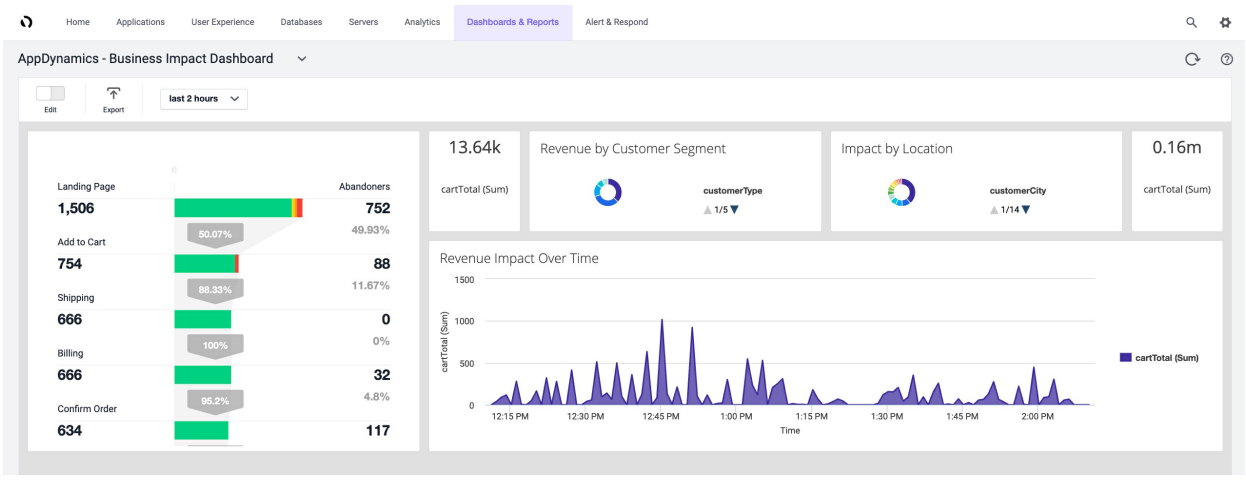


Figure 2: AppDynamics business impact dashboard.

Resources to help identify business metrics

The strategy and business outcomes sections of the Cloud Adoption Framework provide the guidance and strategies to help you identify business outcomes for your organization.

Pre- and post-migration comparison

The cloud offers vast benefits and potential, but the first steps of a migration are often unclear and full of risk. A successful migration must be evaluated by more criteria than the capability for a successful deployment. Understanding the user experience and business performance before and after the cloud migration helps you to adjust and stabilize both as needed, which helps produce successful business outcomes while reinforcing the value that Azure provides throughout your migration journey.

To build on the foundation of understanding how AppDynamics provides business and application metrics, compare those metrics before and after a migration to evaluate its success and if the target business outcomes are met.

Example:

Movie Tickets Online, a fictitious online movie ticket seller, is working to retire their existing datacenters and move their workloads to Azure. Their capacity issues have led to poor business transaction performance, and they look forward to the performance optimizations and capacity in Azure.

In addition to improving performance, they want to ensure that the business outcomes of improving their sales funnels and growing their revenue will be met. As part of their migration, they deployed AppDynamics to their existing on-premises environments to clearly understand their current performance. As part of the cloud deployment, Movie Tickets Online can use AppDynamics native integration with Azure to understand post-migration performance and business outcomes.

In this case, they were able to see an increase in conversion rates from 48 to 79 percent and improvements to underlying performance, response time, and ticket sales volume.

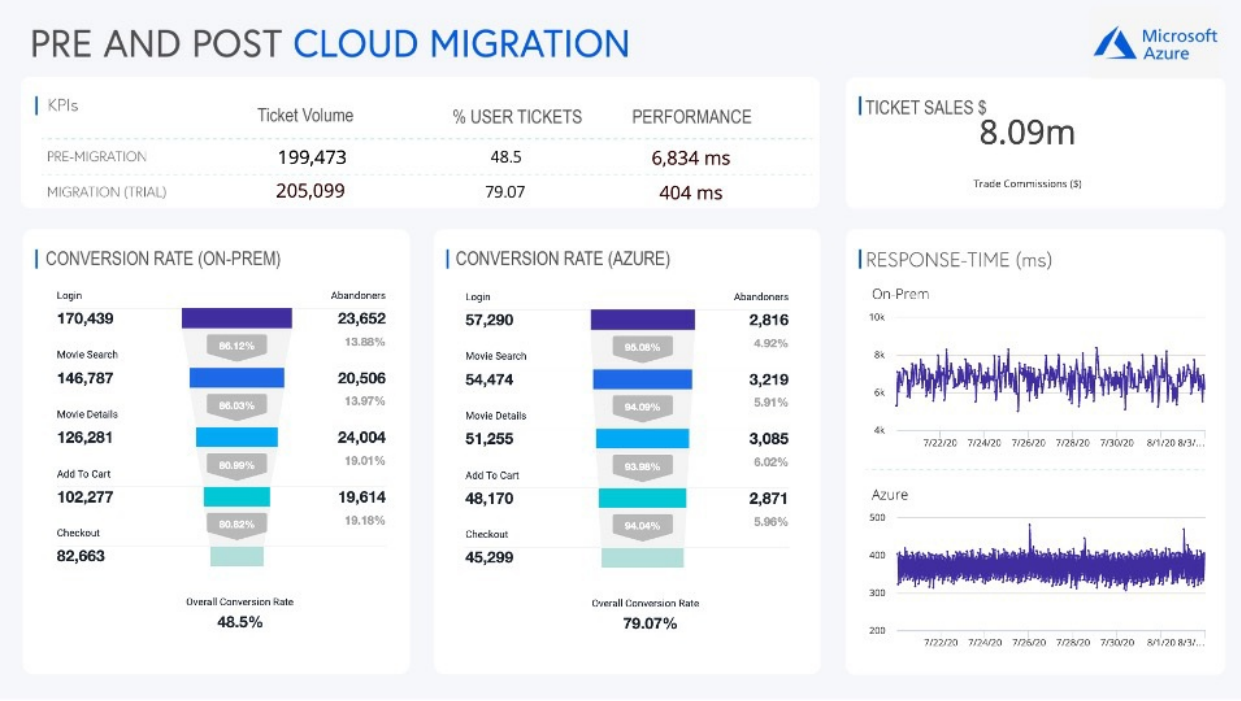


Figure 3: AppDynamics migration comparison.

Next steps

AppDynamics gives organizations the unique ability to measure the business outcomes during their cloud

adoption strategy. Visit AppDynamics to learn more about AppDynamics with Azure.

Initial organization alignment

3/6/2021 • 2 minutes to read • Edit Online

The most important aspect of any cloud adoption plan is the alignment of people who will make the plan a reality. No plan is complete until you understand its people-related aspects.

True organizational alignment takes time. It will become important to establish long-term organizational alignment, especially as cloud adoption scales across the business and IT culture. Alignment is so important that an entire section has been dedicated to it in the Organize methodology of the Cloud Adoption Framework.

Full organization alignment is not a required component of the cloud adoption plan. However, some initial organization alignment is needed. This article outlines a best-practice starting point for organizational alignment. The guidance here can help complete your plan and get your teams ready for cloud adoption. When you're ready, you can use the organization alignment section to customize this guidance to fit your organization.

Initial best-practice structure

To create a balance between speed and control, we recommend that during cloud adoption, at a minimum, you have people accountable for *cloud adoption* and *cloud governance*. This might be a team of people sharing responsibilities for each of these areas, or *capabilities*. It might also be individual people who are both accountable for the outcomes and responsible for the work. In either scenario, cloud adoption and cloud governance are two capabilities that involve natural friction between moving quickly and reducing risks. Here's how the two teams fit together:



It's fairly intuitive that cloud adoption tasks require people to execute those tasks. So, few people are surprised that a cloud adoption team is a requirement. However, those who are new to the cloud may not fully appreciate the importance of a cloud governance team. This challenge often occurs early in adoption cycles. The cloud governance team provides the necessary checks and balances to ensure that cloud adoption doesn't expose the business to any new risks. When risks must be taken, this team ensures that proper processes and controls are implemented to mitigate or govern those risks.

For more information about cloud adoption, cloud governance, and other such capabilities, see the brief section on understanding required cloud capabilities.

Map people to capabilities

Assuming that the suggested structure aligns to your cloud adoption plan, the next step is to map specific people to the necessary capabilities. To do so, answer the following questions:

- What person (or group of people) will be responsible for completing technical tasks in the cloud adoption plan?
- What person will be accountable for the team's ability to deliver technical changes?
- What person (or group of people) will be responsible for implementing protective governance mechanisms?
- What person will be accountable for the defining those governance controls?

- Are there other capabilities or people that will have accountability or responsibility within the cloud adoption plan?

After you've documented the answers to these questions, you can establish plans for skills readiness to define plans to prepare these people for forthcoming work.

Next steps

Learn how to plan for cloud adoption.

Plan for cloud adoption

Adapt existing roles, skills, and processes for the cloud

3/6/2021 • 3 minutes to read • Edit Online

At each phase of the IT industry's history, the most notable changes have often been marked by changes in staff roles. One example is the transition from mainframe computing to client/server computing. The role of the computer operator during this transition has largely disappeared, replaced by the system administrator role. When virtualization arrived, the requirement for individuals working with physical servers was replaced with a need for virtualization specialists.

Roles will likely change as institutions similarly shift to cloud computing. For example, datacenter specialists might be replaced with cloud administrators or cloud architects. In some cases, though IT job titles haven't changed, the daily work of these roles has changed significantly.

IT staff members might feel anxious about their roles and positions because they realize that they need a different set of skills to support cloud solutions. But agile employees who explore and learn new cloud technologies shouldn't fear. They can lead the adoption of cloud services and help the organization learn and embrace the associated changes.

For guidance on building a new skill set, see the skills readiness path.

Capture concerns

As the organization prepares for a cloud adoption effort, each team should document staff concerns as they arise by identifying:

- The type of concern. For example, workers might be resistant to the changes in job duties that come with the adoption effort.
- The impact if the concern isn't addressed. For example, resistance to adoption might result in workers being slow to execute the required changes.
- The area equipped to address the concern. For example, if workers in the IT department are reluctant to acquire new skills, the IT stakeholder's area is best equipped to address this concern. Identifying the area might be clear for some concerns. In these cases, you might need to escalate to executive leadership.

IT staff members commonly have concerns about acquiring the training needed to support expanded functions and new duties. Learning the training preferences of the team helps you prepare a plan. It also allows you to address these concerns.

Identify gaps

Identifying gaps is another important aspect of organization readiness. A *gap* is a role, skill, or process that is required for your digital transformation but doesn't currently exist in your enterprise.

1. Enumerate the responsibilities that come with the digital transformation. Emphasize new responsibilities and existing responsibilities to be retired.
2. Identify the area that aligns with each responsibility. For each new responsibility, check how closely it aligns with the area. Some responsibilities might span several areas. This crossover represents an opportunity for better alignment that you should document as a concern. In the case where no area is identified as being responsible, document this gap.
3. Identify the skills necessary to support each responsibility, and check if your enterprise has existing resources

with those skills. Where there are no existing resources, determine the training programs or talent acquisition necessary to fill the gaps. Also determine the deadline by which you must support each responsibility to keep your digital transformation on schedule.

4. Identify the roles that will execute these skills. Some of your existing workforce will assume parts of the roles. In other cases, entirely new roles might be necessary.

Partner across teams

The skills necessary to fill the gaps in your organization's digital transformation are typically not confined to a single role or even a single department. Skills will have relationships and dependencies that can span a single role or multiple roles. Those roles might exist in several departments. For example, a workload owner might require someone in an IT role to provision core resources like subscriptions and resource groups.

These dependencies represent new processes that your organization implements to manage the workflow among roles. The preceding example shows several types of processes that support the relationship between the workload owner and the IT role. For instance, you can create a workflow tool to manage the process or use an email template.

Track these dependencies and make note of the processes that will support them. Also note whether the processes currently exist. For processes that require tooling, ensure that the timeline for deploying any tools aligns with the overall digital-transformation schedule.

Next steps

Ensuring proper support for the translated roles is a team effort. To act on this guidance, review the organizational readiness overview to identify the right team structures and participants.

Identify the right team structures

Get started on a skills readiness path

3/6/2021 • 2 minutes to read • Edit Online

IT staff members might feel anxious about their roles and positions as they realize a different set of skills is needed to support cloud solutions. Agile employees who explore and learn new cloud technologies don't need to have that fear. They can lead the adoption of cloud services by helping the organization understand and embrace the associated changes.

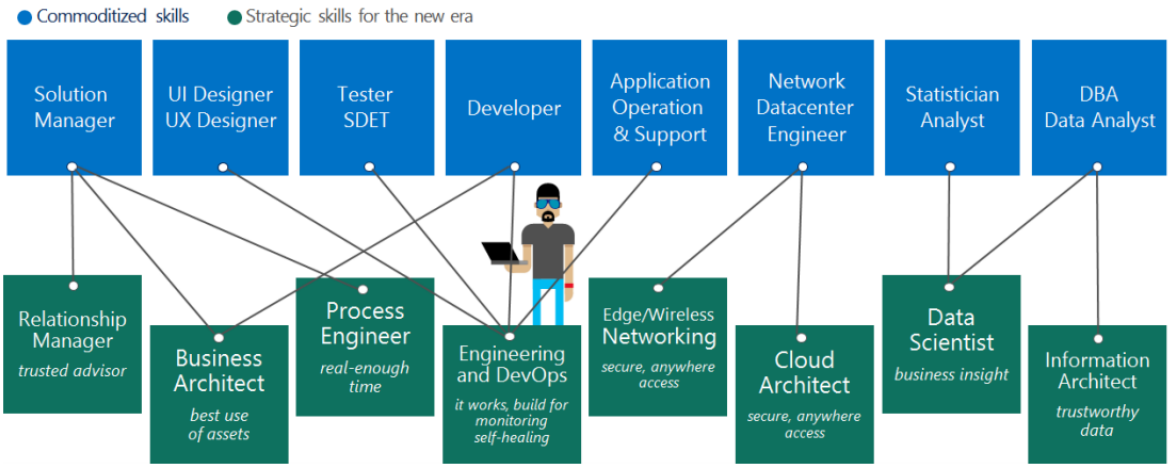


Figure 1: Mapping of skills to IT roles in a cloud-hosted environment.

The Cloud Adoption Framework guides readers through the full adoption lifecycle. Throughout this framework, readers are provided opportunities to build necessary skills. To help you get started on this journey, skills-readiness articles are included in the following outline for easier access. Each of the following links maps to the skills required to be successful in each of those adoption phases.

- **Strategy**: Develop the skills needed to prepare an actionable migration plan. This includes business justification and other required business-planning skills.
- **Plan**: Develop the skills needed to prepare an actionable migration plan. This includes business justification and other required business-planning skills.
- **Ready**: Develop the skills needed to prepare the business, culture, people, and environment for coming changes.
- **Adopt**: Adoption skills are aligned to various technical efforts:
  - **Migrate**: Gain the skills required to implement the cloud migration plan.
  - **Innovate**: Gain the skills needed to deliver innovative new solutions.
- **Operate**: Skills related to the operating model for cloud adoption are aligned to various opportunities to gain skills:
  - **Govern**: Gain the skills needed to govern the cloud environment.
  - **Manage**: Gain the skills needed to manage a cloud environment.
  - **Monitor**: Gain the skills needed to monitor a cloud environment.

Each of the previous learning paths shares opportunities across multiple media types to maximize knowledge acquisition. As you evolve your learning paths take a view at how your employees and learners will want to get certification. The following resource will help map out the certification paths from understanding the portfolio

available to the overview of the certification, the suggested steps and recommended training needed to certify. Azure Certification Deck.

Microsoft Learn

Microsoft Learn is a new approach to learning. Readiness for the new skills and responsibilities that come with cloud adoption doesn't come easily. Microsoft Learn provides a more rewarding approach to hands-on learning that helps you achieve your goals faster. Earn points and levels, and achieve more!

Examples of tailored learning paths on Microsoft Learn that align to the Plan methodology of the Cloud Adoption Framework include:

Evolve your DevOps practices: DevOps is the union of people, process, and products to enable continuous delivery of value to your end users. Azure DevOps is a set of services that gives you the tools you need to do just that. With Azure DevOps, you can build, test, and deploy any application, either in the cloud or on premises.

Azure for the data engineer: Explore how the world of data has evolved and how the advent of cloud technologies is providing new opportunities for business to explore. You will learn the various data platform technologies that are available, and how a data engineer can take advantage of this technology to an organization's benefit.

Learn more

To discover more learning paths, browse the Microsoft Learn catalog. Use the roles filter to align learning paths with your role.

Plan for cloud adoption

3/6/2021 • 2 minutes to read • Edit Online

A plan is an essential requirement for a successful cloud adoption. A cloud adoption plan is an iterative project plan that helps a company transition from traditional IT approaches to transformation over to modern, agile approaches. This article series outlines how a cloud adoption plan helps companies balance their IT portfolio and manage transitions over time. Through this process, business objectives can be clearly translated into tangible technical efforts. Those efforts can then be managed and communicated in ways that make sense to business stakeholders. However, adopting such a process may require some changes to traditional project-management approaches.

Align strategy and planning

Cloud adoption plans start with a well-defined strategy. At a minimum, the strategy should outline the motivations, business outcomes, and business justifications for cloud adoption. Those positive returns are then balanced by the effort required to realize them.

The effort starts with the digital estate (proposed or existing), which translates the strategy into more tangible workloads and assets. You can then map these tangible elements to technical work. From there, skilled people in a proper organizational structure can execute the technical work. The cloud adoption plan combines these topics into one plan that can be forecasted, budgeted, implemented, and managed by means of agile project-management practices. This article series helps you build the plan and provides a few templates to make the job easier.

Transition from sequential to iterative planning

Planning for cloud adoption can be a significant change for some organizations. IT organizations have long focused on the application of linear or sequential models of project management, like the waterfall model. In traditional IT, this approach was entirely logical. Most large IT projects started with a procurement request to acquire expensive hardware resources. Capital expense requests, budget allocations, and equipment acquisition often represented a large percentage of project execution. And after it was acquired, the hardware itself became a constraint on what could be delivered.

The acquisition models of the cloud change the core dependencies that made a sequential model necessary. The replacement of acquisition cycles with an operating-expense approach helps businesses move more quickly and with smaller financial commitments. This approach helps teams to engage in projects before all requirements are well known. It also creates room for a growth mindset, which frees the team to experiment, learn, and deliver without artificial constraints. For all these reasons and more, we highly recommend that teams use agile or iterative approaches to cloud adoption planning.

Build your cloud adoption plan

This article series walks through each step of translating strategy and effort into an actionable cloud adoption plan:

1. **Prerequisites:** Confirm that all prerequisite steps have been completed before you create your plan.
2. **Define and prioritize workloads:** Prioritize your first 10 workloads to establish an initial adoption backlog.
3. **Align assets to workloads:** Identify which assets (proposed or existing) are required to support the prioritized workloads.

4. **Review rationalization decisions:** Review rationalization decisions to refine adoption path decisions: migrate or innovate.
5. **Establish iterations and release plans:** *Iterations* are the time blocks allocated to do work. *Releases* are the definition of the work to be done before triggering a change to production processes.
6. **Estimate timelines:** Establish rough timelines for release planning purposes, based on initial estimates.

Next steps

Before building your cloud adoption plan, ensure that all necessary prerequisites are in place.

Review prerequisites

Prerequisites for an effective cloud adoption plan

3/6/2021 • 2 minutes to read • Edit Online

A plan is only as effective as the data that's put into it. For a cloud adoption plan to be effective, there are two categories of input: *strategic* and *tactical*. The following sections outline the minimum data points required in each category.

Strategic inputs

Accurate strategic inputs ensure that the work being done contributes to achievement of business outcomes. The strategy section of the Cloud Adoption Framework provides a series of exercises to develop a clear strategy. The outputs of those exercises feed the cloud adoption plan. Before developing the plan, ensure that the following items are well defined as a result of those exercises:

- **Clear motivations:** Why are we adopting the cloud?
- **Defined business outcomes:** What results do we expect to see from adopting the cloud?
- **Business justification:** How will the business measure success?

Every member of the team that implements the cloud adoption plan should be able to answer these three strategic questions. Managers and leaders who are accountable for implementation of the plan should understand the metrics behind each question and any progress toward realizing those metrics.

Tactical inputs

Accurate tactical inputs ensure that the work can be planned accurately and managed effectively. The plan section of the Cloud Adoption Framework provides a series of exercises to develop planning artifacts before you develop your plan. These artifacts provide answers to the following questions:

- **Digital estate rationalization:** What are the top 10 priority workloads in the adoption plan? How many additional workloads are likely to be in the plan? How many assets are being considered as candidates for cloud adoption? Are the initial efforts focused more on migration or innovation activities?
- **Organization alignment:** Who will do the technical work in the adoption plan? Who is accountable for adherence to governance and compliance requirements?
- **Skills readiness:** How many people are allocated to perform the required tasks? How well are their skills aligned to cloud adoption efforts? Are partners aligned to support the technical implementation?

These questions are essential to the accuracy of the cloud adoption plan. At a minimum, the questions about digital estate rationalization must be answered to create a plan. To provide accurate timelines, the questions about organization and skills are also important.

Next steps

Define your cloud adoption plan by deploying the template to Azure DevOps Services.

Define your cloud adoption plan using the template

Cloud adoption plan and Azure DevOps

8/4/2021 • 3 minutes to read • Edit Online

Azure DevOps is the set of cloud-based tools for Azure customers who manage iterative projects. It also includes tools for managing deployment pipelines and other important aspects of DevOps.

In this article, you'll learn how to quickly deploy a backlog to Azure DevOps using a template. This template aligns cloud adoption efforts to a standardized process based on the guidance in the Cloud Adoption Framework.

Create your cloud adoption plan

To deploy the cloud adoption plan, open the Azure DevOps demo generator. This tool will deploy the template to your Azure DevOps tenant. Using the tool requires the following steps:

1. Verify that the **Selected Template** field is set to **Cloud Adoption Plan**. If it isn't, select **Choose template** to choose the right template.
2. Select your Azure DevOps organization from the **Select Organization** drop-down list box.
3. Enter a name for your new project. The cloud adoption plan will have this name when it's deployed to your Azure DevOps tenant.
4. Select **Create Project** to create a new project in your tenant, based on the strategy and plan template. A progress bar show your progress toward deploying the project.
5. When deployment is finished, select **Navigate to project** to see your new project.

After your project has been created, continue through this article series to learn how to modify the template to align to your cloud adoption plan.

For additional support and guidance on this tool, see Azure DevOps Services demo generator.

Bulk edit the cloud adoption plan

When the plan project has been deployed, you can use Microsoft Excel to modify it. It's much easier to create new workloads or assets in the plan by using Microsoft Excel than by using the Azure DevOps browser experience.

To prepare your workstation for bulk editing, see Bulk add or modify work items with Microsoft Excel.

Some users may want to use Microsoft Project to track their tasks, create backlog and assign resource. Here are the steps to connect Microsoft Project to Azure DevOps.

Use the cloud adoption plan

The cloud adoption plan organizes activities by activity type:

- **Epics:** An *epic* represents an overall phase of the cloud adoption lifecycle.
- **Features:** Features are used to organize specific objectives within each phase. For instance, migration of a specific workload would be one feature.
- **User stories:** User stories group work into logical collections of activities based on a specific goal.
- **Tasks:** Tasks are the actual work to be done.

At each layer, activities are then sequenced based on dependencies. Activities are linked to articles in the Cloud Adoption Framework to clarify the objective or task at hand.

The clearest view of the cloud adoption plan comes from the **Epics** backlog view. For help with changing to the **Epics** backlog view, see the article on viewing a backlog. From this view, it's easy to plan and manage the work required to complete the current phase of the adoption lifecycle.

> **NOTE**
> The current state of the cloud adoption plan focuses heavily on migration efforts. Tasks related to governance, innovation, or operations must be populated manually.

Align the cloud adoption plan

The overview pages for the Strategy methodology and the Plan methodology each refer to the strategy and plan template. That template organizes the decisions and data points that will align the template for the cloud adoption plan with your specific plans for adoption. Considering completing the exercises in the Strategy methodology and the Plan methodology before aligning your new project.

The following articles support alignment of the cloud adoption plan:

- Workloads: Align features within the cloud migration epic to capture each workload to be migrated or modernized. Add and modify those features to capture the effort to migrate your top 10 workloads.
- Assets: Each asset (virtual machine, application, or data) is represented by the user stories under each workload. Add and modify those user stories to align with your digital estate.
- Rationalization: As each workload is defined, the initial assumptions about that workload can be challenged. This might result in changes to the tasks under each asset.
- Create release plans: Iteration paths establish release plans by aligning efforts with various releases and iterations.
- Establish timelines: Defining start and end dates for each iteration creates a timeline to manage the overall project.

These five articles help with each of the alignment tasks required to start managing your adoption efforts. The next step gets you started on the alignment exercise.

Next steps

Start aligning your plan project by defining and prioritizing workloads.

Define and prioritize workloads

Define and prioritize workloads for a cloud adoption plan

3/6/2021 • 5 minutes to read • Edit Online

Establishing clear, actionable priorities is one of the secrets to successful cloud adoption. The natural temptation is to invest time in defining all workloads that could potentially be affected during cloud adoption. But that's counterproductive, especially early in the adoption process.

Instead, we recommend that your team focus on thoroughly prioritizing and documenting the first 10 workloads. After implementation of the adoption plan begins, the team can maintain a list of the next 10 highest-priority workloads. This approach provides enough information to plan for the next few iterations.

Limiting the plan to 10 workloads encourages agility and alignment of priorities as business criteria change. This approach also makes room for the cloud adoption team to learn and to refine estimates. Most important, it removes extensive planning as a barrier to effective business change.

What is a workload?

In the context of a cloud adoption, a workload is a collection of IT assets (servers, VMs, applications, data, or appliances) that collectively support a defined process. Workloads can support more than one process. Workloads can also depend on other shared assets or larger platforms. However, a workload should have defined boundaries regarding the dependent assets and the processes that depend upon the workload. Often, workloads can be visualized by monitoring network traffic among IT assets.

Prerequisites

The strategic inputs from the prerequisites list make the following tasks much easier to accomplish. For help with gathering the data discussed in this article, review the prerequisites.

Initial workload prioritization

During the process of incremental rationalization, your team should agree on a Power of 10 approach, consisting of 10 priority workloads. These workloads serve as an initial boundary for adoption planning.

If you decide that a digital estate rationalization isn't needed, we recommend that the cloud adoption teams and the cloud strategy team agree on a list of 10 applications to serve as the initial focus of the migration. We recommend further that these 10 workloads contain a mixture of simple workloads (fewer than 10 assets in a self-contained deployment) and more complex workloads. Those 10 workloads will start the workload prioritization process.

> **NOTE**
> The Power of 10 approach serves as an initial boundary for planning, to focus the energy and investment in early-stage analysis. However, the act of analyzing and defining workloads is likely to cause changes in the list of priority workloads.

Add workloads to your cloud adoption plan

In the previous article, Cloud adoption plan and Azure DevOps, you created a cloud adoption plan in Azure DevOps.

You can now represent the workloads in the Power of 10 list in your cloud adoption plan. The easiest way to do this is via bulk editing in Microsoft Excel. To prepare your workstation for bulk editing, see Bulk add or modify work items with Microsoft Excel.

Step 5 in that article tells you to select **Input list**. Instead, select **Query list**. Then, from the **Select a Query** drop-down list, select the **Workload Template** query. That query loads all the efforts related to the migration of a single workload into your spreadsheet.

After the work items for the workload template are loaded, follow these steps to begin adding new workloads:

1. Copy all the items that have the **Workload Template** tag in the far right column.
2. Paste the copied rows below the last line item in the table.
3. Change the title cell for the new feature from **Workload Template** to the name of your new workload.
4. Paste the new workload name cell into the tag column for all rows below the new feature. Be careful to not change the tags or name of the rows related to the actual **Workload Template** feature. You will need those work items when you add the next workload to the cloud adoption plan.
5. Skip to step 8 in the bulk-editing instructions to publish the worksheet. This step creates all the work items required to migrate your workload.

Repeat steps 1 through 5 for each of the workloads in the Power of 10 list.

Define workloads

After initial priorities have been defined and workloads have been added to the plan, each of the workloads can be defined via deeper qualitative analysis. Before including any workload in the cloud adoption plan, try to provide the following data points for each workload.

Business inputs

DATA POINT	DESCRIPTION	INPUT
Workload name	What is this workload called?	
Workload description	In one sentence, what does this workload do?	
Adoption motivations	Which of the cloud adoption motivations are affected by this workload?	
Primary sponsor	Of those stakeholders affected, who is the primary sponsor requesting the preceding motivations?	
Business impact	What is the business impact of this workload?	
Application impact	What impact does this application have on business processes?	
Data impact	What impact does the data have on the business?	
Business unit	Which business unit is responsible for the cost of this workload?	

DATA POINT	DESCRIPTION	INPUT
Business processes	Which business processes will be affected by changes to the workload?	
Business teams	Which business teams will be affected by changes?	
Business stakeholders	Are there any executives whose business will be affected by changes?	
Business outcomes	How will the business measure the success of this effort?	
Metrics	What metrics will be used to track success?	
Compliance	Are there any third-party compliance requirements for this workload?	
Application owners	Who is accountable for the business impact of any applications associated with this workload?	
Business freeze periods	Are there any times during which the business will not permit change?	
Geographies	Are any geographies affected by this workload?	

Technical inputs

DATA POINT	DESCRIPTION	INPUT
Adoption approach	Is this adoption a candidate for migration or innovation?	
Application ops lead	List the parties responsible for performance and availability of this workload.	
SLAs	List any service-level agreements (RTO/RPO requirements).	
Criticality	List the current application criticality.	
Data classification	List the classification of data sensitivity.	
Operating geographies	List any geographies in which the workload is or should be hosted.	
Applications	Specify an initial list or count of any applications included in this workload.	

DATA POINT	DESCRIPTION	INPUT
VMs	Specify an initial list or count of any VMs or servers included in the workload.	
Data sources	Specify an initial list or count of any data sources included in the workload.	
Dependencies	List any asset dependencies not included in the workload.	
User traffic geographies	List geographies that have a significant collection of user traffic.	

Confirm priorities

Based on the assembled data, the cloud strategy team and the cloud adoption team should meet to reevaluate priorities. Clarification of business data points might prompt changes in priorities. Technical complexity or dependencies might result in changes related to staffing allocations, timelines, or sequencing of technical efforts.

After a review, both teams should be comfortable with confirming the resulting priorities. This set of documented, validated, and confirmed priorities is the prioritized cloud adoption backlog.

Next steps

For any workload in the prioritized cloud adoption backlog, the team is now ready to align assets.

Align assets for prioritized workloads

Align assets to prioritized workloads

3/6/2021 • 2 minutes to read • Edit Online

Workload is a conceptual description of a collection of assets: VMs, applications, and data sources. The previous article, Define and prioritize, provided guidance for collecting the data that will define the workload. Before migration, a few of the technical inputs in that list require additional validation. This article helps with validation of the following inputs:

- **Applications:** List any applications included in this workload.
- **VMs and servers:** List any VMs or servers included in the workload.
- **Data sources:** List any data sources included in the workload.
- **Dependencies:** List any asset dependencies not included in the workload.

There are several options for assembling this data. The following are a few of the most common approaches.

Alternative inputs: Migrate, modernize, innovate

The objective of the preceding data points is to capture relative technical effort and dependencies as an aid to prioritization. Depending on the transition you want, you may need to gather alternative data points to support proper prioritization.

Migrate: For pure migration efforts, the existing inventory and asset dependencies serve as a fair measure of relative complexity.

Modernize: When the goal for a workload is to modernize applications or other assets, these data points are still solid measures of complexity. However, it might be wise to add an input for modernization opportunities to the workload documentation.

Innovate: When data or business logic is undergoing material change during a cloud adoption effort, it's considered an *innovate* type of transformation. The same is true when you're creating new data or new business logic. For any innovate scenarios, the migration of assets will likely represent the smallest amount of effort required. For these scenarios, the team should devise a set of technical data inputs to measure relative complexity.

Azure Migrate

Azure Migrate provides a set of grouping functions that can speed up the aggregation of applications, VMs, data sources, and dependencies. After workloads have been defined conceptually, they can be used as the basis for grouping assets based on dependency mapping.

The Azure Migrate documentation provides guidance on how to group machines based on dependencies.

Configuration-management database

Some organizations have a well-maintained configuration-management database (CMDB) within their existing operations-management tooling. They could use the CMDB alternatively to provide the input data points discussed earlier.

Next steps

Review rationalization decisions based on asset alignment and workload definitions.

Review rationalization decisions

Review rationalization decisions

3/6/2021 • 4 minutes to read • Edit Online

During initial strategy and planning stages, we suggest you apply an incremental rationalization approach to the digital estate. But this approach embeds some assumptions into the resulting decisions. We advise the cloud strategy team and the cloud adoption teams to review those decisions in light of expanded-workload documentation. This review is also a good time to involve business stakeholders and the executive sponsor in future state decisions.

> **IMPORTANT**
> Further validation of the rationalization decisions will occur during the assessment phase of migration. This validation focuses on business review of the rationalization to align resources appropriately.

To validate rationalization decisions, use the following questions to facilitate a conversation with the business. The questions are grouped by the likely rationalization alignment.

Innovation indicators

If the joint review of the following questions yields an affirmative answer, a workload might be a better candidate for innovation. Such a workload wouldn't be migrated via a lift and shift or modernize model. Instead, the business logic or data structures would be re-created as a new or rearchitected application. This approach can be more labor-intensive and time-consuming. But for a workload that represents significant business returns, the investment is justified.

- Do the applications in this workload create market differentiation?
- Is there a proposed or approved investment aimed at improving the experiences associated with the applications in this workload?
- Does the data in this workload make new product or service offerings available?
- Is there a proposed or approved investment aimed at taking advantage of the data associated with this workload?
- Can the effect of the market differentiation or new offerings be quantified? If so, does that return justify the increased cost of innovation during cloud adoption?

The following two questions can help you include high-level technical scenarios in the rationalization review. Answering "yes" to either could identify ways of accounting for or reducing the cost associated with innovation.

- Will the data structures or business logic change during the course of cloud adoption?
- Is an existing deployment pipeline used to deploy this workload to production?

If the answer to either question is "yes," the team should consider including this workload as an innovation candidate. At a minimum, the team should flag this workload for architecture review to identify modernization opportunities.

Migration indicators

Migration is a faster and cheaper way of adopting the cloud. But it doesn't take advantage of opportunities to innovate. Before you invest in innovation, answer the following questions. They can help you determine whether a migration model is more applicable for a workload.

- Is the source code supporting this application stable? Do you expect it to remain stable and unchanged during the time frame of this release cycle?
- Does this workload support production business processes today? Will it do so throughout the course of this release cycle?
- Is it a priority that this cloud adoption effort improves the stability and performance of this workload?
- Is cost reduction associated with this workload an objective during this effort?
- Is reducing operational complexity for this workload a goal during this effort?
- Is innovation limited by the current architecture or IT operation processes?

If the answer to any of these questions is "yes," you should consider a migration model for this workload. This recommendation is true even if the workload is a candidate for innovation.

Challenges in operational complexity, costs, performance, or stability can hinder business returns. You can use the cloud to quickly produce improvements related to those challenges. Where it's applicable, we suggest you use the migration approach to first stabilize the workload. Then expand on innovation opportunities in the stable, agile cloud environment. This approach provides short-term returns and reduces the cost required to drive long-term change.

> **IMPORTANT**
> Migration models include incremental modernization. Using platform as a service (PaaS) architectures is a common aspect of migration activities. So too are minor configuration changes that use those platform services. The boundary for migration is defined as a material change to the business logic or supporting business structures. Such change is considered an innovation effort.

Update the project plan

The skills required for a migration effort are different from the skills required for an innovation effort. During implementation of a cloud adoption plan, we suggest that you assign migration and innovation efforts to different teams. Each team has its own iteration, release, and planning cadences. Assigning separate teams provides the process flexibility to maintain one cloud adoption plan while accounting for innovation and migration efforts.

When you manage the cloud adoption plan in Azure DevOps, that management is reflected by changing the parent work item (or epic) from cloud migration to cloud innovation. This subtle change helps ensure all participants in the cloud adoption plan can quickly track the required effort and changes to remediation efforts. This tracking also helps align proper assignments to the relevant cloud adoption team.

For large, complex adoption plans with multiple distinct projects, consider updating the iteration path. Changing the area path makes the workload visible only to the team assigned to that area path. This change can make work easier for the cloud adoption team by reducing the number of visible tasks. But it adds complexity for the project management processes.

Next steps

Establish iterations and release plans to begin planning work.

Establish iterations and release plans to begin planning work.

Establish iterations and release plans

3/6/2021 • 4 minutes to read • Edit Online

Agile and other iterative methodologies are built on the concepts of iterations and releases. This article outlines the assignment of iterations and releases during planning. Those assignments drive timeline visibility to make conversations easier among members of the cloud strategy team. The assignments also align technical tasks in a way that the cloud adoption team can manage during implementation.

Establish iterations

In an iterative approach to technical implementation, you plan technical efforts around recurring time blocks. Iterations tend to be one-week to six-week time blocks. Consensus suggests that two weeks is the average iteration duration for most cloud adoption teams. But the choice of iteration duration depends on the type of technical effort, the administrative overhead, and the team's preference.

To begin aligning efforts to a timeline, we suggest that you define a set of iterations that last 6 to 12 months.

Understand velocity

Aligning efforts to iterations and releases requires an understanding of velocity. Velocity is the amount of work that can be completed in any given iteration. During early planning, velocity is an estimate. After several iterations, velocity becomes a highly valuable indicator of the commitments that the team can make confidently.

You can measure velocity in abstract terms like story points. You can also measure it in more tangible terms like hours. For most iterative frameworks, we recommend using abstract measurements to avoid challenges in precision and perception. Examples in this article represent velocity in hours per sprint. This representation makes the topic more universally understood.

Example: A five-person cloud adoption team has committed to two-week sprints. Given current obligations like meetings and support of other processes, each team member can consistently contribute 20 hours per week to the adoption effort. For this team, the initial velocity estimate is 100 hours per sprint.

Iteration planning

Initially, you plan iterations by evaluating the technical tasks based on the prioritized backlog. Cloud adoption teams estimate the effort required to complete various tasks. Those tasks are then assigned to the first available iteration.

During iteration planning, the cloud adoption teams validate and refine estimates. They do so until they have aligned all available velocity to specific tasks. This process continues for each prioritized workload until all efforts align to a forecasted iteration.

In this process, the team validates the tasks assigned to the next sprint. The team updates its estimates based on the team's conversation about each task. The team then adds each estimated task to the next sprint until the available velocity is met. Finally, the team estimates additional tasks and adds them to the next iteration. The team performs these steps until the velocity of that iteration is also exhausted.

The preceding process continues until all tasks are assigned to an iteration.

Example: Let's build on the previous example. Assume each workload migration requires 40 tasks. Also assume you estimate each task to take an average of one hour. The combined estimation is approximately 40 hours per workload migration. If these estimates remain consistent for all 10 of the prioritized workloads, those workloads

will take 400 hours.

The velocity defined in the previous example suggests that the migration of the first 10 workloads will take four iterations, which is two months of calendar time. The first iteration will consist of 100 tasks that result in the migration of two workloads. In the next iteration, a similar collection of 100 tasks will result in the migration of three workloads.

> **WARNING**
> The preceding numbers of tasks and estimates are strictly used as an example. Technical tasks are seldom that consistent. You shouldn't see this example as a reflection of the amount of time required to migrate a workload.

Release planning

Within cloud adoption, a release is defined as a collection of deliverables that produce enough business value to justify the risk of disruption to business processes.

Releasing any workload-related changes into a production environment creates some changes to business processes. Ideally, these changes are seamless, and the business sees the value of the changes with no significant disruptions to service. But the risk of business disruption is present with any change and shouldn't be taken lightly.

To ensure a change is justified by its potential return, the cloud strategy team should participate in release planning. Once tasks are aligned to sprints, the team can determine a rough timeline of when each workload will be ready for production release. The cloud strategy team would review the timing of each release. The team would then identify the inflection point between risk and business value.

Example: Continuing the previous example, the cloud strategy team has reviewed the iteration plan. The review identified two release points. During the second iteration, a total of five workloads will be ready for migration. Those five workloads will provide significant business value and will trigger the first release. The next release will come two iterations later, when the next five workloads are ready for release.

Assign iteration paths and tags

For customers who manage cloud adoption plans in Azure DevOps, the previous processes are reflected by assigning an iteration path to each task and user story. We also recommend tagging each workload with a specific release. That tagging and assignment feed the automatic population of timeline reports.

Next steps

Estimate timelines to properly communicate expectations.

Estimate timelines

Timelines in a cloud adoption plan

3/6/2021 • 2 minutes to read • Edit Online

In the previous article in this series, workloads and tasks were assigned to releases and iterations. Those assignments feed the timeline estimates in this article.

Work breakdown structures are commonly used in sequential project-management tools. They represent how dependent tasks will be completed over time. Such structures work well when tasks are sequential in nature. The interdependencies in tasks found in cloud adoption make such structures difficult to manage. To fill this gap, you can estimate timelines based on iteration-path assignments by hiding complexity.

Estimate timelines

To develop a timeline, start with releases. Those release objectives create a target date for any business impact. Iterations aid in aligning those releases with specific time durations.

If more granular milestones are required in the timeline, use iteration assignment to indicate milestones. To do this assignment, assume that the last instance of a workload-related task can serve as the final milestone. Teams also commonly tag the final task as a milestone.

For any level of granularity, use the last day of the iteration as the date for each milestone. This ties completion of workload adoption to a specific date. You can track the date in a spreadsheet or a sequential project-management tool like Microsoft Project.

Delivery plans in Azure DevOps

If you're using Azure DevOps to manage your cloud adoption plan, consider using the Microsoft Delivery Plans extension. This extension can quickly create a visual representation of the timeline that is based on iteration and release assignments.

Assess on-premises workloads for migration to Azure

5/18/2021 • 21 minutes to read • Edit Online

This article shows how the fictional company Contoso assesses an on-premises app for migration to Azure. In the example scenario, Contoso's on-premises SmartHotel360 application currently runs on VMware. Contoso assesses the application VMs using the Azure Migrate service, and the SQL Server application database using Data Migration Assistant.

Overview

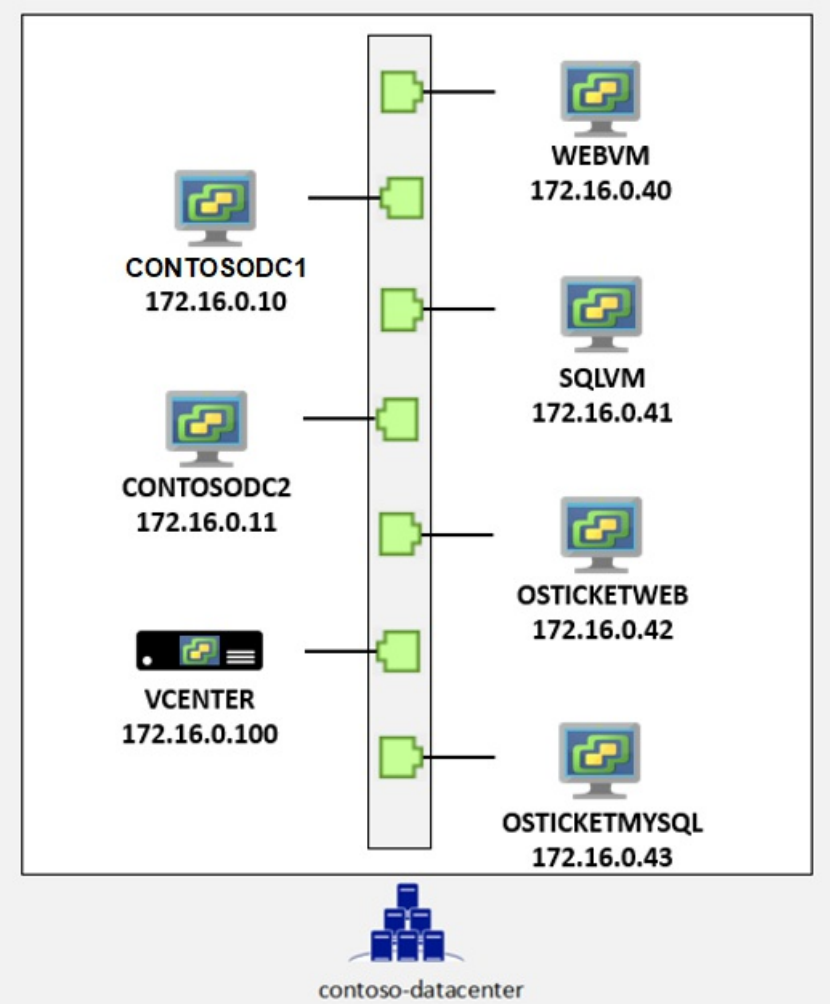
As Contoso considers migrating to Azure, the company needs a technical and financial assessment to determine whether its on-premises workloads are good candidates for cloud migration. In particular, the Contoso team wants to assess machine and database compatibility for migration. It wants to estimate capacity and costs for running Contoso's resources in Azure.

To get started and to better understand the technologies involved, Contoso assesses two of its on-premises apps, summarized in the following table. The company assesses for migration scenarios that rehost and refactor apps for migration. Learn more about rehosting and refactoring in the migration examples overview.

APP NAME	PLATFORM	APP TIERS	DETAILS
SmartHotel360 (manages Contoso travel requirements)	Runs on Windows with a SQL Server database	Two-tiered app. The front-end ASP.NET website runs on one VM (`WEBVM`) and the SQL Server runs on another VM (`SQLVM`).	VMs run on a VMware ESXi host managed by vCenter Server. You can download the sample app from GitHub.
osTicket (Contoso service desk app)	Runs on a LAMP stack.	Two-tiered app. A front-end PHP website runs on one VM (`OSTICKETWEB`) and the MySQL database runs on another VM (`OSTICKETMYSQL`).	The app is used by customer service apps to track issues for internal employees and external customers. You can download the sample from GitHub.

Current architecture

This diagram shows the current Contoso on-premises infrastructure:



- Contoso has one main datacenter. The datacenter is located in the city of New York in the Eastern United States.
- Contoso has three additional local branches across the United States.
- The main datacenter is connected to the internet with a fiber optic Metro Ethernet connection (500 MBps).
- Each branch is connected locally to the internet by using business-class connections with IPsec VPN tunnels back to the main datacenter. The setup allows Contoso's entire network to be permanently connected and optimizes internet connectivity.
- The main datacenter is fully virtualized with VMware. Contoso has two ESXi 6.5 virtualization hosts that are managed by vCenter Server 6.5.
- Contoso uses Active Directory for identity management. Contoso uses DNS servers on the internal network.
- The domain controllers in the datacenter run on VMware VMs. The domain controllers at local branches run on physical servers.

Business drivers

Contoso's IT leadership team has worked closely with the company's business partners to understand what the business wants to achieve with this migration:

- **Address business growth.** Contoso is growing. As a result, pressure has increased on the company's on-premises systems and infrastructure.
- **Increase efficiency.** Contoso needs to remove unnecessary procedures and streamline processes for its developers and users. The business needs IT to be fast and to not waste time or money, so the company can deliver faster on customer requirements.
- **Increase agility.** Contoso IT needs to be more responsive to the needs of the business. It must be able to react faster than the changes that occur in the marketplace for the company to be successful in a global economy. IT at Contoso must not get in the way or become a business blocker.

- **Scale.** As the company's business grows successfully, Contoso IT must provide systems that can grow at the same pace.

Assessment goals

The Contoso cloud team has identified goals for its migration assessments:

- After migration, apps in Azure should have the same performance capabilities that apps have today in Contoso's on-premises VMware environment. Moving to the cloud doesn't mean that app performance is less critical.
- Contoso needs to understand the compatibility of its applications and databases with Azure requirements. Contoso also needs to understand its hosting options in Azure.
- Contoso's database administration should be minimized after apps move to the cloud.
- Contoso wants to understand not only its migration options, but also the costs associated with the infrastructure after it moves to the cloud.

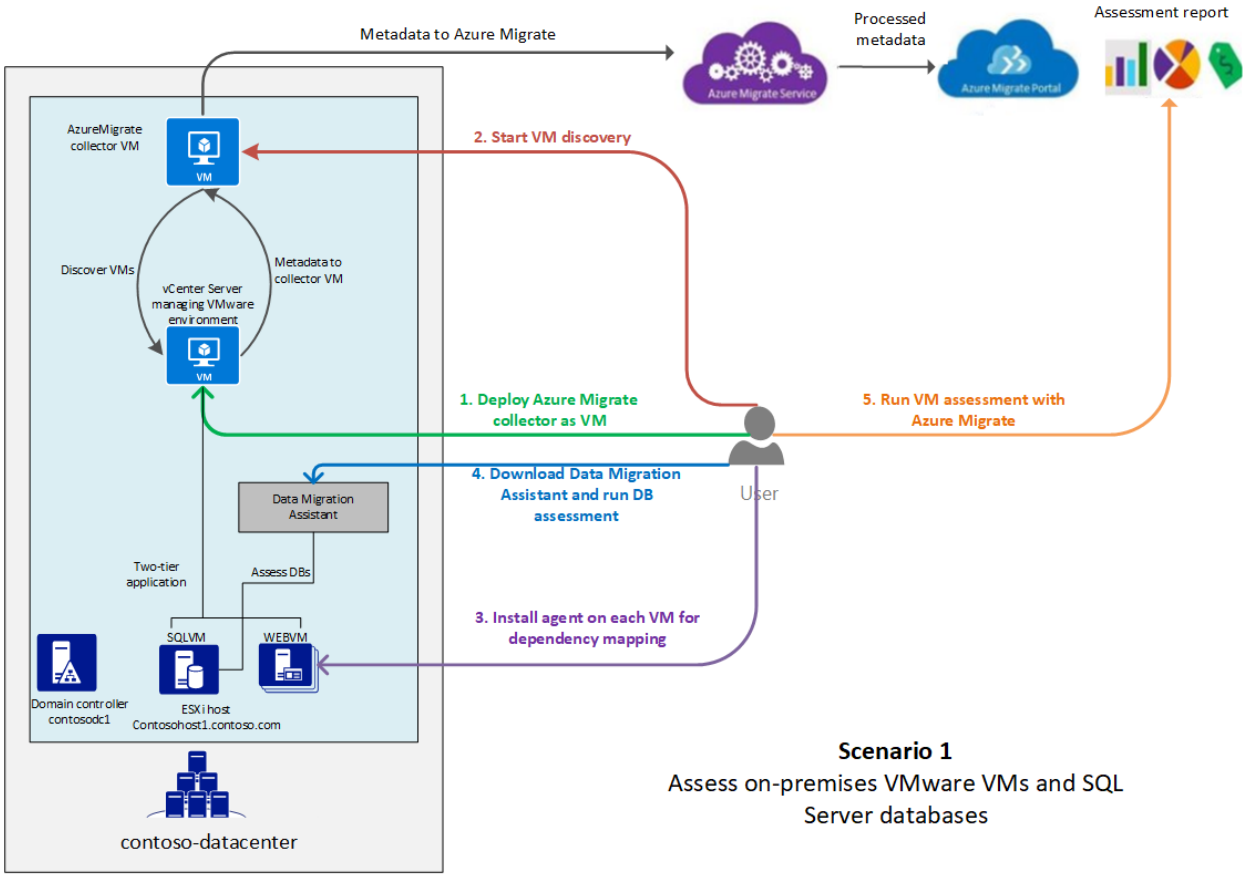
Assessment tools

Contoso uses Microsoft tools for its migration assessment. The tools align with the company's goals and should provide Contoso with all the information it needs.

TECHNOLOGY	DESCRIPTION	COST
Data Migration Assistant	Contoso uses Data Migration Assistant to assess and detect compatibility issues that might affect its database functionality in Azure. Data Migration Assistant assesses feature parity between SQL sources and targets. It recommends performance and reliability improvements.	Data Migration Assistant is a free downloadable tool.
Azure Migrate	Contoso uses the Azure Migrate service to assess its VMware VMs. Azure Migrate assesses the migration suitability of the machines. It provides sizing and cost estimates for running in Azure.	Azure Migrate is available at no additional charge. However, you may incur charges depending on the tools (first-party or ISV) you decide to use for assessment and migration. Learn more about Azure Migrate pricing.
Service Map	Azure Migrate uses Service Map to show dependencies between machines that the company wants to migrate.	Service Map is part of Azure Monitor logs. Currently, Contoso can use Service Map for 180 days without incurring charges.

In this scenario, Contoso downloads and runs Data Migration Assistant to assess the on-premises SQL Server database for its travel app. Contoso uses Azure Migrate with dependency mapping to assess the app VMs before migration to Azure.

Assessment architecture



- Contoso is a fictional name that represents a typical enterprise organization.
- Contoso has an on-premises datacenter (`contoso-datacenter`) and on-premises domain controllers (`CONTOSODC1` , `CONTOSODC2`).
- VMware VMs are located on VMware ESXi hosts running version 6.5 (`contosohost1` , `contosohost2`).
- The VMware environment is managed by vCenter Server 6.5 (`vcenter.contoso.com` , running on a VM).
- The SmartHotel360 travel app has these characteristics:
  - The app is tiered across two VMware VMs (`WEBVM` and `SQLVM`).
  - The VMs are located on VMware ESXi host `contosohost1.contoso.com` .
  - The VMs are running Windows Server 2008 R2 Datacenter with SP1.
- The VMware environment is managed by vCenter Server (`vcenter.contoso.com`) running on a VM.
- The osTicket service desk app:
  - The app is tiered across two VMs (`OSTICKETWEB` and `OSTICKETMYSQL`).
  - The VMs are running Ubuntu Linux Server 16.04-LTS.
  - `OSTICKETWEB` is running Apache 2 and PHP 7.0.
  - `OSTICKETMYSQL` is running MySQL 5.7.22.

Prerequisites

Contoso and other users must meet the following prerequisites for the assessment:

- Owner or Contributor permissions for the Azure subscription, or for a resource group in the Azure subscription.
- An on-premises vCenter Server instance running version 6.5, 6.0, or 5.5.
- A read-only account in vCenter Server, or permissions to create one.
- Permissions to create a VM on the vCenter Server instance by using an .ova template.
- At least one ESXi host running version 5.5 or later.

- At least two on-premises VMware VMs, one running a SQL Server database.
- Permissions to install Azure Migrate agents on each VM.
- The VMs should have direct internet connectivity.
  - You can restrict internet access to the required URLs.
  - If your VMs don't have internet connectivity, the Azure Log Analytics Gateway must be installed on them, and agent traffic directed through it.
- The fully qualified domain name (FQDN) of the VM running the SQL Server instance, for database assessment.
- Windows Firewall running on the SQL Server VM should allow external connections on TCP port 1433 (default). This setup allows Data Migration Assistant to connect.

Assessment overview

Here's how Contoso performs its assessment:

- **Step 1: Download and install Data Migration Assistant.** Contoso prepares Data Migration Assistant for assessment of the on-premises SQL Server database.
- **Step 2: Assess the database by using Data Migration Assistant.** Contoso runs and analyzes the database assessment.
- **Step 3: Prepare for VM assessment by using Azure Migrate.** Contoso sets up on-premises accounts and adjusts VMware settings.
- **Step 4: Discover on-premises VMs by using Azure Migrate.** Contoso creates an Azure Migrate collector VM. Then, Contoso runs the collector to discover VMs for assessment.
- **Step 5: Prepare for dependency analysis by using Azure Migrate.** Contoso installs Azure Migrate agents on the VMs, so the company can see dependency mapping between VMs.
- **Step 6: Assess the VMs by using Azure Migrate.** Contoso checks dependencies, groups the VMs, and runs the assessment. When the assessment is ready, Contoso analyzes the assessment in preparation for migration.

> **NOTE**
> Assessments shouldn't just be limited to using tooling to discover information about your environment. You should also schedule time to speak to business owners, end users, and other members of the IT department to fully understand of what is happening in the environment and understand factors that tooling cannot tell you.

Step 1: Download and install Data Migration Assistant

1. Contoso downloads Data Migration Assistant from the Microsoft Download Center.
   - Data Migration Assistant can be installed on any machine that can connect to the SQL Server instance. Contoso doesn't need to run it on the SQL Server machine.
   - Data Migration Assistant shouldn't be run on the SQL Server host machine.
2. Contoso runs the downloaded setup file `DownloadMigrationAssistant.msi` to begin the installation.
3. On the **Finish** page, Contoso selects **Launch Microsoft Data Migration Assistant** before finishing the wizard.

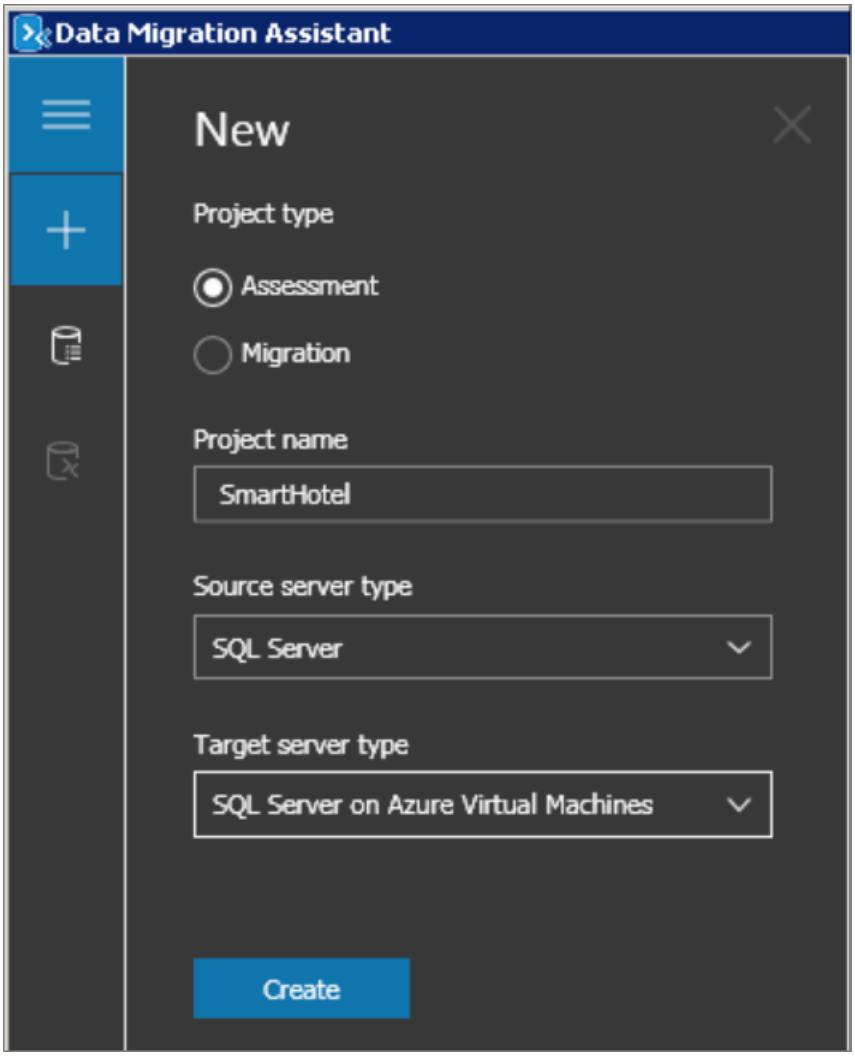
Step 2: Run and analyze the database assessment for SmartHotel360

Now, Contoso can run an assessment to analyze its on-premises SQL Server database for the SmartHotel360 application.

1. In Data Migration Assistant, Contoso selects **New** > **Assessment**, and then gives the assessment a

project name.

2. For **Source server type**, Contoso selects **SQL Server** and for **Target Server type**, Contoso selects **SQL Server on Azure Virtual Machines**

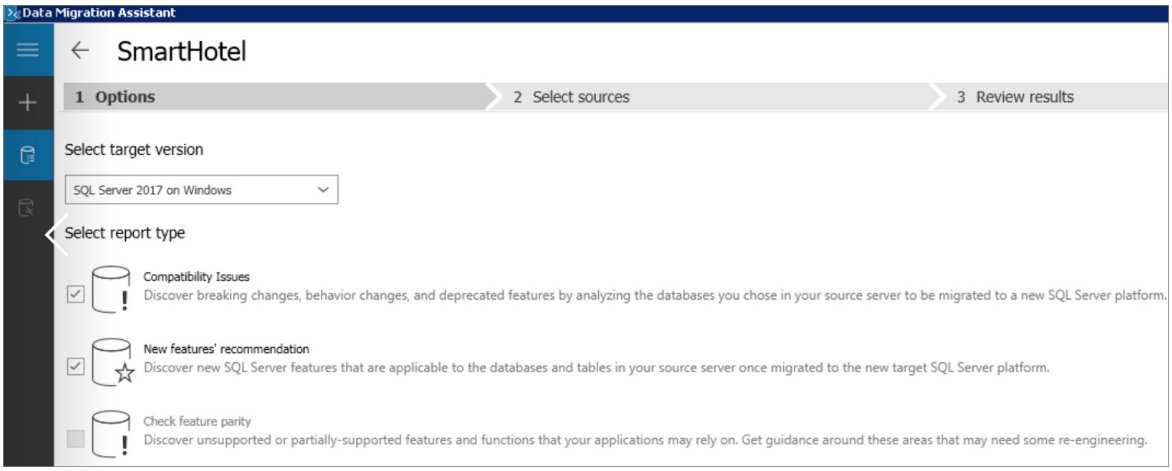


> **NOTE**
> Currently, Data Migration Assistant doesn't support assessment for migrating to Azure SQL Managed Instance. As a workaround, Contoso uses SQL Server on an Azure VM as the supposed target for the assessment.

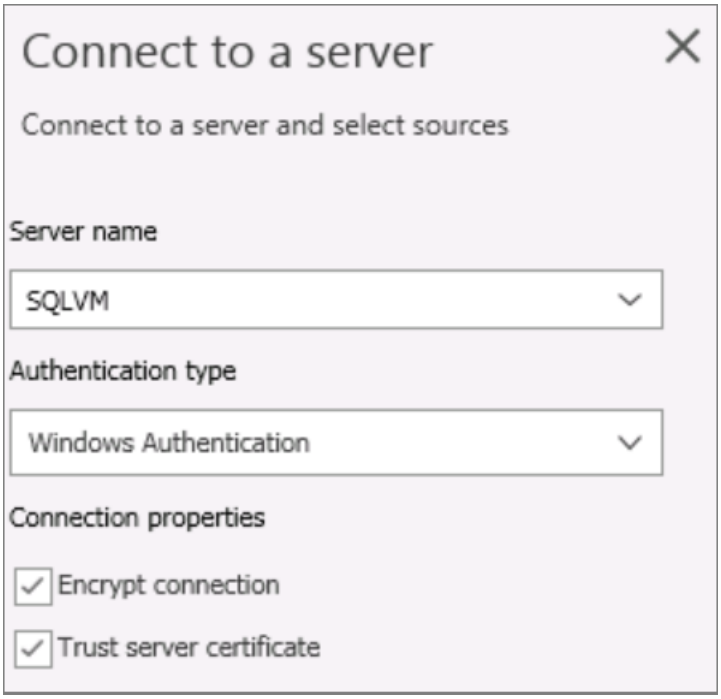
3. In **Select Target Version**, Contoso selects SQL Server 2017 as the target version. Contoso needs to select this version because it's the version that's used by the SQL Managed Instance.

4. Contoso selects reports to help it discover information about compatibility and new features:

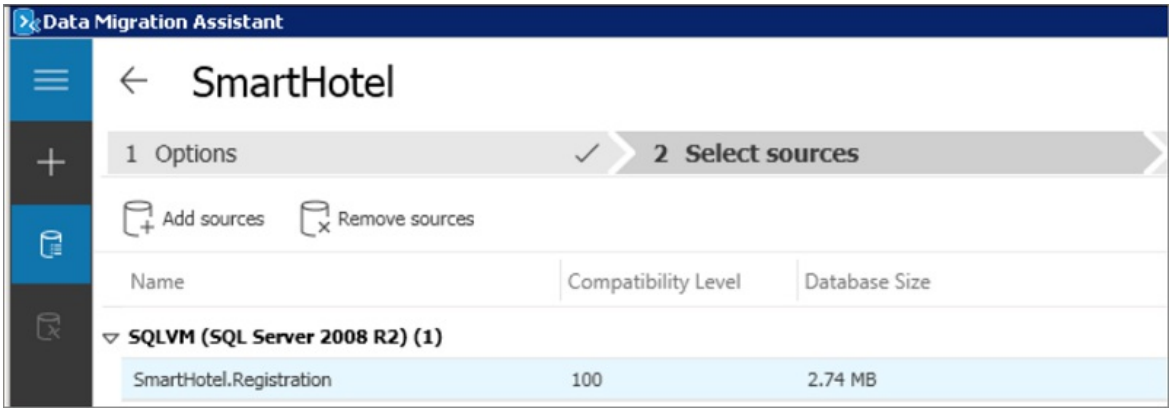
   - **Compatibility issues** note changes that might break migration or that require a minor adjustment before migration. This report keeps Contoso informed about any features currently in use that are deprecated. Issues are organized by compatibility level.
   - **New feature recommendation** notes new features in the target SQL Server platform that can be used for the database after migration. New feature recommendations are organized under the headings **Performance**, **Security**, and **Storage.**



5. In **Connect to a server**, Contoso enters the name of the VM that's running the database and credentials to access it. Contoso selects **Trust server certificate** to make sure the VM can access SQL Server. Then, Contoso selects **Connect**.



6. In **Add source**, Contoso adds the database it wants to assess, then selects **Next** to start the assessment.
7. The assessment is created.



8. In **Review results**, Contoso views the assessment results.

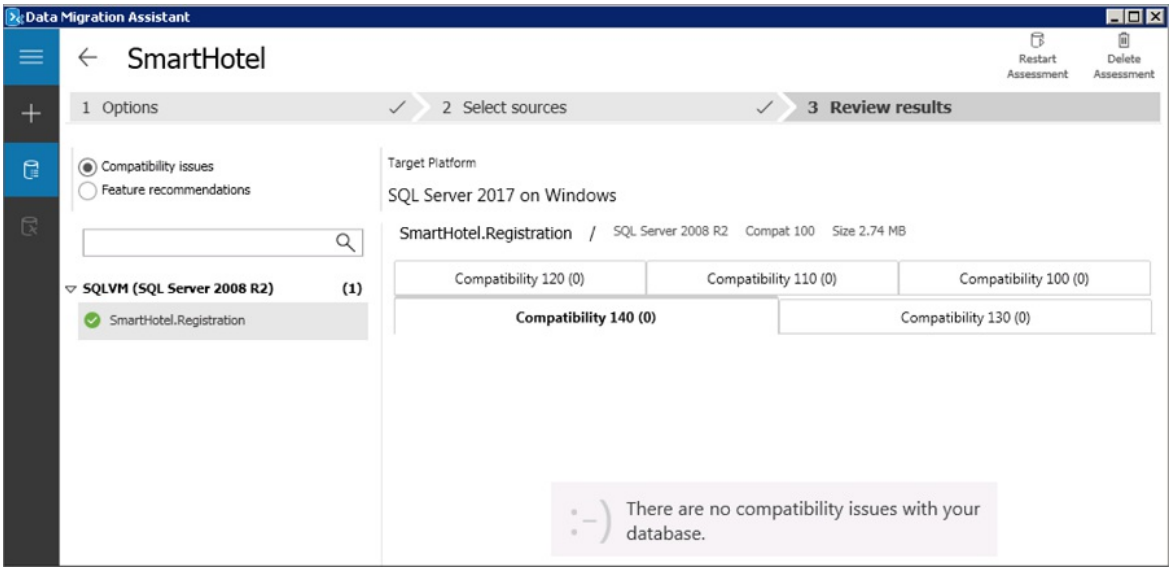
Analyze the database assessment

Results are displayed as soon as they're available. If Contoso fixes issues, it must select **Restart assessment** to

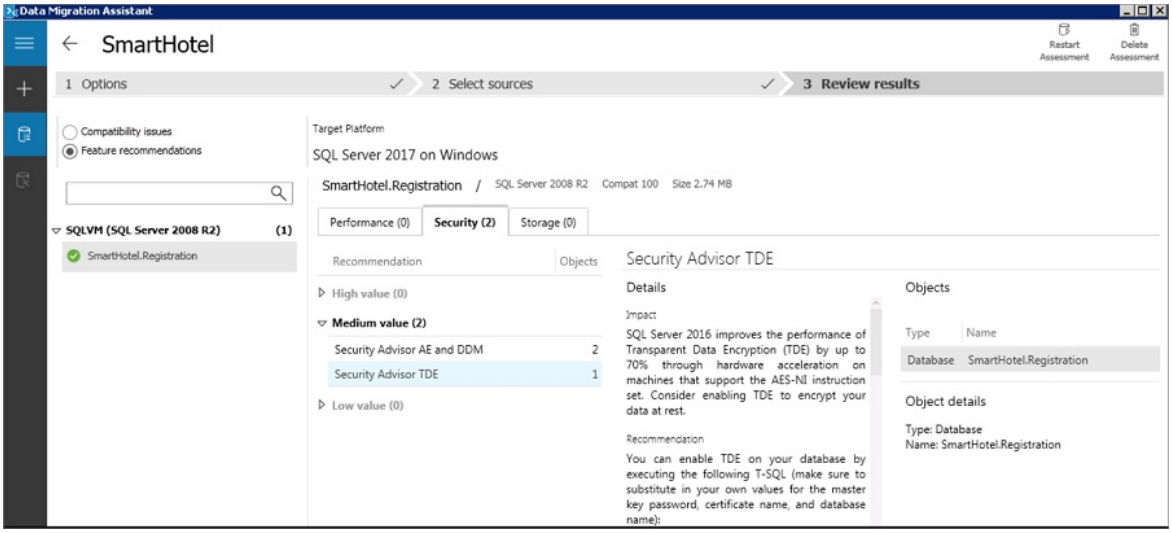
rerun the assessment.

1. In the **Compatibility issues** report, Contoso checks for any issues at each compatibility level. Compatibility levels map to SQL Server versions as follows:

   - 100: SQL Server 2008/Azure SQL Database
   - 110: SQL Server 2012/Azure SQL Database
   - 120: SQL Server 2014/Azure SQL Database
   - 130: SQL Server 2016/Azure SQL Database
   - 140: SQL Server 2017/Azure SQL Database



2. In the **Feature recommendations** report, Contoso views performance, security, and storage features that the assessment recommends after migration. A variety of features are recommended, including In-Memory OLTP, columnstore indexes, Stretch Database, Always Encrypted, dynamic data masking, and transparent data encryption.



> **NOTE**
>
> Contoso should enable transparent data encryption for all SQL Server databases. This is even more critical when a database is in the cloud than when it's hosted on-premises. Transparent data encryption should be enabled only after migration. If transparent data encryption is already enabled, Contoso must move the certificate or asymmetric key to the `master` database of the target server. Learn how to move a transparent data encryption-protected database to another SQL Server instance.

3. Contoso can export the assessment in JSON or CSV format.

NOTE

For large-scale assessments:

- Run multiple assessments concurrently and view the state of the assessments on the **All assessments** page.
- Consolidate assessments into a SQL Server database.
- Consolidate assessments into a Power BI report.

Step 3: Prepare for VM assessment by using Azure Migrate

Contoso needs to create a VMware account that Azure Migrate can use to automatically discover VMs for assessment, verify rights to create a VM, note the ports that need to be opened, and set the statistics settings level.

Set up a VMware account

VM discovery requires a read-only account in vCenter Server that has the following properties:

- **User type:** At least a read-only user.
- **Permissions:** For the datacenter object, select the **Propagate to Child Objects** checkbox. For **Role**, select **Read-only**.
- **Details:** The user is assigned at the datacenter level, with access to all objects in the datacenter.
- To restrict access, assign the **No access** role with the **Propagate to child** object to the child objects (vSphere hosts, data stores, VMs, and networks).

Verify permissions to create a VM

Contoso verifies that it has permissions to create a VM by importing a file in .ova format. Learn how to create and assign a role with privileges.

Verify ports

The Contoso assessment uses dependency mapping. Dependency mapping requires an agent to be installed on VMs that will be assessed. The agent must be able to connect to Azure from TCP port 443 on each VM. Learn about connection requirements.

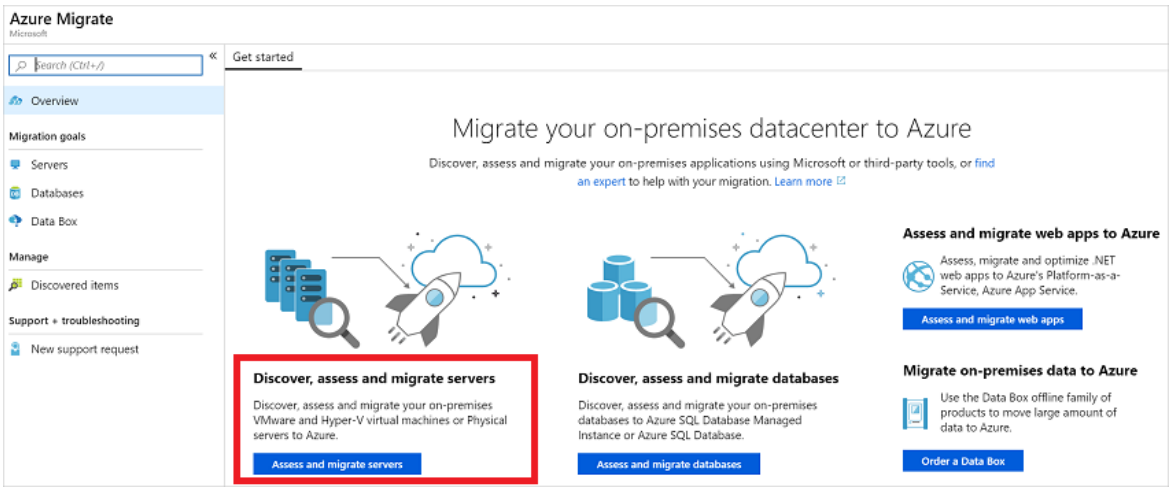
Step 4: Discover VMs

To discover VMs, Contoso creates an Azure Migrate project. Contoso downloads and sets up the collector VM. Then, Contoso runs the collector to discover its on-premises VMs.

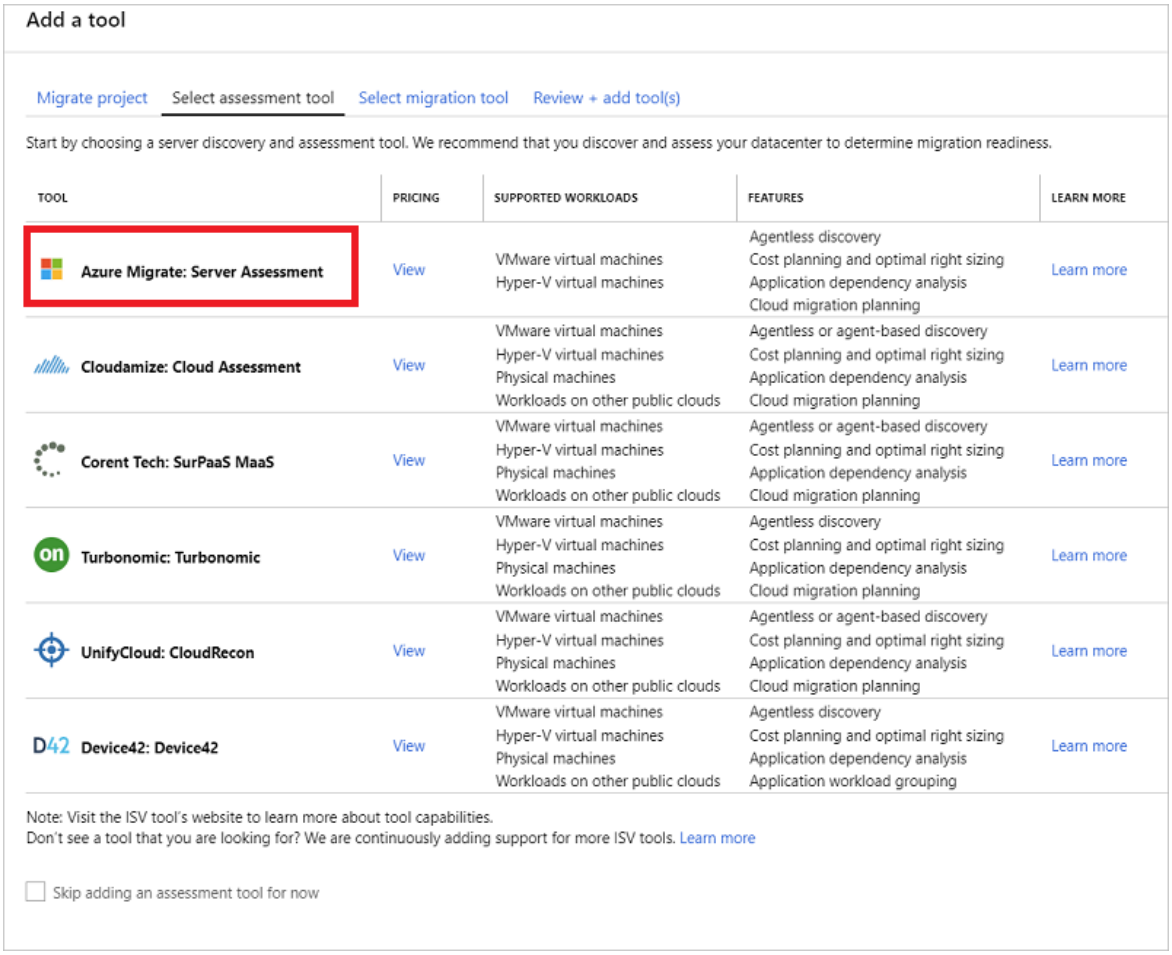
Create a project

Set up a new Azure Migrate project as follows.

1. In the Azure portal > **All services**, search for **Azure Migrate**.
2. Under **Services**, select **Azure Migrate**.
3. In **Overview**, under **Discover, assess and migrate servers**, select **Assess and migrate servers**.



4. In **Getting started**, select **Add tools**.
5. In **Migrate project**, select your Azure subscription, and create a resource group if you don't have one.
6. In **Project Details**, specify the project name, and the geography in which you want to create the project. United States, Asia, Europe, Australia, United Kingdom, Canada, India, and Japan are supported.
   - The project geography is used only to store the metadata gathered from on-premises VMs.
   - You can select any target region when you run a migration.
7. Select **Next**.
8. In **Select assessment tool**, select **Azure Migrate: Server Assessment** > **Next**.

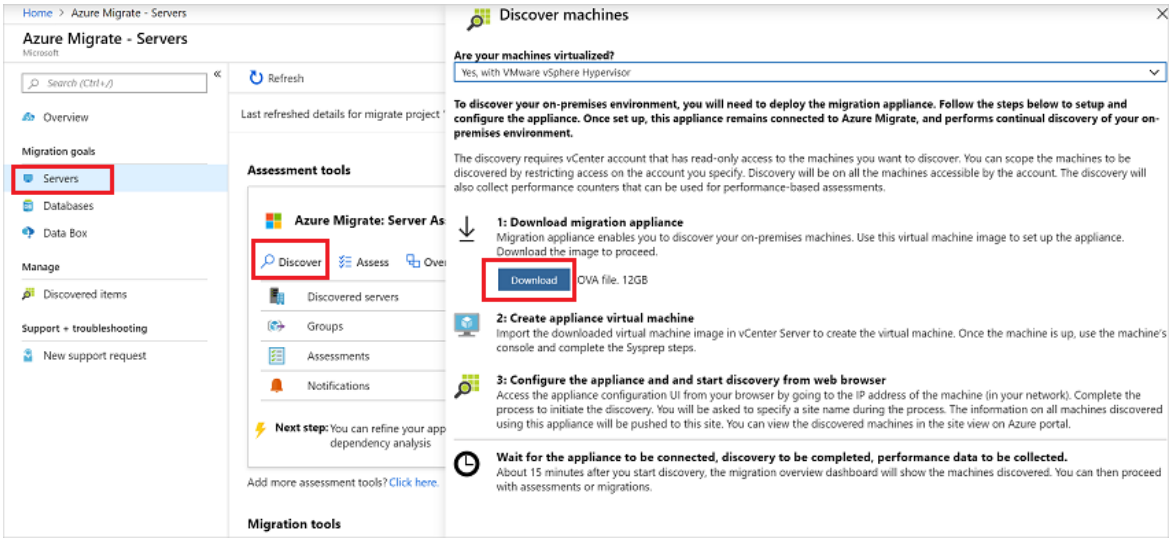


9. In **Select migration tool**, select **Skip adding a migration tool for now** > **Next**.
10. In **Review + add tools**, review the settings, then select **Add tools**.

11. Wait a few minutes for the Azure Migrate project to deploy. You'll be taken to the project page. If you don't see the project, you can access it from **Servers** in the Azure Migrate dashboard.

Download the collector appliance

1. In **Migration Goals** > **Servers** > **Azure Migrate: Server Assessment**, select **Discover**.
2. In **Discover machines** > **Are your machines virtualized?**, select **Yes, with VMware vSphere hypervisor**.
3. Select **Download** to download the .OVA template file.



Verify the collector appliance

Before deploying the VM, Contoso checks that the OVA file is secure:

1. On the machine on which the file was downloaded, Contoso opens an administrator Command Prompt window.
2. Contoso runs the following command to generate the hash for the OVA file:

   ```
   C:\> CertUtil -HashFile <file_location> [Hashing Algorithm]
   ```

   Example:

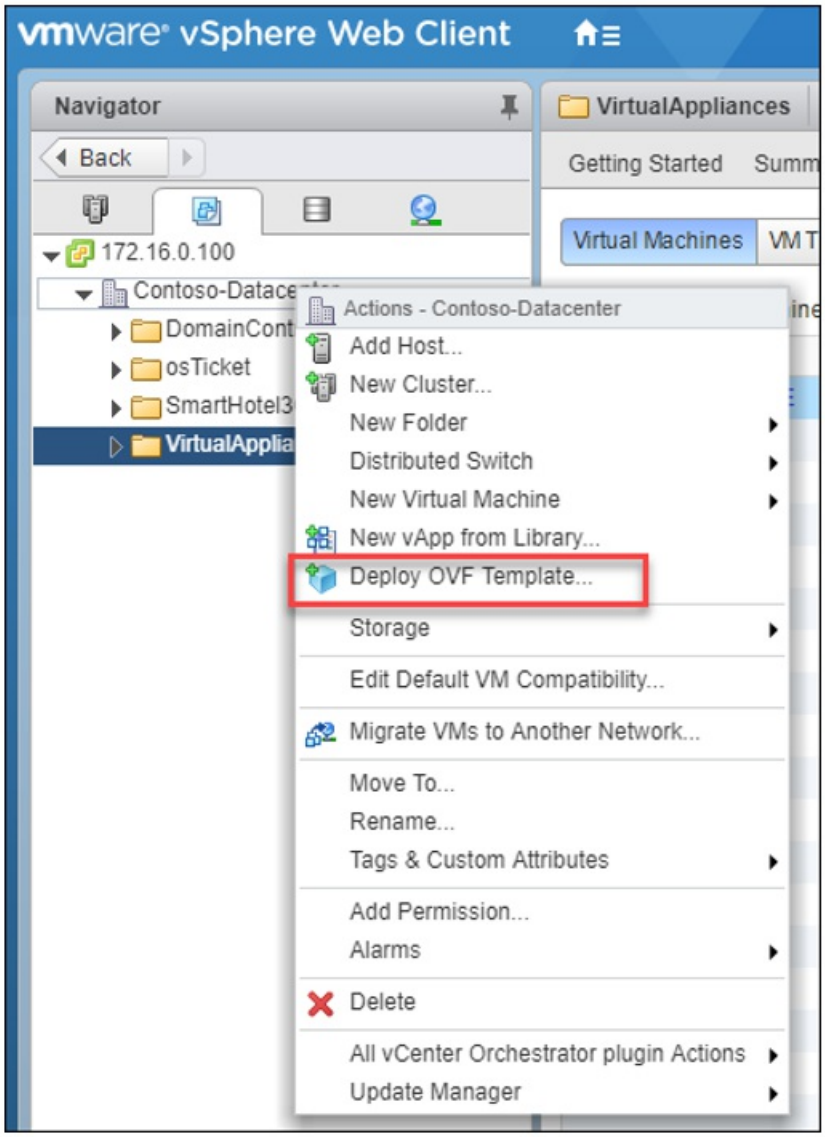
   ```
   C:\> CertUtil -HashFile C:\AzureMigrate\AzureMigrate.ova SHA256
   ```

3. The generated hash should match the hash values listed in the Verify security section of the Assess VMware VMs for migration tutorial.

Create the collector appliance

Now, Contoso can import the downloaded file to the vCenter Server instance and provision the collector appliance VM:

1. In the vSphere Client console, Contoso selects **File** > **Deploy OVF template**.



2. In the Deploy OVF Template Wizard, Contoso selects **Source**, and then specifies the location of the OVA file.
3. In **Name and Location**, Contoso specifies a display name for the collector VM. Then, it selects the inventory location in which to host the VM. Contoso also specifies the host or cluster on which to run the collector appliance.
4. In **Storage**, Contoso specifies the storage location. In **Disk Format**, Contoso selects how it wants to provision the storage.
5. In **Network Mapping**, Contoso specifies the network in which to connect the collector VM. The network needs internet connectivity to send metadata to Azure.
6. Contoso reviews the settings, then selects **Power on after deployment** > **Finish**. A message that confirms successful completion appears when the appliance is created.

Run the collector to discover VMs

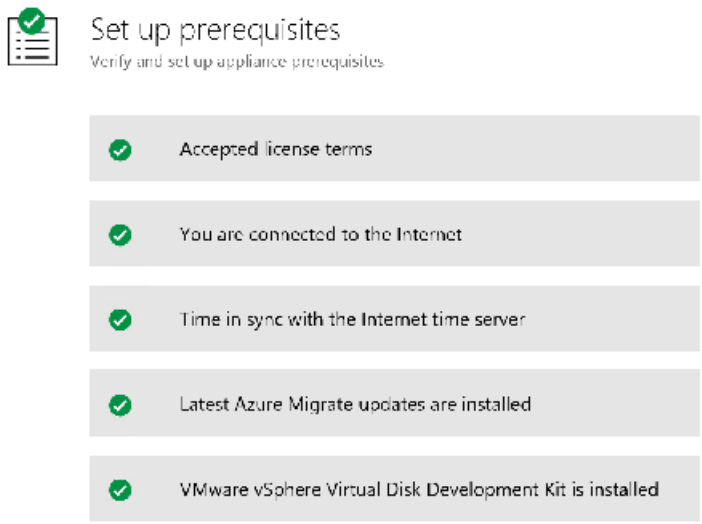
Now, Contoso runs the collector to discover VMs. Currently, the collector currently supports only **English (United States)** as the operating system language and collector interface language.

1. In the vSphere Client console, Contoso selects **Open Console**. Contoso specifies the accepts the licensing terms, and password preferences for the collector VM.
2. On the desktop, Contoso selects the **Microsoft Azure Appliance Configuration Manager** shortcut.



3. In the Azure Migrate Collector, Contoso selects **Set up prerequisites**. Contoso accepts the license terms and reads the third-party information.
4. The collector checks that the VM has internet access, that the time is synced, and that the collector service is running. (The collector service is installed by default on the VM.) Contoso also installs the VMware vSphere Virtual Disk Development Kit.

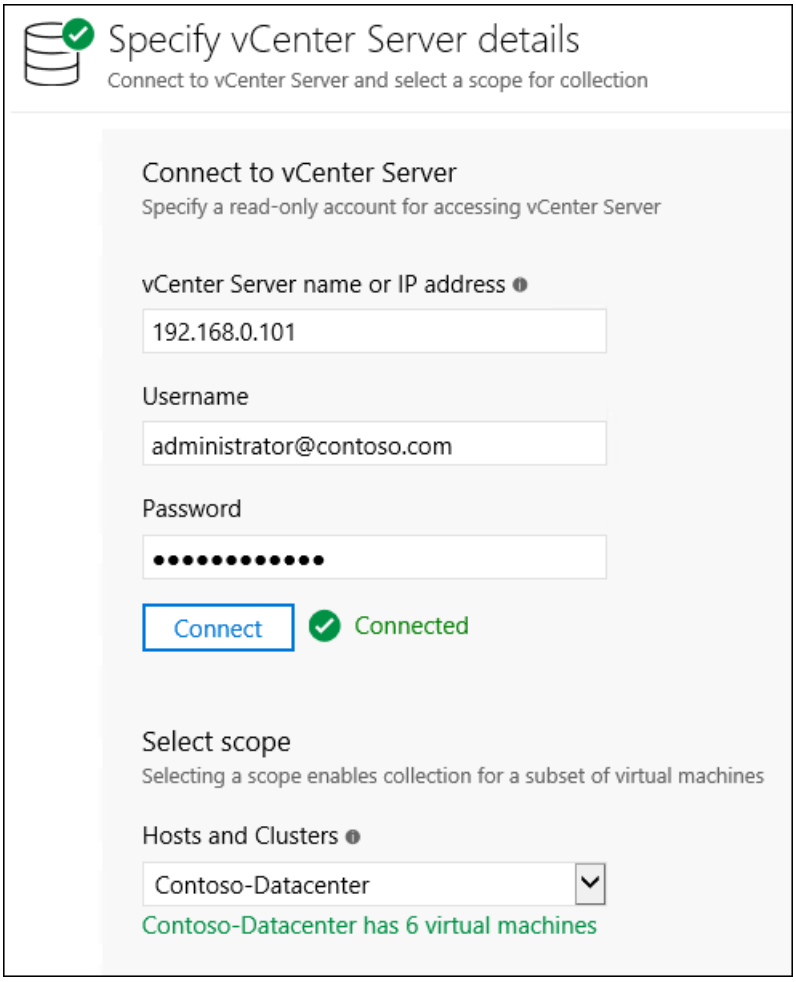
> **NOTE**
> It's assumed that the VM has direct access to the internet without using a proxy.



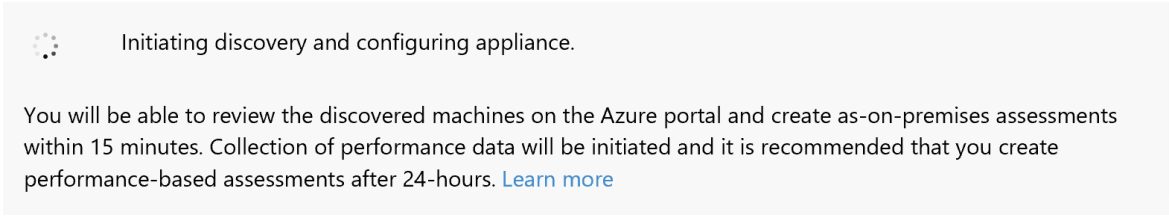
5. Sign in to your Azure account and select the subscription and Migrate project you created earlier. Also enter a name for the **appliance** so you can identify it in the Azure portal.
6. In **Specify vCenter Server details**, Contoso enters the name (FQDN) or IP address of the vCenter

Server instance and the read-only credentials used for discovery.

7. Contoso selects a scope for VM discovery. The collector can discover only VMs that are within the specified scope. The scope can be set to a specific folder, datacenter, or cluster.



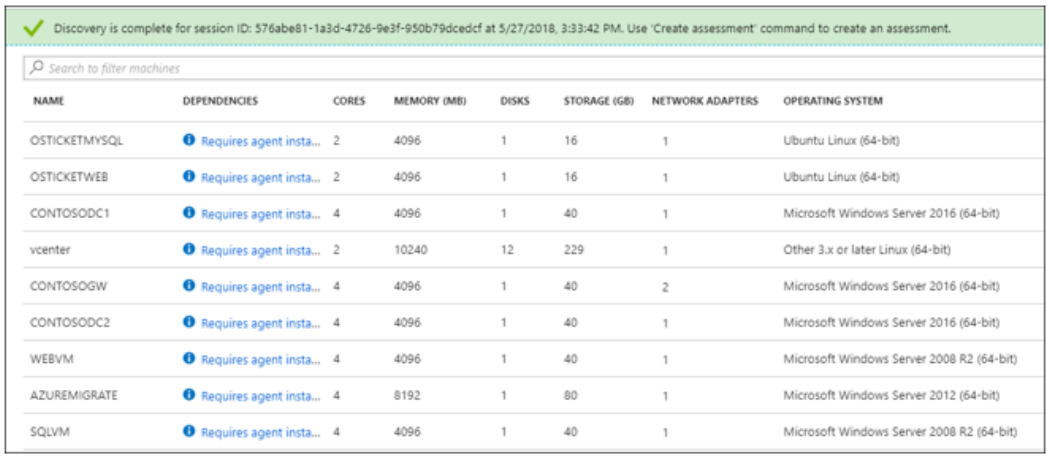
8. The collector will now start to discovery and collect information about the Contoso environment.



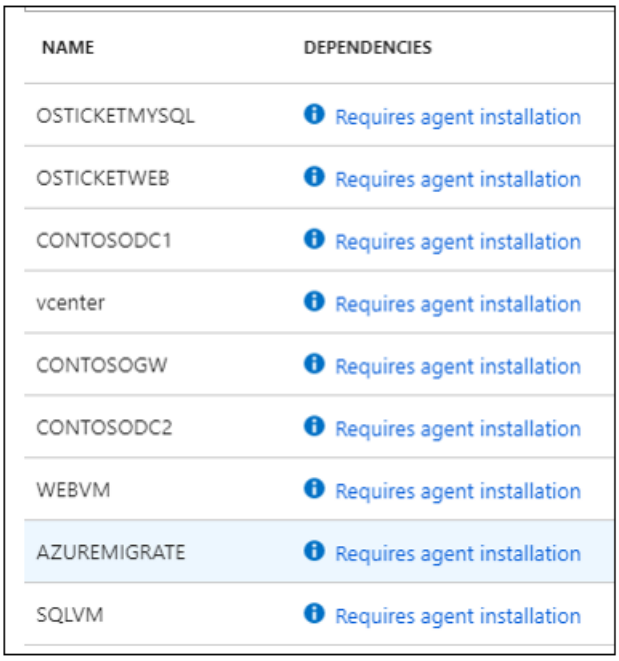
Verify VMs in the portal

When collection is finished, Contoso checks that the VMs appear in the portal:

1. In the Azure Migrate project, Contoso selects **Servers** servers**. Contoso checks that the VMs that it wants to discover are shown.



2. Currently, the machines don't have the Azure Migrate agents installed. Contoso must install the agents to view dependencies.

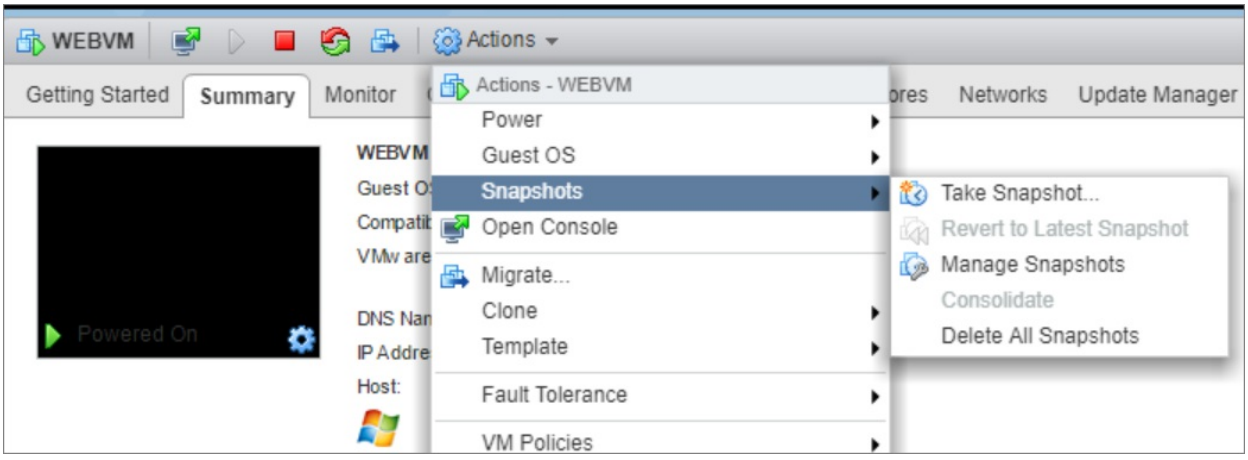


Step 5: Prepare for dependency analysis

To view dependencies between VMs that it wants to assess, Contoso downloads and installs agents on the app VMs. Contoso installs agents on all VMs for its apps, both for Windows and Linux.

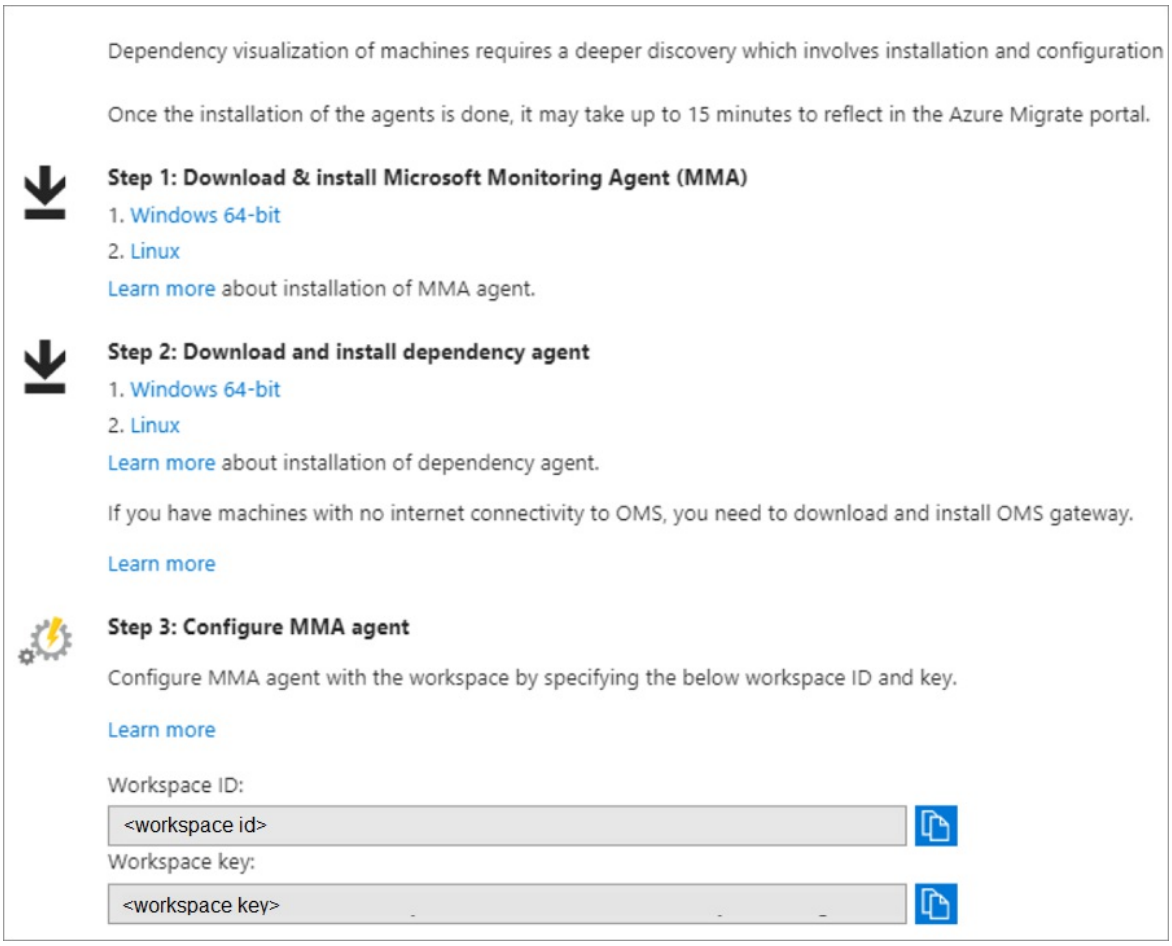
Take a snapshot

To keep a copy of the VMs before modifying them, Contoso takes a snapshot before the agents are installed.



Download and install the VM agents

1. In **Machines**, Contoso selects the machine. In the **Dependencies** column, Contoso selects **Requires installation**.
2. In the **Discover machines** pane, Contoso:
   - Downloads the Microsoft Monitoring Agent and the Microsoft Dependency Agent for each Windows VM.
   - Downloads the Microsoft Monitoring Agent and Microsoft Dependency Agent for each Linux VM.
3. Contoso copies the workspace ID and key. Contoso needs the workspace ID and key when it installs the Microsoft Monitoring Agent.

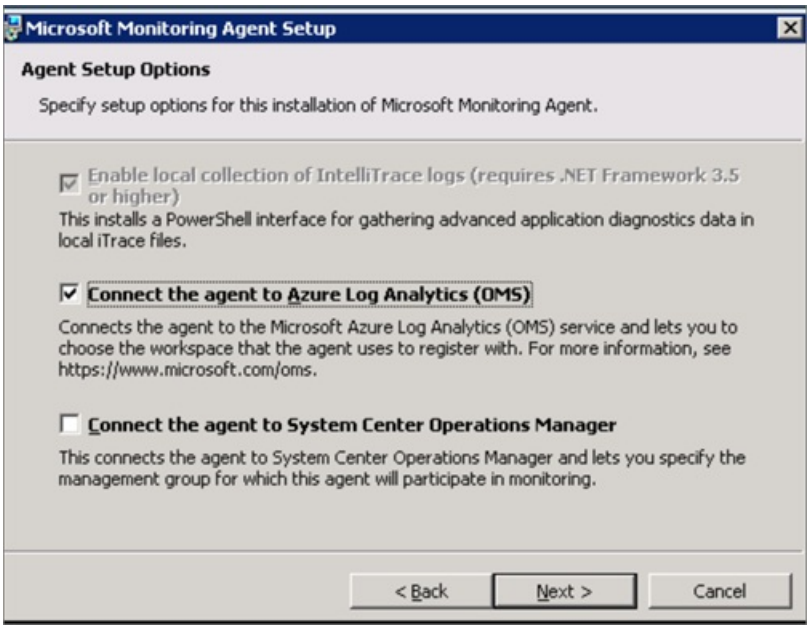


Install the agents on Windows VMs

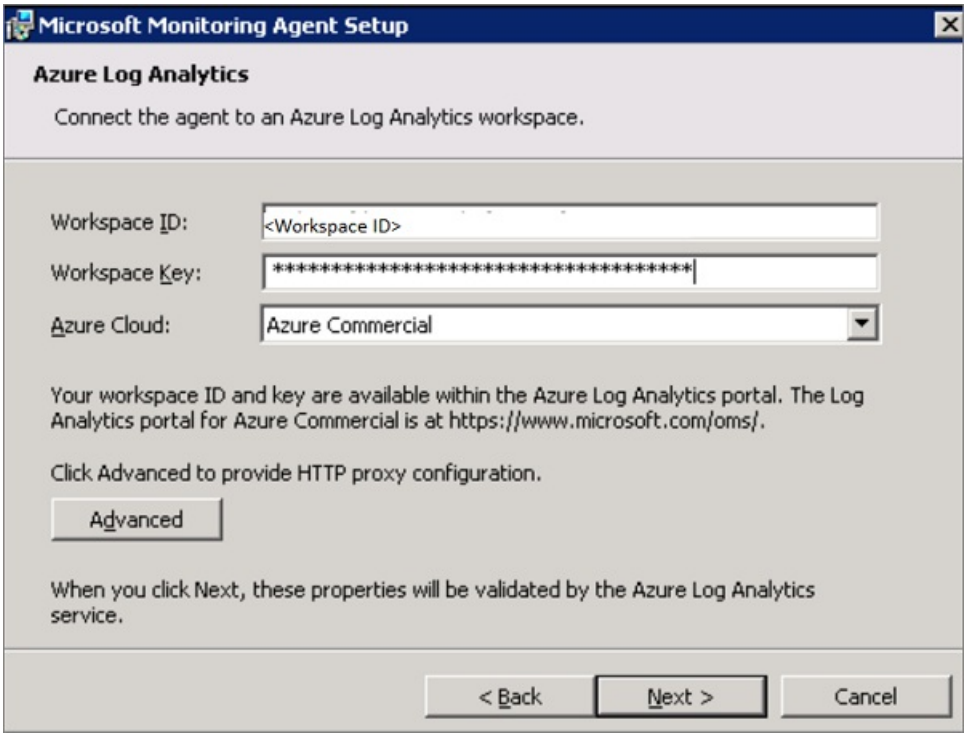
Contoso runs the installation on each VM.

Install the MMA on Windows VMs

1. Contoso double-clicks the downloaded agent.
2. In **Destination Folder**, Contoso keeps the default installation folder, and selects **Next**.
3. In **Agent Setup Options**, Contoso selects **Connect the agent to Azure Log Analytics** > **Next**.



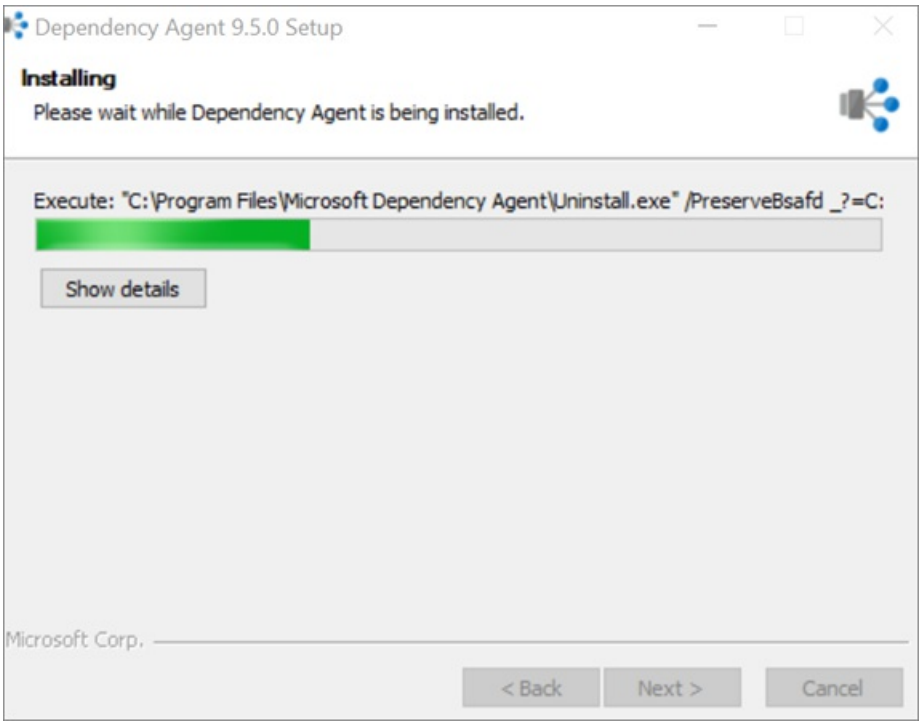
4. In **Azure Log Analytics**, Contoso pastes the workspace ID and key that it copied from the portal.



5. In **Ready to Install**, Contoso installs the MMA.

Install the Microsoft Dependency Agent on Windows VMs

1. Contoso double-clicks the downloaded agent.
2. Contoso accepts the license terms and waits for the installation to finish.



Install the agents on Linux VMs

Contoso runs the installation on each VM.

Install the MMA on Linux VMs

1. Contoso installs the Python ctypes library on each VM by using the following command:

   ```
   sudo apt-get install python-ctypeslib
   ```

2. Contoso must run the command to install the MMA agent as root. To become root, Contoso runs the following command, and then enters the root password:

   ```
   sudo -i
   ```

3. Contoso installs the MMA:

   - Contoso enters the workspace ID and key in the command.
   - Commands are for 64-bit.
   - The workspace ID and primary key are located in the Log Analytics workspace in the Azure portal. Select **Settings**, and select the **Connected Sources** tab.
   - Run the following commands to download the Log Analytics agent, validate the checksum, and install and onboard the agent:

     ```
     wget https://raw.githubusercontent.com/Microsoft/OMS-Agent-for-
     Linux/master/installer/scripts/onboard_agent.sh && sh onboard_agent.sh -w 6b7fcaff-7efb-4356-
     ae06-516cacf5e25d -s
     k7gAMAw5Bk8pFVUTZKmk2lG4eUciswzWfYLDTxGcD8pcyc4oT8c6ZRgsMy3MmsQSHuSOcmBUsCjoRiG2x9A8Mg==
     ```

Install the Microsoft Dependency Agent on Linux VMs

After the Microsoft Monitoring Agent is installed, Contoso installs the Microsoft Dependency Agent on the Linux VMs:

1. The Microsoft Dependency Agent is installed on Linux computers by using `InstallDependencyAgent-Linux64.bin`, a shell script that has a self-extracting binary. Contoso runs the file by using `sh`, or it adds execute permissions to the file itself.

2. Contoso installs the Linux dependency agent as root:

```
wget --content-disposition https://aka.ms/dependencyagentlinux -O InstallDependencyAgent-Linux64.bin
&& sudo sh InstallDependencyAgent-Linux64.bin -s
```

Step 6: Run and analyze the VM assessment

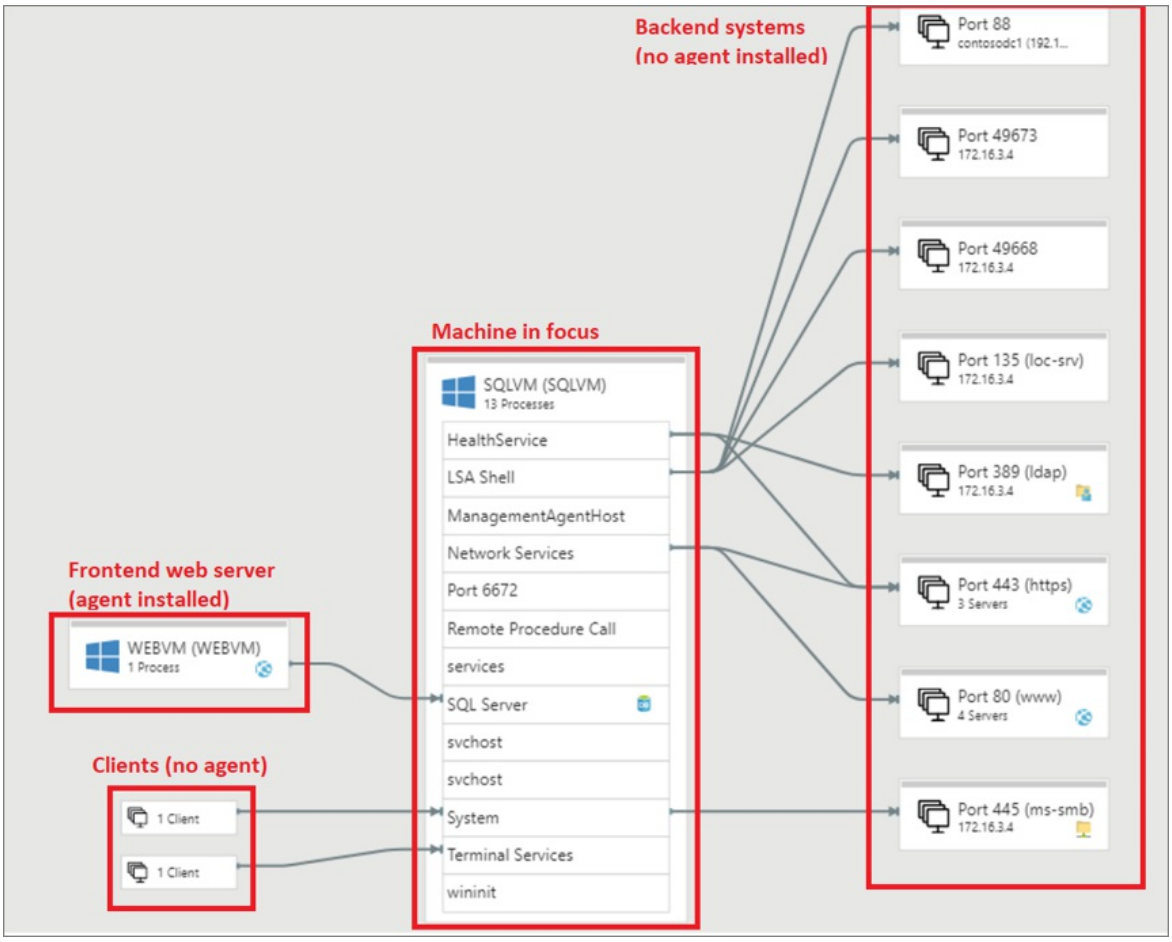
Contoso can now verify machine dependencies and create a group. Then, it runs the assessment for the group.

Verify dependencies and create a group

1. To determine which machines to analyze, Contoso selects **View Dependencies**.



2. For SQLVM, the dependency map shows the following details:
   - Process groups or processes that have active network connections running on SQLVM during the specified time period (an hour, by default).
   - Inbound (client) and outbound (server) TCP connections to and from all dependent machines.
   - Dependent machines that have the Azure Migrate agents installed are shown as separate boxes.
   - Machines that don't have the agents installed show port and IP address information.
3. For machines that have the agent installed (`WEBVM`), Contoso selects the machine box to view more information. The information includes the FQDN, operating system, and MAC address.



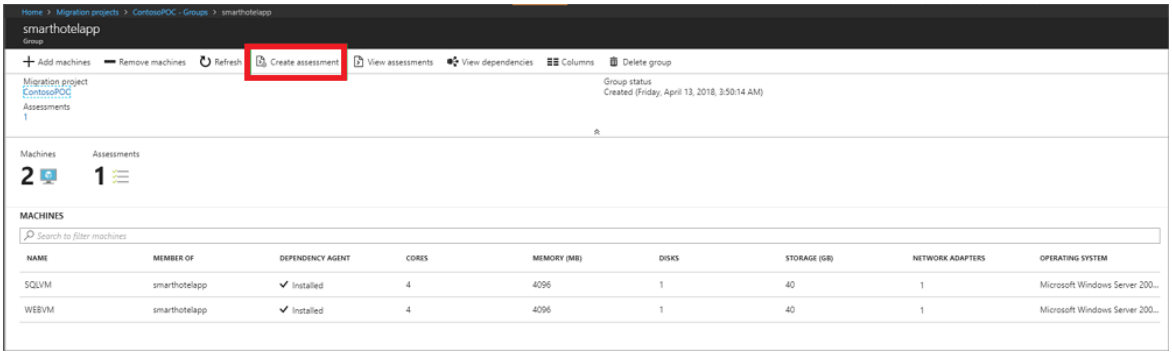
4. Contoso selects the VMs to add to the group (`SQLVM` and `WEBVM`). Contoso holds the `Ctrl` key while selecting multiple VMs.

5. Contoso selects **Create Group**, and then enters a name (`smarthotelapp`).

> **NOTE**
> To view more granular dependencies, you can expand the time range. You can select a specific duration or select start and end dates.

Run an assessment

1. In **Groups**, Contoso opens the group (`smarthotelapp`), then selects **Create assessment**.

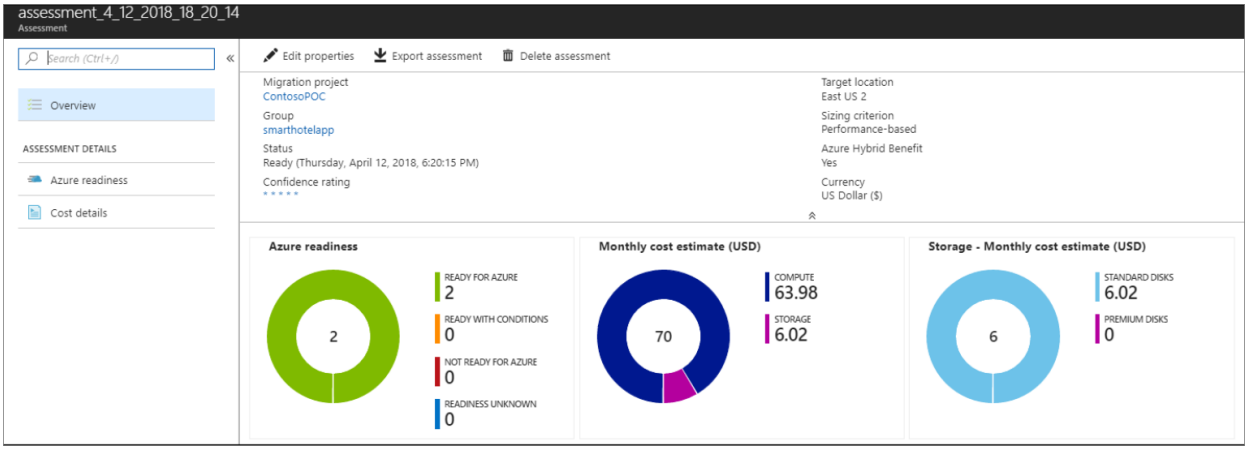


2. To view the assessment, Contoso selects **Manage** > **Assessments**.

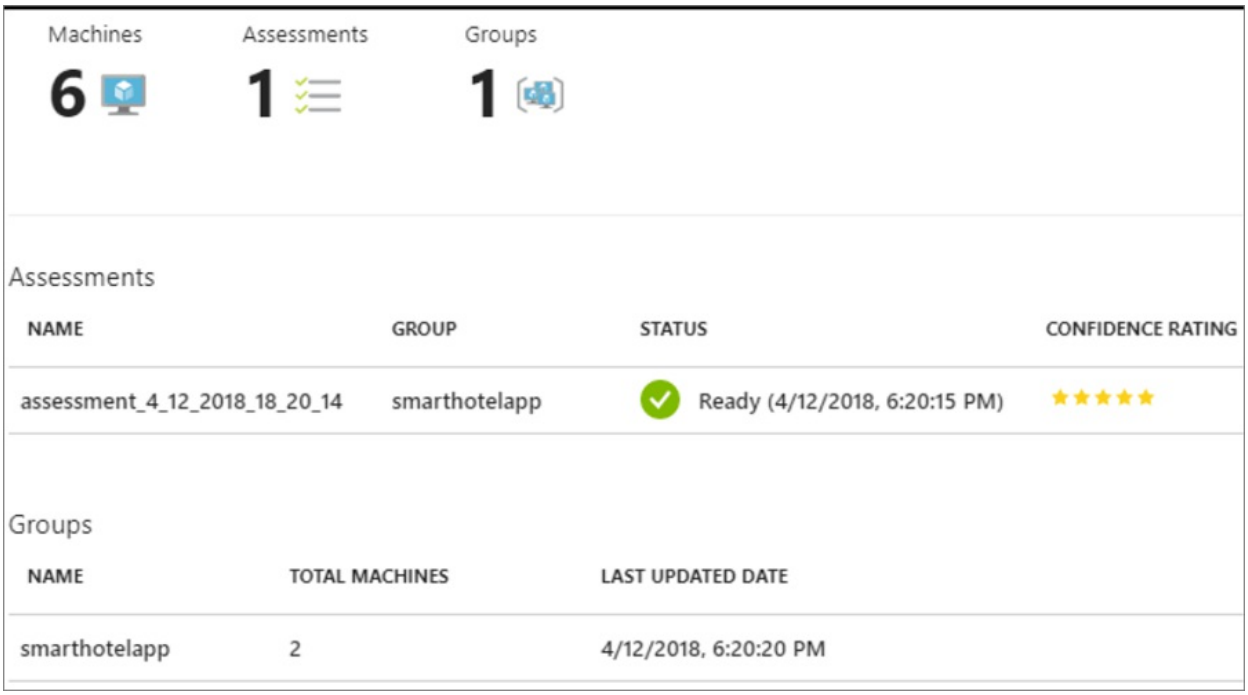
Contoso uses the default assessment settings, but you can customize settings.

Analyze the VM assessment

An Azure Migrate assessment includes information about the compatibility of on-premises with Azure, suggested right-sizing for Azure VM, and estimated monthly Azure costs.



Review confidence rating



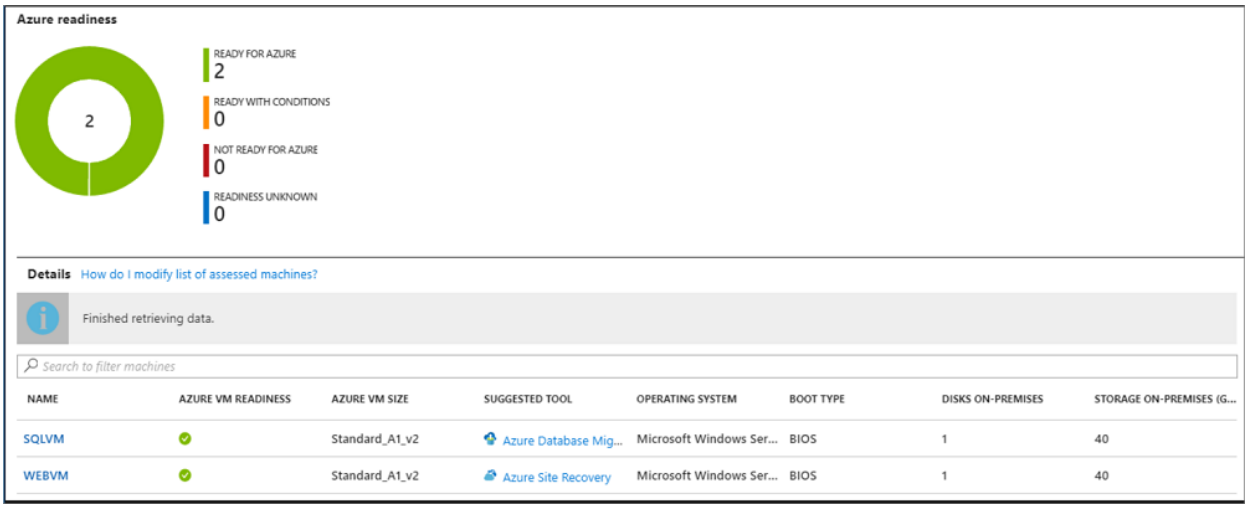
An assessment has a confidence rating of from 1 star to 5 stars (1 star is the lowest and 5 stars is the highest).

- The confidence rating is assigned to an assessment based on the availability of data points that are needed to compute the assessment.
- The rating helps you estimate the reliability of the size recommendations that are provided by Azure Migrate.
- The confidence rating is useful when you are doing *performance-based sizing*. Azure Migrate might not have enough data points for utilization-based sizing. For *as on-premises* sizing, the confidence rating is always 5 stars because Azure Migrate has all the data points it needs to size the VM.
- Depending on the percentage of data points available, the confidence rating for the assessment is provided:

AVAILABILITY OF DATA POINTS	CONFIDENCE RATING
0%-20%	1 star
21%-40%	2 stars
41%-60%	3 stars

AVAILABILITY OF DATA POINTS	CONFIDENCE RATING
61%-80%	4 stars
81%-100%	5 stars

Verify Azure readiness



The assessment report shows the information that's summarized in the table. To show performance-based sizing, Azure Migrate needs the following information. If the information can't be collected, sizing assessment might not be accurate.

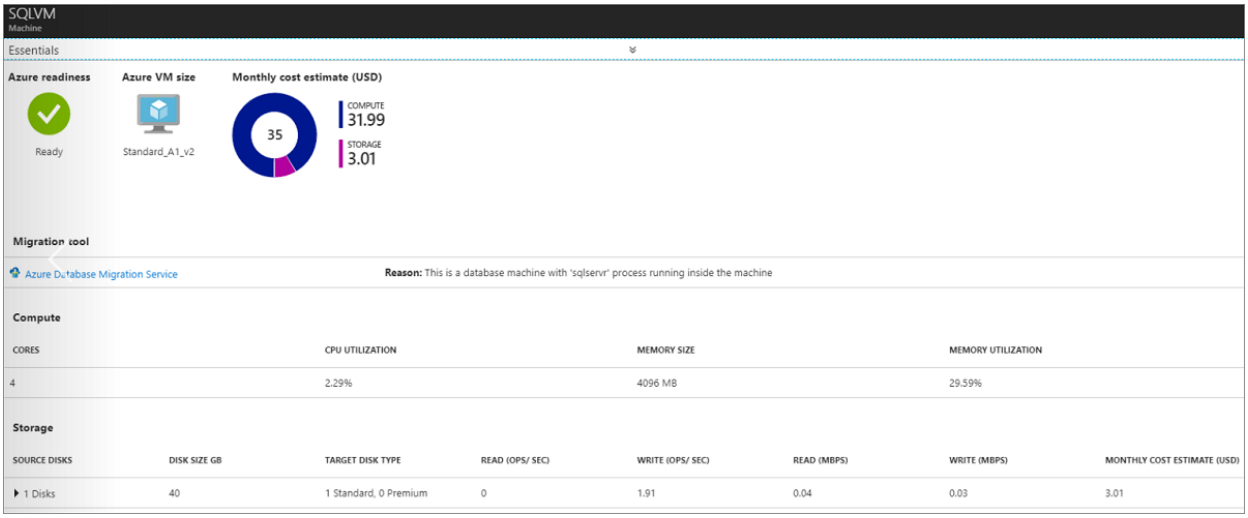
- Utilization data for CPU and memory.
- Read/write IOPS and throughput for each disk attached to the VM.
- Network in/out information for each network adapter attached to the VM.

SETTING	INDICATION	DETAILS
Azure VM readiness	Indicates whether the VM is ready for migration.	Possible states: • Ready for Azure • Ready with conditions • Not ready for Azure • Readiness unknown If a VM isn't ready, Azure Migrate shows some remediation steps.
Azure VM size	For ready VMs, Azure Migrate provides an Azure VM size recommendation.	Sizing recommendation depends on assessment properties: • If you used performance-based sizing, then sizing considers the performance history of the VMs. • If you used *as on-premises* sizing, then sizing is based on the on-premises VM size and utilization. • Data isn't used.
Suggested tool	Because Azure machines are running the agents, Azure Migrate looks at the processes that are running inside the machine. It identifies whether the machine is a database machine.	

SETTING	INDICATION	DETAILS
VM information	The report shows settings for the on-premises VM, including operating system, boot type, and disk and storage information.	

Review monthly cost estimates

This view shows the total compute and storage cost of running the VMs in Azure. It also shows details for each machine.



- Cost estimates are calculated by using the size recommendations for a machine.
- Estimated monthly costs for compute and storage are aggregated for all VMs in the group.

Clean up after assessment

- When the assessment finishes, Contoso retains the Azure Migrate appliance to use in future evaluations.
- Contoso turns off the VMware VM. Contoso will use it again when it evaluates additional VMs.
- Contoso keeps the `Contoso Migration` project in Azure. The project currently is deployed in the `ContosoFailoverRG` resource group in the East US Azure region.
- The collector VM has a 180-day evaluation license. If this limit expires, Contoso will need to download the collector and set it up again.

Conclusion

In this scenario, Contoso assesses its SmartHotel360 app database by using the Data Migration Assistant tool. It assesses the on-premises VMs by using the Azure Migrate service. Contoso reviews the assessments to make sure that on-premises resources are ready for migration to Azure.

Next steps

After Contoso assesses this workload as a potential migration candidate, it can begin preparing its on-premises infrastructure and its Azure infrastructure for migration. See the deploy Azure infrastructure article in the Cloud Adoption Framework migrate best practices section for an example of how Contoso performs these processes.

Plan a data warehouse migration

3/6/2021 • 14 minutes to read • Edit Online

A data warehouse migration is a challenge for any company. In order to execute it well and avoid any unwelcome surprises and unplanned costs, you need to thoroughly research the challenge, mitigate risk, and plan your migration to ensure that you're as ready as possible. At a high level, your plan should cover the core data warehouse migration process steps and any tasks within them. The main process steps are:

- Pre-migration preparation
- Migration strategy and execution
- Post-migration

For example, preparation includes things like readying your data warehouse migration team in terms of skills training and technology familiarization. It also includes setting up a proof of concept lab, understanding how you will manage test and production environments, gaining appropriate clearance to migrate your data and a production system outside of the corporate firewall and setting up migration software in your datacenter to enable migration to proceed.

For a data warehouse migration to proceed smoothly, your plan should establish a clear understanding of:

- Your business case, including its drivers, business benefits, and risks.
- Migration team roles and responsibilities.
- The skill set and training required to enable successful migration.
- Allocated budget for the complete migration.
- Your migration strategy.
- How you can avoid risk in the migration project to avoid delays or rework.
- Your existing data warehouse system, its architecture, schema, data volumes, data flows, security, and operational dependencies.
- Differences between your existing on-premises data warehouse DBMS and Azure Synapse, like data types, SQL functions, logic, and other considerations.
- What needs to be migrated and priorities.
- The migration tasks, approaches, order, and deadlines.
- How you will control migration.
- How to prevent user disruption while undertaking the migration.
- What you need to do on-premises to avoid delays and enable migration.
- Tools to enable secure migration of schemas, data, and ETL processing to Azure.
- Data model design changes that are required during and after migration.
- Any pre-migration or post-migration technology changes and how to minimize rework.
- Post-migration technology deprecation.
- How you will implement testing and quality assurance to prove success.
- Your checkpoints to assess progress and enable decisions to be made.
- Your contingency plan and points of rollback in case things go wrong.

In order to achieve this understanding, we need to prepare and begin specific activities before any migration starts. Let's look at what that entails in more detail.

Pre-migration preparation

There are several things that should be addressed before you even begin a data warehouse migration.

Key roles in a data warehouse migration team

Key roles in a migration project include:

- Business owner
- Project manager (with agile methodology experience such as Scrum)
- Project coordinator
- Cloud engineer
- Database administrator (existing data warehouse DBMS and Azure Synapse)
- Data modelers
- ETL developers
- Data virtualization specialist (possibly a database administrator)
- Testing engineer
- Business analysts (to help test BI tool queries, reports, and analyses)

In addition, the team need the support of your on-premises infrastructure team.

Skills and training to ready the team for migration

With respect to skills, expertise is important in a data warehouse migration. Therefore, ensure the appropriate members of your migration team are trained in Azure cloud fundamentals, Azure Blob storage, Azure Data Lake Storage, Azure Data Box, ExpressRoute, Azure identity management, Azure Data Factory, and Azure Synapse. Your data modelers will most likely need to fine-tune your Microsoft Azure Synapse data models once migration from your existing data warehouse has occurred.

Assessing your existing data warehouse

Another part of preparing to migrate is the need for a full assessment of your existing data warehouse to fully understand the architecture, data stores, schema, business logic, data flows, the DBMS functionality used, warehouse operation, and the dependencies. The more understanding is gained here the better. A detailed knowledge of how the system works helps to communicate and cover off all bases.

The purpose of the assessment is not just to ensure detailed understanding of the current setup across the migration team but also to understand strengths and weaknesses in the current setup. The outcome of an assessment of your current data warehouse therefore can impact your migration strategy in terms of lift and shift versus something broader. For example, if the outcome of an assessment is that your data warehouse is at end of life then clearly the strategy would be more of a data migration to a newly designed data warehouse on Azure Synapse versus a lift-and-shift approach.

On-premises preparation for data migration

In addition to preparing and readying your migration team for your target environment and assessing your current setup, it is equally important to also set things in motion on-premises as production data warehouses tend to be heavily controlled by IT procedures and approval processes. To avoid delays, ensure that your datacenter infrastructure and operations teams are ready for migrating your data, schema, ETL jobs, and so on, to the Azure cloud. Data migration can occur via:

- AzCopy to Azure Blob storage.
- Microsoft Azure ExpressRoute to transfer compressed data directly to Azure.
- File export to Azure Data Box.

The main factors influencing which of these options is selected are data volume size (in terabytes) and network speed (in Mbps). A calculation is needed to determine how long it would take to migrate the data via the network, considering that data might be compressed in your data warehouse and become uncompressed when you export it. This situation can slow data transfer. Recompress data via Gzip when moving data by any of the above methods. PolyBase can process gzipped data directly. Large data volumes will likely be migrated via Azure

Data Box if it will take too long to move the data.

Additionally, for Azure Data Factory to control the execution of exports of your existing data warehouse data from Azure, self-hosted integration run-time software must be installed in your datacenter to enable migration to proceed. Given these requirements if formal approval is needed to make this possible, then starting the appropriate approval processes early to enable this to happen will help avoid delays down the line.

Azure preparation for schema and data migration

In terms of preparation on the Azure side, data import will need to be managed either via Microsoft Azure ExpressRoute or Microsoft Azure Data Box. Azure Data Factory pipelines are an ideal way to load your data into Azure Blob storage and then load from there into Azure Synapse using PolyBase. Therefore preparation is needed on the Azure side to develop such a pipeline.

The alternative is to use your existing ETL tool on Azure if it supports Azure Synapse, which means setting up the tool on Azure from Azure Marketplace and readying a pipeline to import your data and load it into Azure Blob storage.

Defining a migration strategy

Migration goals

In any strategy, there needs to be a set of objectives or goals that should be defined to indicate success. Targets can then be set to achieve these goals and people given responsibility for reaching them. Examples of migration goals and corresponding metrics to set targets for in a cloud data warehouse migration project are shown in the table below:

Types of goal and metric examples:

Improve overall performance

- Data migration performance
- ELT performance
- Data loading performance
- Complex query performance
- Number of concurrent users

Run at lower cost

- Cost of compute by workload, for example, number of compute hours x cost per hour for:
  - Standard reporting
  - Ad hoc queries
  - Batch ELT processing
- Cost of storage (staging, production tables, indexes, temporary space)

Operate with better availability and service levels

- Service-level agreements
- High availability

Improve productively

- Tasks automated, reduced administrative headcount

A successful data warehouse migration can therefore be interpreted as a data warehouse that runs as fast or faster and at lower cost than the legacy system you migrated from. Assigning owners of these goals creates accountability for reaching them. It also ensures that testing in a proof of concept lab (as defined in the de-risking section in this guide) will be deemed successful if the tests identify ways that the goals can be achieved.

Migration approach

You have several strategic options for migrating your existing data warehouse to Azure Synapse:

- Lift and shift your existing data warehouse as-is.
- Simplify your existing data warehouse and then migrate it.
- Completely redesign your data warehouse on Azure Synapse and migrate your data.

The findings of the assessment of your existing data warehouse should significantly influence your strategy. A good assessment outcome might recommend a lift and shift strategy. A mediocre outcome due to a low agility rating might indicate that simplification is needed before migration. A poor outcome might indicate a complete redesign is needed.

Lift and shift leaves your architecture as-is, trying to minimize the work in moving your existing system. If your existing ETL tool already supports Azure Synapse, you might be able to change the target with minimal effort. Nevertheless there will be differences in table types, data types, SQL functions, views, stored procedure business logic etc. These differences and ways around them are detailed in lower-level documents in this migration series.

Simplifying your existing data warehouse prior to migration is about reducing complexity to ease migration. It could include:

- Removing or archiving unused tables before migrating to avoid migrating data that is not used.
- Converting physical data marts to virtual data marts using data virtualization software to reduce what you have to migrate. Conversion also improves agility and reduces total cost of ownership, so it could be considered as modernization during migration.

You can also simplify first and then lift and shift what remains.

Migration scope

Whatever strategy you choose, you should clearly define the scope of the migration, what will be migrated, and whether you'll migrate incrementally or all at once. One example of incremental migration is migrating your data marts first, followed by your data warehouse. This approach would allow you to focus on high-priority business areas while allowing your team to incrementally build expertise as each mart is individually migrated, before migrating the data warehouse itself.

Defining what has to be migrated

Make an inventory of everything that needs to be migrated. This includes schema, data, ETL processes (pipelines), authorization privileges, users, BI tool semantic access layers, and analytic applications. A detail understanding of what's involved in migrating the inventory is provided in each of the lower-level migration articles in this series. Links to these are shown below.

- Schema migration, design, and performance considerations.
- Data migration, ETL processing, and load.
- Access security and data warehouse operations.
- Migration of visualization and reports.
- Minimizing the impact of SQL issues.
- Third-party tools to help you in your data warehouse migration.

If you are uncertain about the best approach, conduct tests in a proof of concept lab to identify optimal techniques. For more information, see the section on de-risking your data warehouse migration project.

Migration control

Data warehouse migration to Azure Synapse involves tasks that need to be conducted:

- On-premises, such as data export.
- On the network, such as data transfer.
- In the Azure cloud, such as data transformation, integration, and load.

The problem is that managing these tasks can be complicated if scripts and utilities are all being developed,

tested, and run independently in both on-premises and Azure environments. It adds complexity especially if version control, test management and migration execution are not coordinated.

You should avoid these complexities and control them from a common place via a source control repository to manage change from development to testing and production. Migration execution will involve tasks that need to be performed on-premises, on the network, and in Azure. Because Azure Synapse is the target environment, controlling migration execution from Azure simplifies management. Use Azure Data Factory to create a migration control pipeline to control execution both on-premises and on Azure. This introduces automation and minimizes errors. Data Factory becomes a migration orchestration tool, not just an enterprise data integration tool.

Other options to control migration available from Microsoft partners running on Azure include data warehouse automation tools to try to automate migration. Vendors like WhereScape and Attunity for example. Most of these automation tools are aimed at a lift- and-shift approach to migration. Even then, there may be some things that may not be supported by such tools, for example, stored procedures. These products and several others are detailed in a separate guide dedicated to third-party tools to help you migrate to Azure Synapse.

Migration testing

The first thing you need for testing is to define a series of tests and a set of required outcomes for each test that need to be run to verify and indicate success. It is important to ensure that all aspects are tested and compared across your existing and migrated systems including:

- Schema
- Data types converted where necessary
- Use user-defined schema in Azure Synapse to distinguish between data warehouse and data mart tables
- Users
- Roles and assignments of users to those roles
- Data access security privileges
- Data privacy and compliance
- Privileges that govern administration capabilities
- Data quality and integrity
- ETL processing that populates Azure Synapse both into the data warehouse and from the data warehouse to any data marts, including testing
- All rows are correct in all tables including history
- Slowly changing dimension processing
- Change data capture processing
- Calculations and aggregations that use functions that could differ across systems
- Results of all known queries, reports, and dashboards
- Performance and scalability
- Analytical functionality
- Costs in the new pay-as-you-go environment

Automate testing as much as possible, making each test repeatable and enabling a consistent approach to evaluating results. If reports and dashboards are inconsistent, then having the ability to compare metadata lineage across original and migrated systems is valuable during migration testing, since it can highlight differences and pinpoint where they occurred when these are not easy to detect.

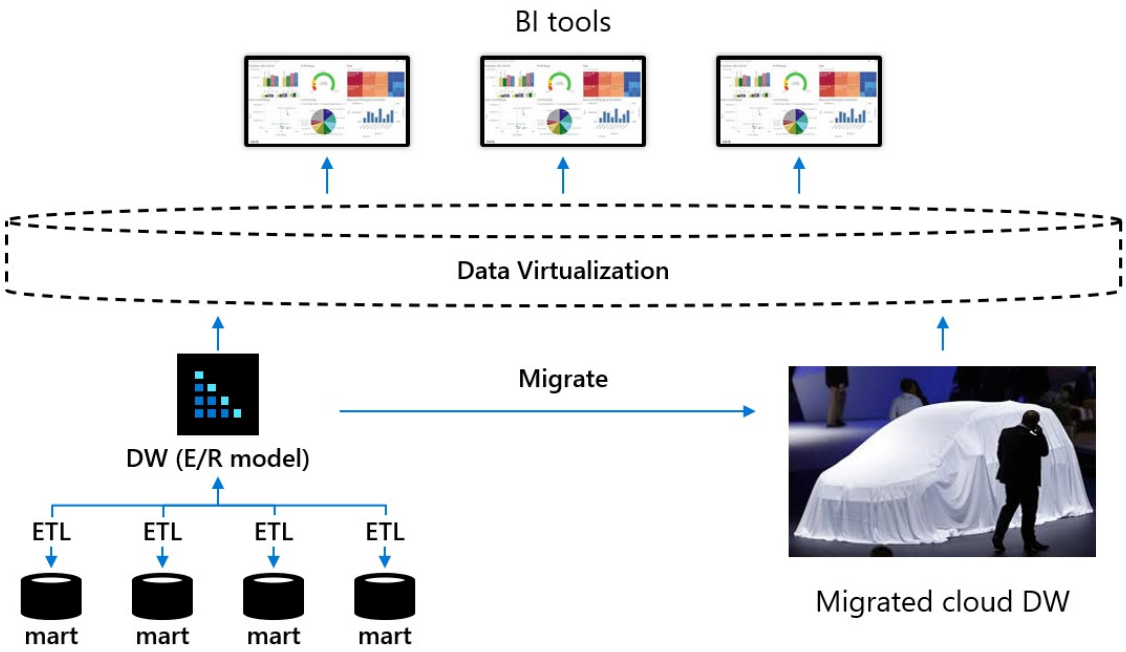
The best way to do this securely is to create roles, assign access privileges to roles and then attach users to roles. To access your newly migrated data warehouse, set up an automated process to create new users and assign roles. Do the same to remove users from roles.

Communicate the cutover to all users so they know what's changing and what to expect.

De-risking your data warehouse migration project

Another critical factor in data warehouse migration is de-risking the project in order to maximize the likelihood of a success. There are several things that can be done to de-risk a data warehouse migration. They include:

- Establishing a proof-of-concept lab to enable your team to try things, conduct tests, understand any issues and identify fixes and optimizations that help that validate migration approaches, improve performance and lower costs. It also helps establish ways to automate tasks, use built-in tools and build templates to capture best practices, learn from experience, and track lessons learned. It's an invaluable way to mitigate risk and increase your chances of success. In addition, you can assign owners to tests who are accountable for achieving migration goals and targets as defined in your migration strategy.
- Introduce data virtualization between BI tools and your data warehouse and data marts. Introduce user transparency using data virtualization to reduce risk in a data warehouse migration, and hide the migration from users by using data virtualization BI tools, as shown in the following diagram.



The purpose of this is to break the dependency between business users using self-service BI tools and the physical schema of the underlying data warehouse and data marts that are being migrated. By introducing data virtualization, any schema alternations made during data warehouse and data mart migration to Azure Synapse (for example, to optimize performance) can be hidden from business users because they only access virtual tables in the data virtualization layer. If structural change is needed, only the mappings between the data warehouse or data marts and any virtual tables would need to be changed so that users remain unaware of those changes and unaware of the migration.

- Look to archive any existing tables identified as never used prior to data warehouse migration as there is little point migrating tables that are not used. One possible way of doing this is to archive the unused data to Azure Blob storage or Azure Data Lake Storage and create external tables in Azure Synapse to that data so that it is still online.
- Consider the possibility of introducing a virtual machine (VM) on Azure with a development version (usually free) of the existing legacy data warehouse DBMS running on this VM. This allows you to quickly move existing data warehouse schema to the VM and them move it on into Azure Synapse while working entirely on the Azure cloud.
- Define migration order and dependencies.
- Ensure your infrastructure and operations teams are ready for the migration of your data as early as possible into the migration project.

- Identify the differences in DBMS functionality and where proprietary business logic could become a problem. For example, using stored procedures for ELT processing is unlikely to migrate easily and won't contain any metadata lineage since the transformations are buried in code.
- Considering a strategy to migrate data marts first followed by the data warehouse that is the source to the data marts. The reason for this is that it enables incremental migration, it makes it more manageable and it is possible to prioritize migration based on business needs.
- Considering the possibility of using data virtualization to simplify your current data warehouse architecture before you migrate, for example, to replace data marts with virtual data marts so that you can eliminate physical data stores and ETL jobs for data marts without losing any functionality prior to migration. Doing this would reduce the number of data stores to migrate, reduce copies of data, reduce the total cost of ownership and improve agility. This requires switching from physical to virtual data marts before migrating your data warehouse. In many ways, you could consider this a data warehouse modernization step prior to migration.

Next steps

For more information on data warehouse migrations, attend a virtual cloud data warehouse modernization workshop on Azure from Informatica.

Strategic Migration Assessment and Readiness Tool (SMART)

9/1/2021 • 2 minutes to read • Edit Online

From business planning to training to security and governance, find out what you need to do to prepare for your Microsoft Azure migration with the Strategic Migration Assessment and Readiness Tool (SMART).

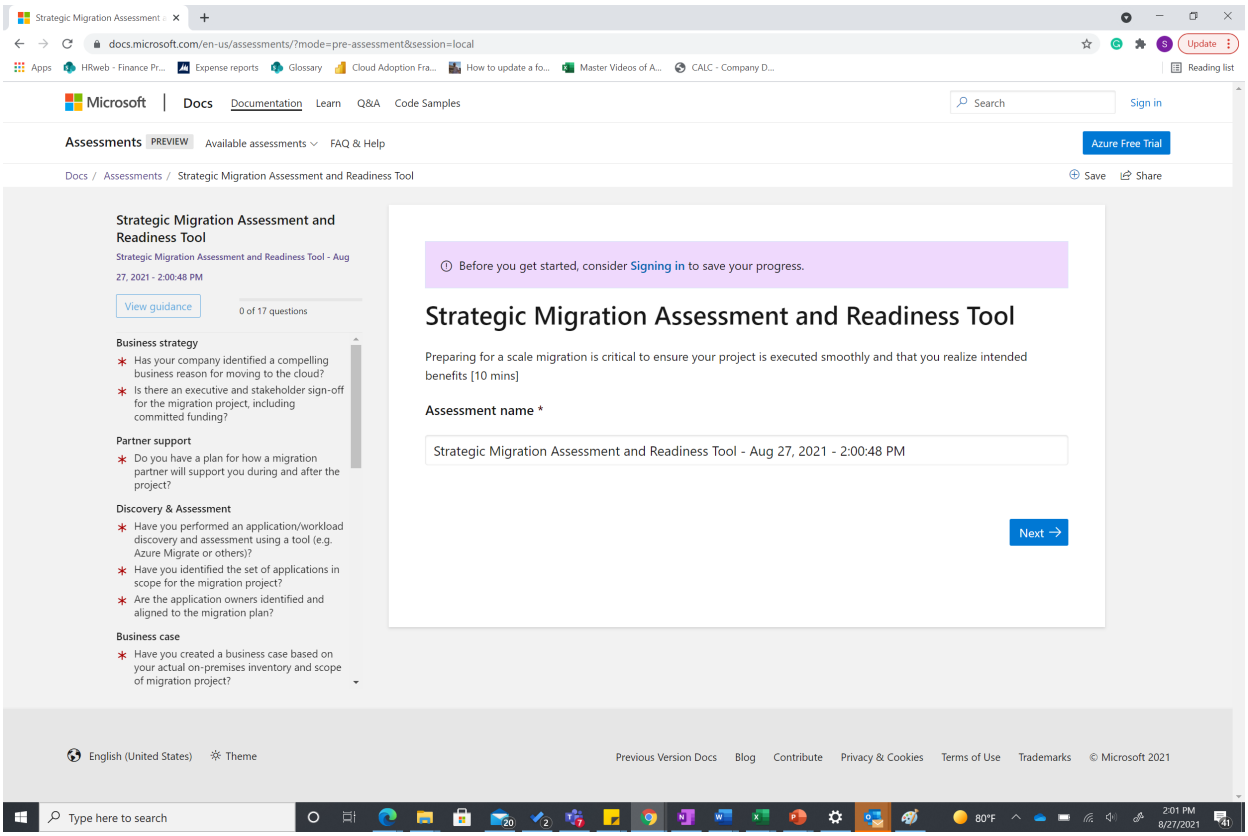
The assessment walks you through a series of questions to help assess your readiness in regards to:

- Your current business strategy
- Your partner support system
- Your inventory and assets
- Your business goals
- Your migration plan
- Your organizations technical and migration skills
- Your landing zone environment
- The governance you have in place
- How you will manage assets and services in the cloud

You have the ability to pick an area of focus, or answer all the assessment questions. When you finish, you'll receive a report that identifies areas for improvement and resources that can help you make those improvements. After you make improvements, you can come back and take the assessment again to gage your progress toward migration readiness.

Access the assessment

You can start the assessment by visiting `https://aka.ms/smarttool`.

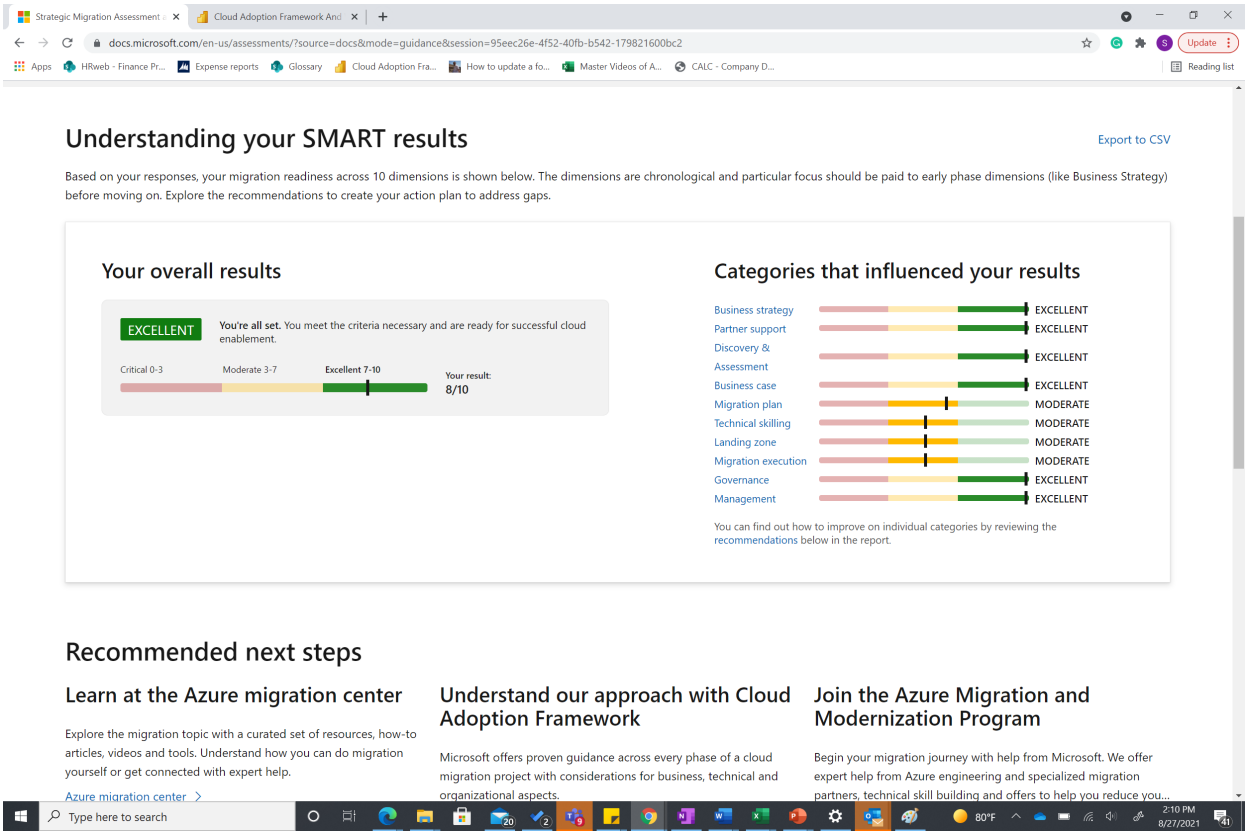


The skills necessary to fill the gaps in your organization's digital transformation typically won't be confined to a single role or even a single department. Skills will have relationships and dependencies that can span a single role or multiple roles. Those roles might exist in several departments. For example, a workload owner might require someone in an IT role to create core resources like subscriptions and resource groups.

To ensure quality results, there are two ways we recommend you approach the 17 questions to ensure a smooth experience. The first option is to complete the assessment with a Microsoft account member. The second option is to set aside about 30 minutes with experts in each respective category and then complete the assessment yourself.

Results

After you finish running the SMART tool, a results page is generated. The results page offers tailored recommendations and next steps to improve your score in each category. There's a tool to export the results to a CSV file that allows you to see scores offline.

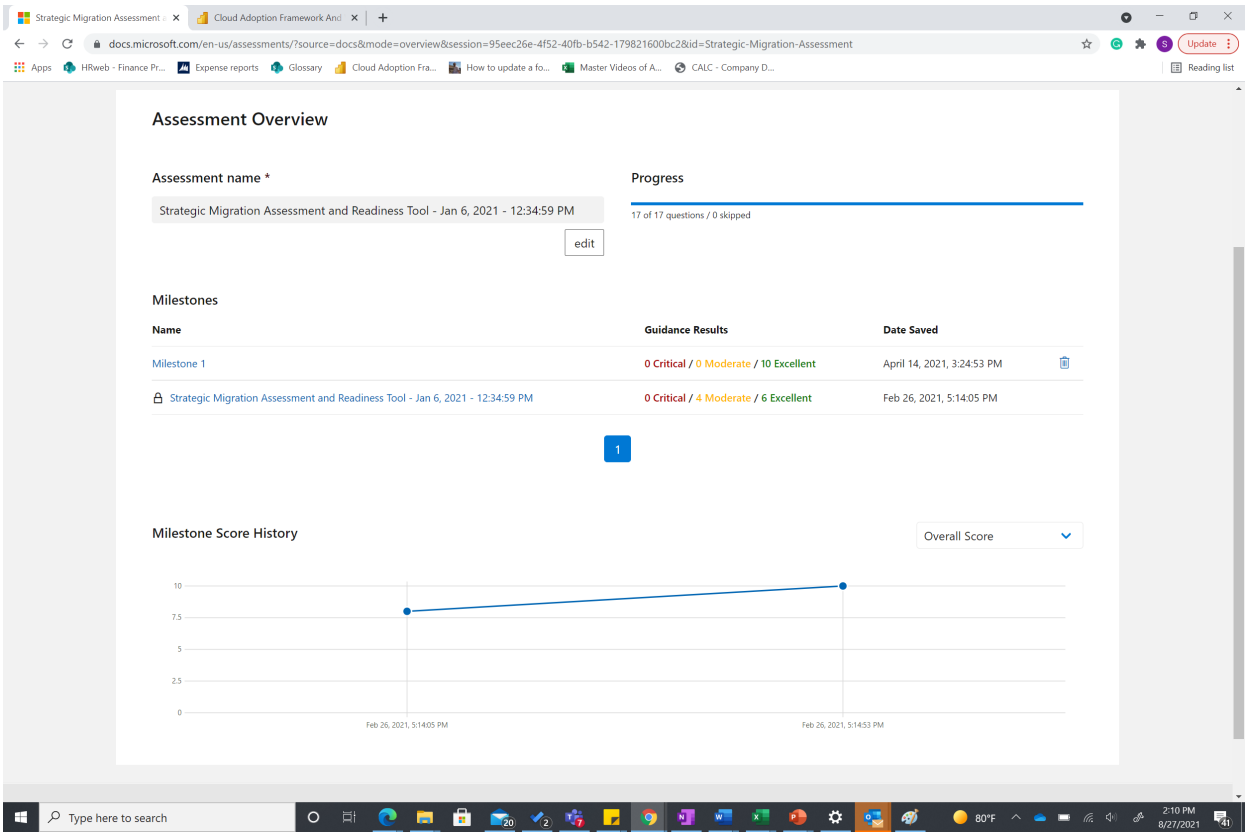


There's a matrix that shows your areas of strength and your opportunities for growth based on how you score in each area. You receive a grade across all 10 dimensions that give you a better understanding of what to do next. There's a **Recommended next steps** section and individual recommendations for each category.

To review your answers, you can go to the answer summary. The answer summary can help you understand why you received specific recommendations or what contributed to the individual scores.

Assessment milestones

The milestone feature of the SMART tool allows you to measure your growth and improvement as you take the assessment multiple times. When you use the new milestone feature, you can look at how you improved individually in each area.



You can compare across all 10 dimensions to see resources that contributed to your evolution.

Next steps

- Strategic Migration Assessment and Readiness Tool (SMART)
- Find out how ready you are to start migrating - the SMART way

Cloud adoption plan antipatterns

3/29/2021 • 4 minutes to read • Edit Online

Customers often experience antipatterns while planning a cloud adoption, for many reasons:

- Misaligned operating models can lead to increased time to market, misunderstandings, and increased pressure on IT departments.
- Companies sometimes choose the wrong service model when they assume that platform as a service (PaaS) decreases costs.
- When an organization's architecture changes, major replacement projects can result. Managing these projects is often complex and cost intensive.

Antipattern: Choose the wrong cloud operating model

A company's strategic priorities and the scope of its portfolio determine its cloud operating model. Models can have different types of accountability, landing zones, and focus. When models don't line up with company goals, problems can result:

- Increased time to market
- Misunderstandings
- Increased pressure on IT departments

Example: Assign too much responsibility to a small team

A corporation introduces an operating model that makes the IT department accountable for everything that runs inside the cloud. The team that's responsible for the cloud contains three people. This setup leads to a slow adoption journey, because:

- The team only approves measures after fully understanding their impact on the business, operations, and security.
- These issues aren't the team's main area of expertise.

Subject matter experts would like to use the cloud service, so business units increase pressure. Shadow IT will probably emerge as business units use company credit cards to create environments for themselves.

Preferred outcome: Compare models and build a readiness plan

Review strategic priorities, portfolio scope, requirements, and constraints. Explore operating model options by comparing the four most common cloud operations patterns with your current cloud operating model. Identify one or more cloud operating models that suit your organization. Then decide on a model. Because roles change with operating models, build a skill readiness plan before moving to the cloud.

Antipattern: Choose the wrong service model

Companies sometimes assume that PaaS solutions cost less than infrastructure as a service (IaaS) solutions. This assumption can lead to the wrong choice of service model. Cost-conscious companies often make this mistake when their main reason for moving to the cloud is to save costs. These companies forget that they also need to change processes when they adopt PaaS, especially when they move certain responsibilities to cloud providers. Switching to PaaS introduces fundamental changes in coordination efforts, engineering practices, and delivery pipelines. Unexpected cost increases and delays can result.

Example: Choose PaaS over IaaS

A publisher launches a program to migrate its datacenters to the cloud. The executives would like to modernize

their current application architecture and tooling all at once. Their reasons include:

- Maximizing cost efficiency.
- Developing a more modern application portfolio.

For their adoption strategy, they choose PaaS over IaaS. A year into their cloud adoption journey, they have a slow adoption rate. They've had to change numerous processes, practices, and tools to adopt PaaS to the full extent. The board doesn't see the usual impacts and benefits associated with PaaS. At the same time, IT is slower than ever, while datacenter costs remain the same.

Preferred outcome: Minimize disruption to your business

To reduce coordination efforts, start with IaaS for initial cloud adoption projects. Adopting new processes and practices is more manageable when you move to the cloud later instead of at the outset. Adopt IaaS first, especially in datacenter transformation scenarios. At the same time, launch a cloud skills initiative.

Gradually modernize and adopt PaaS later, after the workload is already in the cloud. The experience that you've gained will help you adopt PaaS faster. You'll need to learn fewer new skills and processes for modernization. You also won't significantly disrupt your business processes.

Evaluate digital assets according to cloud rationalization. This article describes the most common migration and modernization paths, or the five Rs:

- Rehost
- Refactor
- Rearchitect
- Rebuild
- Replace

Antipattern: Replace architecture

Applications that are based on PaaS and software as a service (SaaS) are relatively easy to maintain. They usually require little effort from management. As a result, many companies redesign old, complex architecture landscapes by replacing them with SaaS and cloud-native concepts. This architecture change usually leads to major replacement projects. It's a complex, cost-intensive task to manage and execute these projects. Changing processes and the operating model also involves other substantial risks.

Example: Choose replacement over modernization

A corporation has a large SAP environment. The IT department would like to replace this landscape, which is causing several performance and stability issues. After IT starts on a replacement project, the due diligence list for replacing the entire environment gets longer every day.

Preferred outcome: Rationalize your digital estate

Before you replace a large or complex application environment, consider incrementally improving your environment by modernizing instead. Relatively small changes to your application environment can have a huge impact on performance and reliability. For instance, changing the hosting platform to Azure can provide stability and quick results. Improved performance and reliability result, at a fraction of the estimated replacement cost.

When deciding on an innovation strategy, explore different modernization options. Evaluate these options in a proof of concept (POC).

Understand your company's digital estate. Determine which of the five Rs of rationalization works best for modernizing or migrating your assets:

- Rehost
- Refactor

- Rearchitect
- Rebuild
- Replace

Next steps

- Compare common cloud operating models
- Build a skill readiness plan
- Cloud rationalization
- What is a digital estate?

Ensure the environment is prepared for the cloud adoption plan

8/25/2021 • 2 minutes to read • Edit Online

Before adoption can begin, you must create a landing zone to host the workloads that you plan to build in or migrate to the cloud . This section of the framework guides you through how a landing zone is created.

The following exercises help guide you through the process of creating a landing zone to support cloud adoption.

	Azure setup guide: Review the Azure setup guide to become familiar with the tools and approaches you need to use to create a landing zone.
	Azure landing zones: Choose the most appropriate landing zone option, to establish a code-based starting point for your environment.
	Expand the landing zone: Meet the platform requirements of your cloud adoption plan by expanding the first landing zone.
	Best practices: Validate landing zone modifications against best practices to ensure the proper configuration of your current and future landing zones.

At a minimum, to get ready for cloud adoption, review the Azure setup guide.

Azure setup guide overview

8/25/2021 • 2 minutes to read • Edit Online

> **NOTE**
> This guide provides a starting point for readiness guidance in the Cloud Adoption Framework and is also available in the Azure Quickstart Center. See the tip in the article for a link.

Before you start building and deploying solutions using Azure services, you need to prepare your environment. In this guide, we introduce features that help you organize resources, control costs, and secure and manage your organization. For more information, best practices, and considerations related to preparing your cloud environment, see the Cloud Adoption Framework's readiness section.

You'll learn how to:

- **Organize resources**: Set up a management hierarchy to consistently apply access control, policy, and compliance to groups of resources and use tagging to track related resources.
- **Manage access**: Use Azure role-based access control to make sure that users have only the permissions they really need.
- **Manage costs and billing**: Identify your subscription type, understand how billing works, and learn how to control costs.
- **Plan for governance, security, and compliance**: Enforce and automate policies and security settings that help you follow applicable legal requirements.
- **Use monitoring and reporting**: Get visibility across resources to find and fix problems, optimize performance, and gain insight into customer behavior.
- **Stay current with Azure**: Track product updates to enable a proactive approach to change management.

> **TIP**
> For an interactive experience, view this guide in the Azure portal. Go to the Azure Quickstart Center in the Azure portal, select **Azure Setup Guide**, and then follow the step-by-step instructions.

Next steps:

- Organize your resources to simplify how you apply settings

This guide provides interactive steps that let you try features as they're introduced. To come back to where you left off, use the breadcrumb for navigation.

Organize your Azure resources effectively

8/25/2021 • 6 minutes to read • Edit Online

Organizing your cloud-based resources is critical to securing, managing, and tracking the costs related to your workloads. To organize your resources, define a management group hierarchy, follow a well-considered naming convention and apply resource tagging.

- Azure management groups and hierarchy
- Naming standards
- Resource tags

Azure provides four levels of management scope: management groups, subscriptions, resource groups, and resources. The following image shows the relationship of these levels.

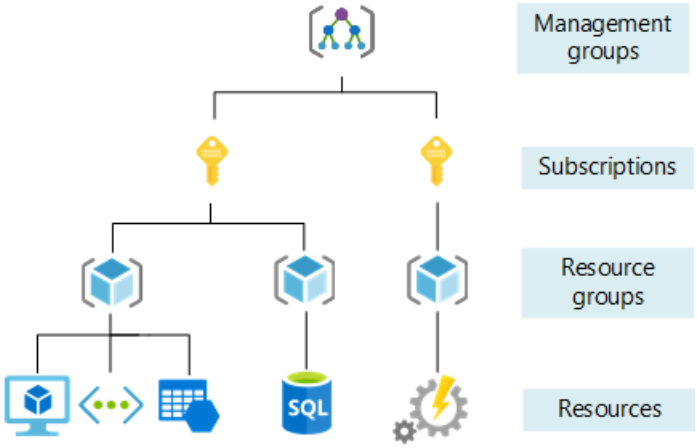


Figure 1: How the four management-scope levels relate to each other.

- **Management groups:** These groups are containers that help you manage access, policy, and compliance for multiple subscriptions. All subscriptions in a management group automatically inherit the conditions applied to the management group.
- **Subscriptions:** A subscription logically associates user accounts and the resources that were created by those user accounts. Each subscription has limits or quotas on the amount of resources you can create and use. Organizations can use subscriptions to manage costs and the resources that are created by users, teams, or projects.
- **Resource groups:** A resource group is a logical container into which Azure resources like web apps, databases, and storage accounts are deployed and managed.
- **Resources:** Resources are instances of services that you create, like virtual machines, storage, or SQL databases.

Scope of management settings

You can apply management settings like policies and Azure role-based access control at any of the management levels. The level you select determines how widely the setting is applied. Lower levels inherit settings from higher levels. For example, when you apply a policy to a subscription, that policy is also applied to all resource groups and resources in that subscription.

Usually, it makes sense to apply critical settings at higher levels and project-specific requirements at lower levels. For example, you might want to make sure all resources for your organization are deployed to certain regions. To do that, apply a policy to the subscription that specifies the allowed locations. As other users in your organization add new resource groups and resources, the allowed locations are automatically enforced. Learn

more about policies in the governance, security, and compliance section of this guide.

If you have only a few subscriptions, it's relatively simple to manage them independently. If the number of subscriptions you use increases, consider creating a management group hierarchy to simplify the management of your subscriptions and resources. For more information, see Organize and manage your Azure subscriptions.

As you plan your compliance strategy, work with people in your organization with these roles: security and compliance, IT administration, enterprise architecture, networking, finance, and procurement.

Create a management level

You can create a management group, more subscriptions, or resource groups.

Create a management group

Create a management group to help you manage access, policy, and compliance for multiple subscriptions.

1. Go to Management groups.
2. Select **Add management group**.

Create a subscription

Use subscriptions to manage costs and resources that are created by users, teams, or projects.

1. Go to Subscriptions.
2. Select **Add**.

> **NOTE**
> Subscriptions can also be created programmatically. For more information, see Programmatically create Azure subscriptions.

Create a resource group

Create a resource group to hold resources like web apps, databases, and storage accounts that share the same lifecycle, permissions, and policies.

1. Go to Resource groups.
2. Select **Add**.
3. Select the **Subscription** that you want your resource group created under.
4. Enter a name for the **Resource group**.
5. Select a **Region** for the resource group location.

Learn more

To learn more, see:

- Azure fundamentals
- Create your initial subscriptions
- Create additional Azure subscriptions to scale your Azure environment
- Organize and manage your Azure subscriptions
- Organize your resources with Azure management groups
- Understand resource access management in Azure
- Subscription service limits

Actions

Create a management group:

Create a management group to help you manage access, policy, and compliance for multiple subscriptions.

1. Go to **Management groups**.

2. Select **Add management group**.

GO TO MANAGEMENT GROUPS

Create an additional subscription:

Use subscriptions to manage costs and resources that are created by users, teams, or projects.

1. Go to **Subscriptions**.
2. Select **Add**.

GO TO SUBSCRIPTIONS

Create a resource group:

Create a resource group to hold resources like web apps, databases, and storage accounts that share the same lifecycle, permissions, and policies.

1. Go to **Resource groups**.
2. Select **Add**.
3. Select the **Subscription** that you want your resource group created under.
4. Enter a name for the **Resource group**.
5. Select a **Region** for the resource group location.

CREATE A RESOURCE GROUP

Manage access to your Azure environment with Azure role-based access control

3/6/2021 • 2 minutes to read • Edit Online

Managing who can access your Azure resources and subscriptions is an important part of your Azure governance strategy, and assigning group-based access rights and privileges is a good practice. Dealing with groups rather than individual users simplifies maintenance of access policies, provides consistent access management across teams, and reduces configuration errors. Azure role-based access control (Azure RBAC) is the primary method of managing access in Azure.

Azure RBAC provides detailed access management of resources in Azure. It helps you manage who has access to Azure resources, what they can do with those resources, and what scopes they can access.

When you plan your access control strategy, grant users the least privilege required to get their work done. The following image shows a suggested pattern for assigning Azure RBAC.

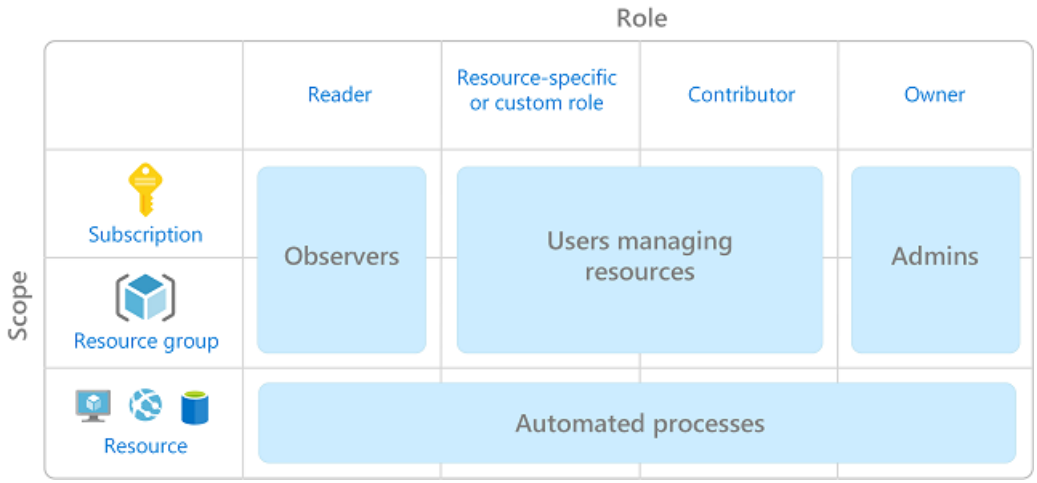


Figure 1: Azure roles.

When you plan your access control methodology, we recommend that you work with people in your organizations with the following roles: security and compliance, IT administration, and enterprise architect.

The Cloud Adoption Framework offers additional guidance on using Azure role-based access control in your cloud adoption efforts.

Actions

Grant resource group access:

To grant a user access to a resource group:

1. Go to **Resource groups**.
2. Select a resource group.
3. Select **Access control (IAM)**.
4. Select **+ Add** > **Add role assignment**.
5. Select a role, and then assign access to a user, group, or service principal.

GO TO RESOURCE GROUPS

Grant subscription access:

To grant a user access to a subscription:

1. Go to **Subscriptions**.
2. Select a subscription.
3. Select **Access control (IAM)**.
4. Select **+ Add** > **Add role assignment**.
5. Select a role, and then assign access to a user, group, or service principal.

GO TO SUBSCRIPTIONS

Grant resource group access

To grant a user access to a resource group:

1. Go to Resource groups.
2. Select a resource group.
3. Select **Access control (IAM)**.
4. Select **+ Add** > **Add role assignment**.
5. Select a role, and then assign access to a user, group, or service principal.

Grant subscription access

To grant a user access to a subscription:

1. Go to Subscriptions.
2. Select a subscription.
3. Select **Access control (IAM)**.
4. Select **+ Add** > **Add role assignment**.
5. Select a role, and then assign access to a user, group, or service principal.

Learn more

To learn more, see:

- What is Azure role-based access control (Azure RBAC)?
- Cloud Adoption Framework: Use Azure role-based access control

Manage costs and billing for your Azure resources

3/6/2021 • 2 minutes to read • Edit Online

Cost management is the process of effectively planning and controlling costs involved in your business. Cost management tasks are typically performed by finance, management, and application teams. Azure Cost Management + Billing can help you plan with cost in mind. It can also help you to analyze costs effectively and take action to optimize cloud spending.

For more information about integrating cloud cost management processes throughout your organization, see the Cloud Adoption Framework article on how to track costs across business units, environments, or projects.

Manage your costs with Azure Cost Management + Billing

Azure Cost Management + Billing provides a few ways to help you predict and manage costs:

- **Analyze cloud costs** helps you explore and analyze your costs. You can view aggregated cost for your account or view accumulated costs over time.
- **Monitor with budgets** allows you to create a budget and then configure alerts to warn you when you're close to exceeding it.
- **Optimize with recommendations** helps identify idle and underused resources so you can take action to reduce waste.
- **Manage invoices and payments** gives you visibility to your cloud investment.

Predict and manage costs

1. Go to Cost Management + Billing.
2. Select **Cost Management**.
3. Explore the features that help to analyze and optimize cloud costs.

Manage invoices and payment methods

1. Go to Cost Management + Billing.
2. Select **Invoices** or **Payment methods** from the **Billing** section in the left pane.

Billing and subscription support

We offer 24-hour access every day for billing and subscription support to Azure customers. If you need assistance to understand Azure usage, create a support request.

Create a support request

To submit a new support request:

1. Go to Help + support.
2. Select **New support request**.

View a support request

To view your support requests and their status:

1. Go to Help + support.
2. Select **All support requests**.

Learn more

To learn more, see:

- Azure Cost Management + Billing documentation
- Cloud Adoption Framework: Track costs across business units, environments, or projects
- Cloud Adoption Framework: Cost Management discipline

Actions

Predict and manage costs:

1. Go to **Cost Management + Billing**.
2. Select **Cost Management**.

Manage invoices and payment methods:

1. Go to **Cost Management + Billing**.
2. Select **Invoices** or **Payment methods** from the **Billing** section in the left pane.

GO TO COST MANAGEMENT + BILLING

Billing and subscription support:

We offer 24-hour access every day for billing and subscription support to Azure customers. If you need assistance to understand Azure usage, create a support request.

Create a support request:

To submit a new support request:

1. Go to **Help + Support**.
2. Select **New support request**.

View a support request: To view your support requests and their status:

1. Go to **Help + Support**.
2. Select **All support requests**.

GO TO HELP + SUPPORT

Governance, security, and compliance in Azure

6/25/2021 • 3 minutes to read • Edit Online

As you establish corporate policy and plan your governance strategies, you can use tools and services like Azure Policy, Azure Blueprints, and Azure Security Center to enforce and automate your organization's governance decisions. Before you start your governance planning, use the governance benchmark tool to identify potential gaps in your organization's cloud governance approach. For more information about developing governance processes, see the Govern methodology.

- Azure Blueprints
- Azure Policy
- Azure Security Center

Azure Blueprints enables cloud architects and central information technology groups to define a repeatable set of Azure resources that implements and adheres to an organization's standards, patterns, and requirements. Azure Blueprints makes it possible for development teams to rapidly build and create new environments and trust that they're building within organizational compliance using a set of built-in components like networking to speed up development and delivery.

Blueprints are a declarative way to orchestrate the deployment of various resource templates and other artifacts like:

- Role assignments.
- Policy assignments.
- Azure Resource Manager templates.
- Resource groups.

Create a blueprint

To create a blueprint:

1. Go to **Blueprints: Getting started**.
2. In the **Create a Blueprint** section, select **Create**.
3. Filter the list of blueprints to select the appropriate blueprint.
4. Enter the **Blueprint name**, then select the appropriate **Definition location**.
5. Select **Next : Artifacts**, then review the artifacts included in the blueprint.
6. Select **Save Draft**.

CREATE A BLUEPRINT

1. In the Azure portal, go to Blueprints: Get started.
2. In the **Create a Blueprint** section, select **Create**.
3. Filter the list of blueprints to select the appropriate blueprint.
4. Enter the **Blueprint name**, then select the appropriate **Definition location**.
5. Select **Next : Artifacts**, then review the artifacts included in the blueprint.
6. Select **Save Draft**.

Publish a blueprint

To publish blueprint artifacts to your subscription:

1. Go to **Blueprints: Blueprint definitions**.
2. Select the blueprint you created in the previous steps.
3. Review the blueprint definition, then select **Publish blueprint**.
4. Provide a **Version** (such as `1.0`) and any **Change notes**, then select **Publish**.



1. In the Azure portal, go to Blueprints: Blueprint definitions.
2. Select the blueprint definition you created in the previous steps.
3. Review the blueprint definition, then select **Publish blueprint**.
4. Provide a **Version** (such as `1.0`) and any **Change notes**, then select **Publish**.

Learn more

To learn more, see:

- Azure Blueprints
- Cloud Adoption Framework: Resource consistency decision guide
- Standards-based blueprints samples

Monitoring and reporting in Azure

3/6/2021 • 4 minutes to read • Edit Online

Azure offers many services that together provide a comprehensive solution for collecting, analyzing, and acting on telemetry from your applications and the Azure resources that support them. In addition, these services can extend to monitoring critical on-premises resources to provide a hybrid monitoring environment.

- Azure Monitor
- Azure Service Health
- Azure Advisor
- Azure Security Center

Azure Monitor provides a single unified hub for all monitoring and diagnostics data in Azure. You can use it to get visibility across your resources. With Azure Monitor, you can find and fix problems and optimize performance. You also can understand customer behavior.

- **Monitor and visualize metrics.** Metrics are numerical values available from Azure resources that help you understand the health of your systems. Customize charts for your dashboards, and use workbooks for reporting.
- **Query and analyze logs.** Logs include activity logs and diagnostic logs from Azure. Collect additional logs from other monitoring and management solutions for your cloud or on-premises resources. Log Analytics provides a central repository to aggregate all this data. From there, you can run queries to help troubleshoot issues or to visualize data.
- **Set up alerts and actions.** Alerts proactively notify you of critical conditions. Corrective actions can be taken based on triggers from metrics, logs, or service health issues. You can set up different notifications and actions and send data to your IT service management tools.

Start monitoring your:

- Applications
- Containers
- Virtual machines
- Networks

To monitor other resources, find additional solutions in the Azure Marketplace.

To explore Azure Monitor, go to the Azure portal.

Learn more

To learn more, see Azure Monitor documentation.

Action

EXPLORE AZURE MONITOR

Stay current with Azure

5/18/2021 • 2 minutes to read • Edit Online

Cloud platforms like Azure change faster than many organizations are accustomed to. This pace of change means that organizations have to adapt people and processes to a new cadence. If you're responsible for helping your organization keep up with change, you might feel overwhelmed at times. The resources listed in this section can help you stay up to date.

- Top resources
- Additional resources

The following resources can help you stay current with Azure:

- **Azure Service Health:** Service Health alerts provide timely notifications about ongoing service issues, planned maintenance, and health advisories. These alerts also includes information about Azure features scheduled for retirement.
- **Azure updates:** Review Azure updates for announcements about product updates. Brief summaries link to additional details, making the updates easy to follow. Subscribe via the Azure updates RSS feed.
- **Azure blog:** The Azure blog communicates the most important announcements for the Azure platform. Follow this blog to stay up to date on critical information. Subscribe via the Azure blog RSS feed.
- **Service-specific blogs:** Many individual Azure services publish blogs that you can follow if you rely on those services. Find the ones you're interested in via a web search.
- **Azure Info Hub:** The unofficial Azure Info Hub pulls together most of the resources listed here. Follow links to individual services to get detailed information and find service-specific blogs. Subscribe via the Azure Info Hub RSS feed. *

Understand cloud operating models

5/4/2021 • 3 minutes to read • Edit Online

Adopting the cloud creates an opportunity to revisit how you operate technology systems. This article series will clarify cloud operating models and the considerations that impact your cloud adoption strategy. But first, let's clarify the term *cloud operating model*.

Define your operating model

Before deploying your cloud architecture, it's important to understand how you want to operate in the cloud. Understanding your strategic direction, people organization, and governance, risk, and compliance (GRC) needs helps define your future state cloud operating model. Then, Azure landing zones can provide a variety of architecture and implementation options to support your operating model. The next few articles will share a few foundational terms and provide examples of common operating models based on actual customer experiences, which together can guide your decision about the right Azure landing zone to implement.

What is an operating model?

Prior to the existence of cloud technologies, technology teams established operating models to define how technology would support the business. Any company's IT operating model has a number of factors, but a few remain consistent: *alignment to business strategy, organization of people, change management (or adoption processes), operations management, governance/compliance, and security*. Each factor is essential to long-term technology operations.

When technology operations shift to the cloud, these vital processes are still relevant, but they're likely to change in some ways. Current operating models focus heavily on physical assets in physical locations funded largely through capital expenditure cycles. These assets are used to support the workloads that the business needs to maintain business operations. The mission of most operating models is to prioritize stability of the workloads by investing in the stability of the underlying physical assets.

How is a cloud operating model different?

Redundancy in the hardware stack is a never-ending cycle. Physical hardware breaks down. Performance degrades. The degradation of hardware rarely aligns with the predictable budgetary cycles of an organization's capital expenditure planning cycles. Operating in the cloud breaks the treadmill of hardware refreshes and midnight patches by shifting the focus upstream to the digital assets: operating systems, applications, and data. This shift from physical to digital also shifts the technology operating model.

As your operating model shifts to the cloud, you still need the same people and processes, but they also shift to focus on a higher level of operations. If your people no longer focus on server uptime, then their success metrics will change. If security is no longer protected by the four walls of a datacenter, then your threat profile changes. When procurement is no longer a blocker to innovation, then the pace at which you manage change also changes.

A *cloud operating model* is the collection of processes and procedures that define how you want to operate technology in the cloud.

Purpose of a cloud operating model

When hardware is removed as the most common unit of operations, the focus shifts to the digital assets and the workloads they support. As such, the purpose of the operating model shifts from keeping the lights on to

ensuring consistent operations.

The Microsoft Azure Well-Architected Framework does a great job of decomposing workload considerations into a set of common architectural principles: cost optimization, operational excellence, performance efficiency, reliability, and security.

When moving to a higher level of operations, these common architectural principles help reframe the purpose of the cloud operating model. How do we ensure that all assets and workloads in the portfolio balance these architecture principles? What processes are needed to scale the application of those principles?

Reimagine your operating model

If you updated your operating model to remove every reference to the procurement, change, operations, or protection of physical assets, what's left? For some organizations, their operating model is now a clean slate. For most organizations, the constraints that have developed over the years are now reduced. In either case, there is an opportunity to think about how you would like to operate in the cloud.

To help you imagine your future state operating model, these articles discuss the following subjects:

- Define your cloud operating model
- Compare common cloud operating models
- Implement your operating model with Azure landing zones

Next steps

Learn how the Cloud Adoption Framework helps you define your operating model.

Compare common cloud operating models

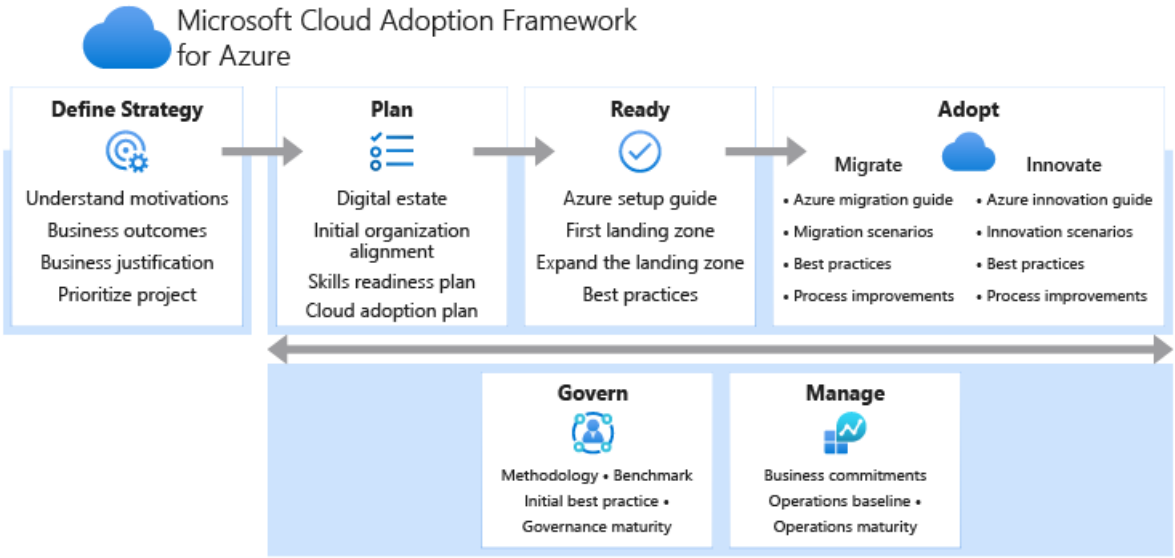
Define your cloud operating model

5/4/2021 • 2 minutes to read • Edit Online

Cloud operating models are complex. Countless customers become blocked by minor details while defining their cloud operating model. It's easy to fall into a series of circular references. To avoid circular references, the Cloud Adoption Framework provides a series of complimentary and incremental methodologies that decompose the volume of decisions into smaller exercises.

Cloud Adoption Framework alignment

To help you define the cloud operating model for your business, the Cloud Adoption Framework breaks down each aspect of the operating model into methodologies. Each methodology and the associated actionable exercises are designed to help you define your future state operations.



Support to develop your operating model

The following areas of the Cloud Adoption Framework are incremental methodologies designed to help grow areas of your operating model.

- Manage: Align ongoing processes for operational management of the technology to maximize value attainment and minimize disruptions.
- Govern: Ensure consistency across adoption efforts. Align to governance or compliance requirements to maintain a cross-cloud environment.
- Security strategy: Help for defining your overall security strategy.
- Organize: Outlines the functions needed in the cloud. Also defines ways to organize people and align responsibilities.

Collective output of the operating model

Your environment should represent how you want to operate. As you define your operating model, environmental readiness should match your operations, governance, security, and organizational requirements.

- Ready: Azure landing zones provide deployment guidance and reference implementations to act on operating model decisions in the form of environmental configuration.

> **NOTE**
>
> The Ready methodology provides several implementation options to Azure landing zones:
>
> - **Start small and expand:** Designed to build your cloud platform as you define each aspect of your operating model.
> - **Enterprise-scale:** Build out an enterprise ready architecture based on a set of defined operating-model decisions.

Dependencies and inputs to operating model decisions

The business strategy and collective cloud adoption plans are inputs that should be considered when defining your operating model.

- Strategy: Guidance to capture business strategy and map those to efforts that can be enabled by a cloud adoption strategy.
- Plan: Agile-based change management guidance to establish backlogs and align ongoing change.

Next steps

Before engaging any of the above methodologies, use the next article to compare common cloud operating models and find a model that closely matches your requirements. That article will help establish the most actionable starting point and set of exercises to move you towards the desired operating model across your cloud platform.

Compare common cloud operating models

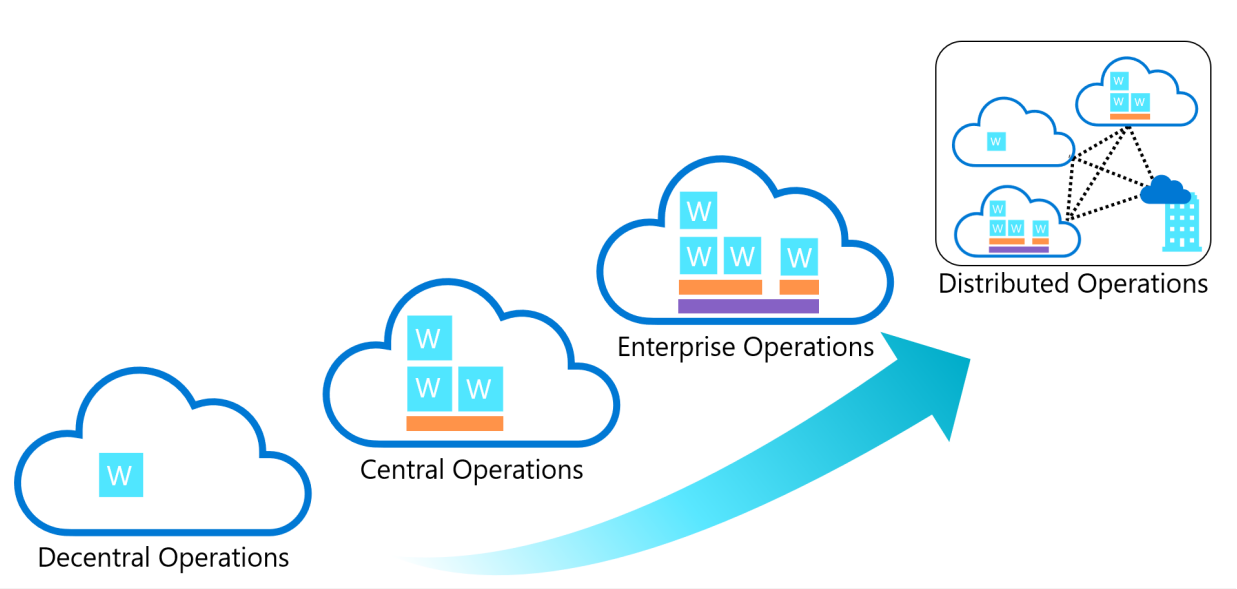
Compare common cloud operating models

5/25/2021 • 21 minutes to read • Edit Online

Operating models are unique and specific to the business they support, based on their current requirements and constraints. But, this uniqueness shouldn't suggest that operating models are *snowflakes*. There are a few common patterns of customer operating models. This article outlines the four most common patterns.

Operating model comparison

The following image maps common operating models based on the range of complexity, from least complex (decentralized) to most complex (global operations). The following tables compares the same operating models based on relative value of a few other attributes.



Priorities or scope

A cloud operating model is primarily driven by two factors:

- Strategic priorities and motivations.
- The scope of the portfolio to be managed.

	DECENTRALIZED OPERATIONS (OPS)	CENTRALIZED OPERATIONS (OPS)	ENTERPRISE OPERATIONS (OPS)	DISTRIBUTED OPERATIONS (OPS)
Strategic priority	Innovation	Control	Democratization	Integration
Portfolio scope	Workload	Landing zone	Cloud platform	Full portfolio
Workload environment	High complexity	Low complexity	Medium complexity	Medium or variable complexity
Landing zone	N/A	High complexity	Medium to low complexity	Low complexity
Foundation utilities	N/A	N/A or low support	Centralized and more support	Most support

	DECENTRALIZED OPERATIONS (OPS)	CENTRALIZED OPERATIONS (OPS)	ENTERPRISE OPERATIONS (OPS)	DISTRIBUTED OPERATIONS (OPS)
Cloud foundation	N/A	N/A	Hybrid, provider specific, or regional foundations	Distributed and synchronized

- **Strategic priorities or motivations**: Each operating model is capable of delivering the typical strategic motivations for cloud adoption. However, some operating models simplify realizing specific motivations.
- Portfolio scope: The portfolio scope row below identifies the largest scope that a specific operating model is designed to support. For example, centralized operations is designed for a small number of landing zones. But that operating model decision could inject operational risks for an organization that's trying to manage a large, complex portfolio that might require many landing zones or variable complexity in landing zone design.

> **IMPORTANT**
> Adopting the cloud often triggers a reflection on the current operating model and might lead to a shift from one of the common operating models to another. But cloud adoption isn't the only trigger. As business priorities and the scope of cloud adoption change how the portfolio needs to be supported, there could be other shifts in the most-appropriately aligned operating model. When the board or other executive teams develop 5 to 10 year business plans, those plans often include a requirement (explicit or implied) to adjust the operating model. While these common models are a good reference for guiding decisions, keep in mind that your operating model might change or you might need to customize one of these models to meet your requirements and constraints.

Accountability alignment

While many teams and individuals will be responsible for supporting different functions, each of the common operating models assigns final accountability for decisions and their outcomes to one team or one individual. This approach affects how the operating model is funded and what level of support is provided for each function.

	DECENTRALIZED OPS	CENTRALIZED OPS	ENTERPRISE OPS	DISTRIBUTED OPS
Business alignment	Workload team	Central cloud strategy	CCoE	Variable - form a broad cloud strategy team?
Cloud operations	Workload team	Central IT	CCoE	Based on portfolio analysis - see Business alignment and business commitments
Cloud governance	Workload team	Central IT	CCoE	Multiple layers of governance
Cloud security	Workload team	Security operations center (SOC)	CCoE + SOC	Mixed - see Define a security strategy
Cloud automation and DevOps	Workload team	Central IT or N/A	CCoE	Based on portfolio analysis - see Business alignment and business commitments

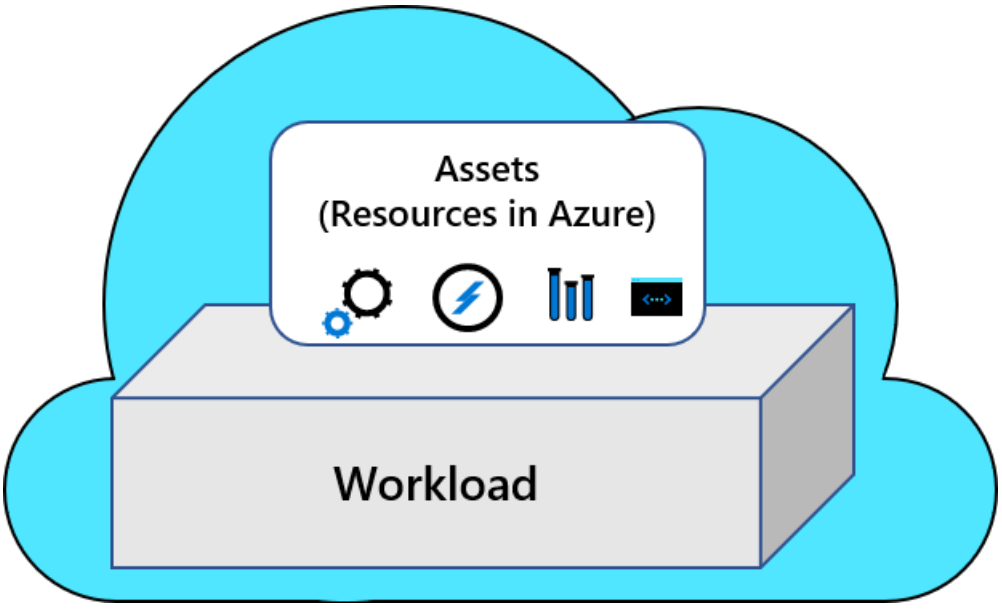
Accelerate operating model implementation in Azure

As discussed in Define your operating model, each methodology of the Cloud Adoption Framework provides a structured path to iteratively developing each aspect of your operating model. Following the most relevant methodology will help you overcome blockers to adoption that stem from gaps in the cloud operating model.

But there are ways to accelerate your operating model implementation, as outlined in the following table.

	DECENTRALIZED OPS	CENTRALIZED OPS	ENTERPRISE OPS	DISTRIBUTED OPS
Starting point	Azure Well-Architected Framework (WAF)	Azure landing zones: start-small options	Azure landing zones: CAF enterprise-scale	Business alignment
Iterations	A focus on workloads allows the team to iterate within WAF.	The start-small option requires additional iteration on each methodology, but that can be done as cloud adoption efforts mature.	As illustrated by the reference implementations, future iterations typically focus on minor configuration additions.	Review the Azure landing zone implementation options to start with the option that best meets your operations baseline. Follow the iteration path defined in that option's design principles.

Decentralized operations



Operations is always complex. By limiting the scope of operations to one workload or a small collection of workloads, that complexity can be controlled. As such, decentralized operations is the least complex of the common operating models. In this form of operations, all workloads are operated independently by dedicated workload teams.

- **Priorities:** Innovation is prioritized over centralized control or standardization across multiple workloads.
- **Distinct advantage:** Maximizes speed of innovation by placing workload and business teams in full control of design, build, and operations.
- **Distinct disadvantage:** Reduction in cross-workload standardization, economies of scale through shared services, and consistent governance centralized compliance efforts.
- **Risk:** This approach introduces risk when managing a portfolio of workloads. Since the workload team is less likely to have specialized teams dedicated to central IT functions, this operating model is viewed as a

high risk option by some organizations, especially companies that are required to follow third-party compliance requirements.

- **Guidance:** Decentralized operations are limited to workload level decisions. The Microsoft Azure Well-Architected Framework is designed to support the decisions made within that scope. The processes and guidance within the Cloud Adoption Framework are likely to add overhead that are not required by decentralized operations.

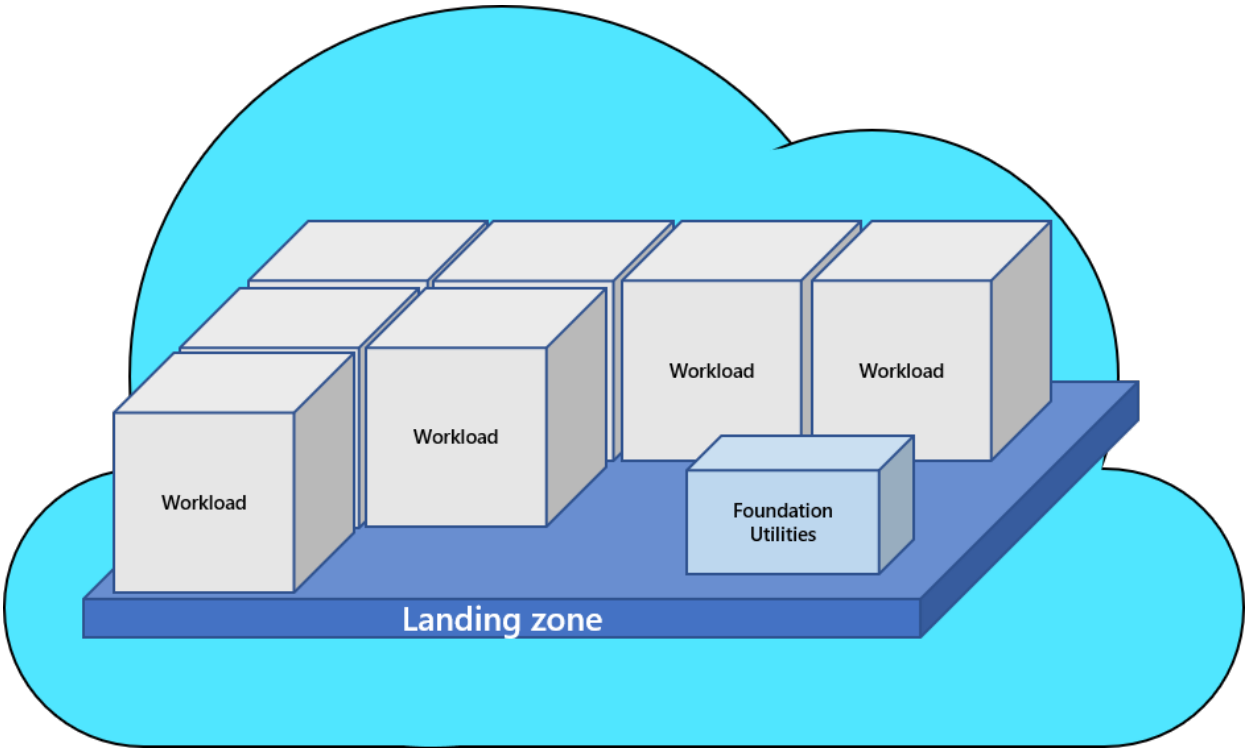
Advantages of decentralized operations

- **Cost management:** Cost of operations can be easily mapped to a single business unit. Workload-specific operations allow for greater workload optimization.
- **Responsibilities:** Typically, this form of operations is highly dependent on automation to minimize overhead. Responsibilities tend to focus on DevOps and pipelines for release management. This allows for faster deployments and shorter feedback cycles during development.
- **Standardization:** Source code and deployment pipeline should be used standardize the environment from release to release.
- **Operations support:** Decisions that impact operations are only concerned with the needs of that workload, simplifying operations decisions. Many in the DevOps community would argue that this is the purest form of operations because of the tighter operational scope.
- **Expertise:** DevOps and development teams are most empowered by this approach and experience the least resistance to driving market change.
- **Landing zone design:** No specific operational advantage.
- **Foundational utilities:** No specific operational advantage.
- **Separation of duties:** No specific operational advantage.

Disadvantages of decentralized operations

- **Cost management:** Enterprise costs are harder to calculate. Lack of centralized governance teams make it harder to implement uniform cost controls or optimization. At scale, this model can be costly, since each workload would likely have duplication in deployed assets and staffing assignments.
- **Responsibilities:** Lack of centralized supporting teams means that the workload team is entirely responsible for governance, security, operations, and change management. This is a detriment when those tasks have not been automated in code review and release pipelines.
- **Standardization:** Standardization across a portfolio of workloads can become variable and inconsistent.
- **Operations support:** Scale efficiencies are often missed. As our uniform best practices across multiple workloads.
- **Expertise:** Team members have a greater responsibility to make wise and ethical decisions regarding governance, security, operations, and change management decisions within the application design and configuration. The Microsoft Azure Well-Architected Review and Azure Well-Architected Framework should be consulted frequently to improve the required expertise.
- **Landing zone design:** Landing zones are not workload-specific and are not considered in this approach.
- **Foundational utilities:** Few (if any) foundational services are shared across workloads, reducing scale efficiencies.
- **Separation of duties:** Higher requirements for DevOps and development teams increase the usage of elevated privileges from those teams. If separation of duties is required, heavy investment in DevOps maturity will be needed to operate in this approach.

Centralized operations



Stable state environments might not require as much focus on the architecture or distinct operational requirements of the individual workloads. Central operations tend to be the norm for technology environments that consist primarily of stable-state workloads. Examples of a stable-state of operations include things like commercial-off-the-shelf (COTS) applications or well-established custom applications that have a slow release cadence. When rate of change is driven by a regular drumbeat of updates and patches (over the high change rate of innovation), centralization of operations is an effective means to manage the portfolio.

- **Priorities:** Prioritizes central control over innovation. Also prioritizes continuation of existing operational processes over cultural shift to modern cloud operations options.
- **Distinct advantage:** Centralization introduces economies of scale, best-of-breed controls, and standardized operations. This approach works best with the cloud environment needs specific configurations integrate cloud operations into existing operations and processes. This approach is most advantageous to centralized teams with a portfolio of a few hundred workloads with modest architectural complexity and compliance requirements.
- **Distinct disadvantage:** Scaling to meet the demands of a large portfolio of workloads can place significant strain on a centralized team making operational decisions for production workloads. If technical assets are expected to scale beyond 1,000 or so VMs, applications, or data sources in the cloud within the next 18-24 months, an enterprise model should be considered.
- **Risk:** This approach limits centralization to a smaller number of subscriptions (often one production subscription). There is a risk of significant refactoring later in the cloud journey that could interfere with adoption plans. Specifically, care should be given to segmentation, environment boundaries, identity tooling, and other foundational elements to avoid significant rework in the future.
- **Guidance:** Azure landing zone implementation options aligned to the "start small and expand" development velocity creates a sound starting point. Those can be used to accelerate adoption efforts. To be successful, clear policies must be established to guide early adoption efforts within acceptable risk tolerances. Govern and Manage methodologies create processes to mature operations in parallel. Those steps serve as stage gates that must be completed before allowing for increased risk as operations matures.

Advantages of centralized operations

- **Cost management:** Centralizing shared services across a number of workloads creates economies of scale and eliminates duplicated tasks. Central teams can more quickly implement cost reductions through enterprise-wide sizing and scale optimizations.
- **Responsibilities:** Centralized expertize and standardization is likely to lead to higher stability, better

operational performance, and lower risk of change-related outages. This reduces broad skilling pressures on the workload focused teams.

- **Standardization:** In general, standardization and cost of operations is lowest with a centralized model because there are fewer duplicated systems or tasks.
- **Operations support:** Reducing complexity and centralizing operations makes it easy for smaller IT teams to support operations.
- **Expertise:** Centralization of supporting teams allow for experts in the fields of security, risk, governance, and operations to drive business critical decisions.
- **Landing zone design:** Central IT tends to minimize the number of landing zones and subscriptions to reduce complexity. Landing zone designs tend to mimic the preceding datacenter designs, which reduces transition time. As adoption progresses, shared resources can then be moved out into a separate subscription or platform foundation.
- **Foundational utilities:** Carrying existing datacenter designs into the cloud results in foundational, shared services that mimic on-premises tools and operations. When on-premises operations are your primary operating model, this can be an advantage (and beware the following disadvantages). Doing so reduces transition time, capitalizes on economies of scale, and allows for consistent operational processes between on-premises and cloud hosted workloads. This approach can reduce short-term complexity/effort and allow smaller teams to support cloud operations with reduced learning curves.
- **Separation of duties:** Separation of duties is clear in central operations. Central IT maintains control of the production environments reducing the need for any elevated permissions from other teams. This reduces the surface area of breach by reducing the number of accounts with elevated privileges.

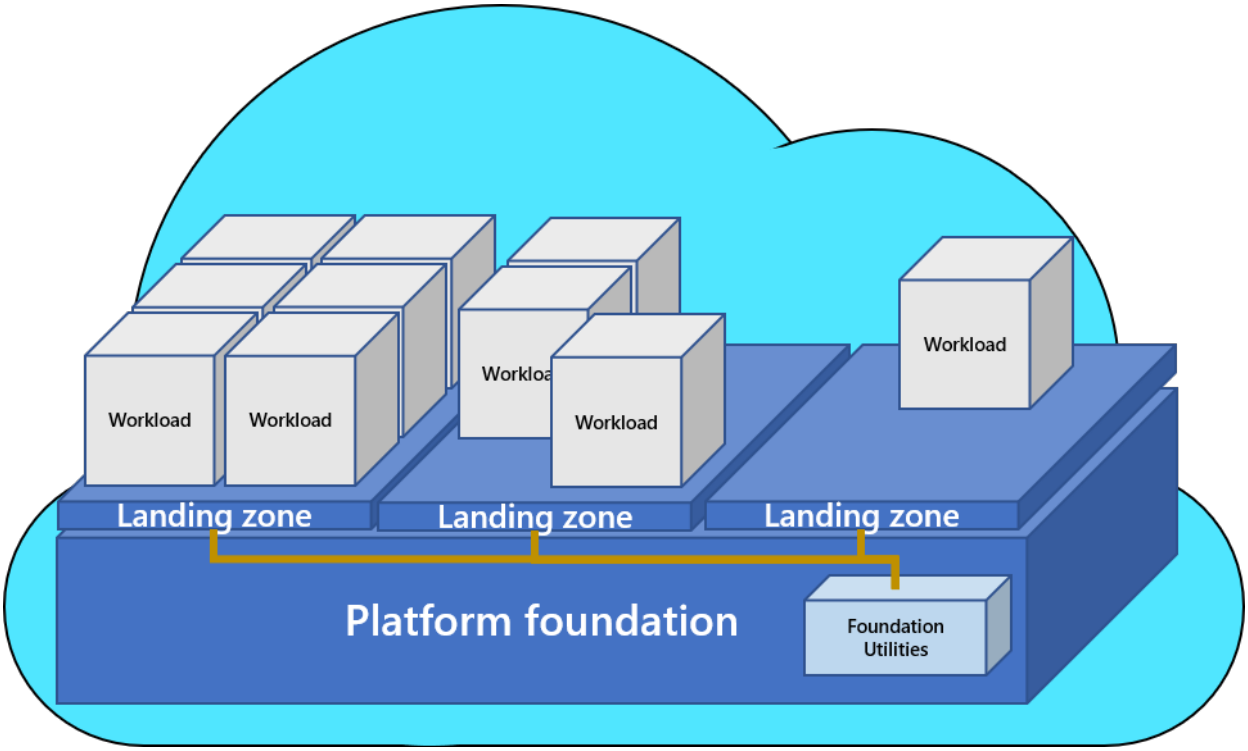
Disadvantages of centralized operations

- **Cost management:** Central teams rarely have enough understanding of the workload architectures to produce impactful optimizations at the workload level. This limits the amount of cost savings that can come from well-tuned workload operations. Further, lack of workload architecture understanding can cause centralized cost optimizations to have a direct impact on performance, scale, or other pillars of a well-architected workload. Before applying enterprise-wide cost changes to high profile workloads, the Microsoft Azure Well-Architected Review should be completed and considered by the central IT team.
- **Responsibilities:** Centralizing production support and access places a higher operational burden on a smaller number of people. It also places greater pressure on those individuals to perform deeper reviews of the deployed workloads to validate adherence to detailed security governance and compliance requirements.
- **Standardization:** Central IT approaches make it difficult to scale standardization without a linear scaling of central IT staff.
- **Operations support:** Not the disadvantage and risks listed above. The greatest disadvantages of this approach are associated with significant scale and shifts that prioritize innovation.
- **Expertise:** Developer and DevOps experts are at risk of being under-valued or too constrained in this type of environment.
- **Landing zone design:** Datacenter designs are based on the constraints of preceding approaches, which aren't always relevant to the cloud. Following this approach reduces the opportunities to rethink environment segmentation and empower innovation opportunities. Lack of landing zone segmentation also increases the potential impact of breach, increases complexity of governance/compliance adherence, and could create blockers to adoption later in the cloud journey. See the risks section above.
- **Foundational utilities:** During digital transformation, cloud might become the primary operating model. Persisting central tools built for on-premises operations reduces opportunities to modernize operations and drive increased operational efficiencies. Choosing not to modernize operations early in the adoption process can be overcome through creation of a platform foundations subscription later in the cloud adoption journey, but that effort can be complex, costly, and time consuming without advanced planning.
- **Separation of duties:** Central operations generally follow one of two paths, both of which can hinder innovation.
  - **Option 1:** Teams outside of central IT are granted limited access to development environments that

mimic production. This option hinders experimentation.

- **Option 2:** Teams develop and test in non-supported environments. This option hinders deployment processes and slows post-deployment integration testing.

Enterprise operations



Enterprise operations is the suggested target state for all cloud operations. Enterprise operations balances the need for control and innovation by democratizing decisions and responsibilities. Central IT is replaced by a more facilitative cloud center of excellence or CCoE team, which supports workload teams and hold them accountable for decisions, as opposed to controlling or limiting their actions. Workload teams are granted more power and more responsibility to drive innovation, within well-defined guardrails.

- **Priorities:** Prioritizes democratization of technical decisions. Democratization of technical decisions shifts responsibilities previously held by central IT to workload teams when applicable. To deliver this shift in priorities, decisions become less dependent on human-run review processes and more dependent on automated review, governance, and enforcement using cloud-native tools.
- **Distinct advantage:** Segmentation of environments and separation of duties allow for balance between control and innovation. Central operations can maintain centralized operations for workloads that require increase compliance, stable state operations, or represent greater security risks. Conversely, this approach allows for reduction in centralized control of workloads and environments that require greater innovation. Since larger portfolios are more likely to struggle with the balance between control and innovation, this flexibility makes it easier to scale to thousands of workloads with reductions in operational pains.
- **Distinct disadvantage:** What worked well on-premises might not work well in enterprise cloud operations. This approach to operations requires changes on many fronts. Cultural shifts in control and responsibility are often the biggest challenge. Operational shifts that follow the cultural shift take time and committed effort to implement, mature, and stabilize. Architectural shifts are sometimes required in otherwise stable workloads. Tooling shifts are required to empower and support the cultural, operational, and architectural shifts, which might require commitments to a primary cloud provider. Adoption efforts made prior to these changes might require significant rework that goes beyond typical refactoring efforts.
- **Risk:** This approach requires executive commitment to the change strategy. It also requires commitment from the technical teams to overcome learning curves and deliver the required change. Long-term cooperation between business, CCoE/central IT, and workload teams is required to see long-term benefits.
- **Guidance:** Azure landing zone implementation options defined as "enterprise-scale" provide reference

implementations to demonstrate how the technical changes are delivered using cloud-native tooling in Azure. The enterprise-scale approach guides teams through the operational and cultural shifts required to take full advantage of those implementations. That same approach can be used to tailor the reference architecture to configure the environment to meet your adoption strategy and compliance constraints. Once enterprise-scale has been implemented, the Govern and Manage methodologies can be used to define processes and expand your compliance and operations capabilities to meet your specific operational needs.

Advantages of enterprise operations

- **Cost management:** Central teams act on cross-portfolio optimizations and hold individual workload teams accountable for deeper workload optimization. Workload focused teams are empowered to make decisions and provided clarity when those decisions have a negative cost impact. Central and workload teams share accountability for cost decisions at the right level.
- **Responsibilities:** Central teams use cloud-native tools to define, enforce, and automate guardrails. Efforts of the workload teams are accelerated through CCoE automation and practices. The workload teams are then empowered to drive innovation and make decisions within those guardrails.
- **Standardization:** Centralized guardrails and foundational services create consistency across all environments, regardless of scale.
- **Operations support:** Workloads that require centralized operations support are segmented to environments with stable-state controls. Segmentation and separation of duties empower workload teams to take accountability for operational support in their own dedicated environments. Automated, cloud native tools ensure a minimum operations baseline for all environments with centralized operational support for the baseline offering.
- **Expertise:** Centralization of core services such as security, risk, governance, and operations ensures proper central expertise. Clear processes and guardrails educates and empowers all members of the workload teams to make more detailed decisions that expand the impact of the centralized experts, without needing to scale that staff linearly with technology scale.
- **Landing zone design:** Landing zone design replicates the needs of the portfolio, creating clear security, governance, and accountability boundaries required to operate workloads in the cloud. Segmentation practices are unlikely to resemble the constraints created by preceding datacenter designs. In enterprise operations, landing-zone design is less complex, allowing for faster scale and reduced barriers to self-service demand.
- **Foundational utilities:** Foundational utilities are hosted in separate centrally controlled subscriptions referred to as the platform foundation. Central tools are then "piped" into each landing zone as a utility services. Separating foundational utilities from the landing zones maximizes consistency, economy of scale, and creates clear distinctions between centrally managed responsibilities and workload level responsibilities.
- **Separation of duties:** Clear separation of duties between foundational utilities and landing zones is one of the biggest advantages of this approach to operations. Cloud-native tools and sound processes allow for just-in-time access and proper balance of control between centralized teams and workload teams, based on the requirements of the individual landing zones and the workloads hosted in those landing zone segments.

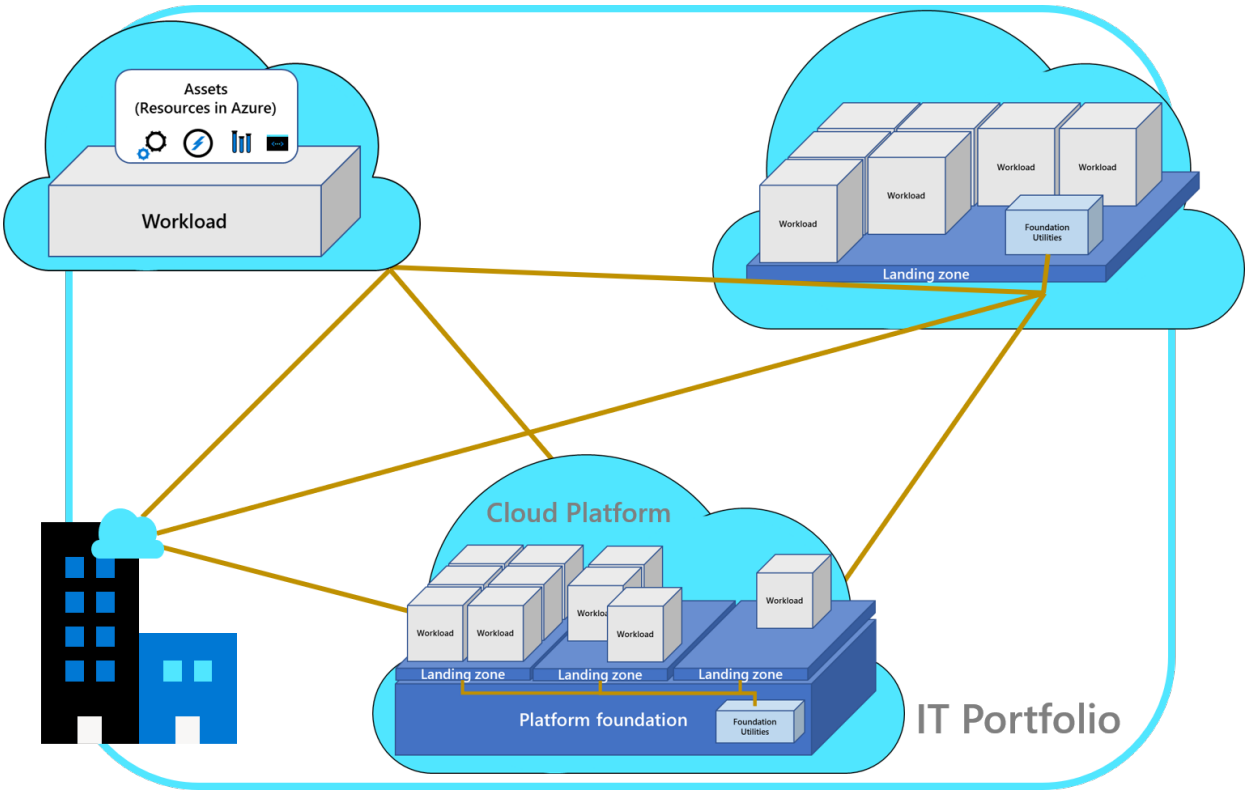
Disadvantages of enterprise operations

- **Cost management:** Central teams are more dependent on workload teams to make production changes within landing zones. This shift does create a risk of potential budget overruns and slower right-sizing of actual spend. Cost control processes, clear budgets, automated controls, and regular reviews must be in place early to avoid cost surprises.
- **Responsibilities:** Enterprise operations requires greater cultural and operational requirements on central and workload teams to ensure clarity in responsibilities and accountability between teams.
- Traditional change management processes or change advisory boards (cabs) are less likely to maintain the pace and balance required in this operating model. Those processes should be reflected in the automation of processes and procedures to safely scale cloud adoption.
- Lack of commitment to change will first materialize in negotiation and alignment of responsibilities. Inability

to align on shifts in responsibility is an indication that central IT operating models might be required during short-term cloud adoption efforts.

- **Standardization:** Lack of investment in centralized guardrails or automation create risks to standardization that are more difficult to overcome through manual review processes. Additionally, operational dependencies between workloads in the landing zones and shared services in the platform foundation creates greater risk to standardization during upgrade cycles or future versions of the foundational utilities. During platform foundation revisions, improved or even automated testing is required of all supported landing zones and the workloads they host.
- **Operations support:** The operations baseline provided through automation and centralized operations might be sufficient for low impact or low criticality workloads. However, workload teams or other forms of dedicated operations will likely be required for complex or high criticality workloads. This might necessitate a shift in operations budgets, requiring business units to allocate operating expenses to those forms of advanced operations. If central IT is required to maintain sole accountability for the cost of operations, then enterprise operations will be difficult to implement.
- **Expertise:** Central IT team members will be required to develop expertise regarding automation of central controls previously delivered via manual processes. Those teams might also need to develop a proficiency for infrastructure-as-code approaches to defining the environment, along with an understanding of branching, merging, and deployment pipelines. At a minimum, a platform automation team will need these skills to act on decisions made by the cloud center of excellence or central operations teams. Workload teams will be required to develop additional knowledge related to the controls and processes that will govern their decisions.
- **Landing zone design:** Landing zone design takes a dependency on the foundational utilities. To avoid duplication of effort (or errors/conflicts with automated governance), each workload focused team should understand what is included in the design and what is forbidden. Exception processes should also be factored in to landing zone designs to create flexibility.
- **Foundational utilities:** Centralization of foundational utilities does take some time to consider options and develop a solution that will scale to meet various adoption plans. This can delay early adoption efforts, but should be offset in the long term by accelerations and blocker avoidance later in the process.
- **Separation of duties:** Ensuring clear separation of duties does require mature identity management processes. There might be additional maintenance associated with the proper alignment of users, groups, and onboarding/off-boarding activities. New processes will likely be needed to accommodate just-in-time access via elevated privileges.

Distributed operations



The existing operating model might be too engrained for the entire organization to shift to a new operating model. For others, global operations and various compliance requirements might prevent specific business units from making a change. For those companies, a distributing operations approach might be required. This is by far the most complex approach, as it requires an integration of one or more of the previously mentioned operating models.

While heavily discouraged, this approach to operations might be required for some organizations who consist of a loose collection of disparate business units. Especially when those business units span a diverse base of customer segments or regional operations.

- **Priorities:** Integration of multiple existing operating models.
- Transitional state with a focus on moving the entire organization to one of the previously mentioned operating models, over time.
- Longer term operational approach when the organization is too large or too complex to align to a single operating model.
- **Distinct advantage:** Integration of common operating model elements from each business unit. Creates a vehicle to group operating units into a hierarchy and help them mature operations using consistent repeatable processes.
- **Distinct disadvantage:** Consistency and standardization across multiple operating models is difficult to maintain for extended periods. This operational approach requires deep awareness of the portfolio and how various segments of the technology portfolio are operated.
- **Risk:** Lack of commitment to a primary operating model could lead to confusion across teams. This operating model should only be used when there is no way to align to a single operating model.
- **Guidance:** Start with a thorough review of the portfolio using the approach outlined in the business alignment articles. Take care to group the portfolio by desired state operating model (decentralized, centralized, or enterprise).
- Develop a management group hierarchy that reflects the operating model groupings, followed by other organizational patterns for region, business unit, or other criteria that map the workload clusters from least common to most common buckets.
- Evaluate the alignment of workloads to operating models to find the most relevant operating model cluster to start with. Follow the guidance that maps to that operating model for all workloads under that node of the

management group hierarchy.

- Use the Govern and Manage methodologies to find common corporate policies and required operational management practices at various points of the hierarchy. Apply common Azure policies to automate the shared corporate policies.
- As those Azure policies are tested with various deployments, attempt to move them higher in the management group hierarchy applying those policies to greater numbers of workloads to find commonalities and distinct operation needs.
- Over time this approach will help define an operating model that scales across your various other operating models and unifies teams through a set of common policies and procedures.

Advantages and disadvantages of this approach are purposefully blank. After you complete the business alignment of your portfolio, see the predominant operating model section above for clarity on advantages and disadvantages.

Next steps

Learn the terminology associated with operating models. The terminology helps you understand how an operating model fits into the bigger theme of corporate planning.

Operating model terminology

Learn how a landing zone provides the basic building block of any cloud adoption environment.

Compare common cloud operating models

Operating model terminology

5/4/2021 • 2 minutes to read • Edit Online

The term operating model has many definitions. This intro article establishes terminology associated with operating models. To understand an operating model as it relates to the cloud, we first have to understand how an operating model fits into the bigger theme of corporate planning.

Terms

Business model: Business models tend to define corporate value (*what* the business does to provide value) and mission/vision statements (*why* the business has chosen to add value in that way). At a minimum, business models should be able to represent the *what* and *why* in the form of financial projections. There are many different schools of thought regarding how far a business model goes beyond these basic leadership principles. However, to create a sound operating model, the business models should include high-level statements to establish directional goals. It's even more effective if those goals can be represented in metrics or KPIs to track progress.

Customer experience: All good business models ground the *why* side of a business's strategy in the experience of their customers. This process could involve a customer acquiring a product or service. It could include interactions between a company and its business customers. Another example could center around the long-term management of a customer's financial or health needs, as opposed to a single transaction or process. Regardless of the type of experience, the majority of successful companies realize that they exist to operate and improve the experiences that drive their *why* statements.

Digital transformation: Digital transformation has become an industry buzzword. However, it is a vital component in the fulfillment of modern business models. Since the advent of the smartphone and other portable computing form factors, customer experiences have become increasingly digital. This shift is painfully obvious in some industries like DVD rentals, print media, automotive, or retail. In each case, digitized experiences have had a significant impact on the customer experience. In some cases, physical media have been entirely replaced with digital media, upsetting the entire industry vertical. In others, digital experiences are seen as a standard augmentation of the experience. To deliver business value (*what* statements), the customer experience (*why* statements) must factor in the impact of digital experiences on the customers' experiences. This process is digital transformation. Digital transformation is seldom the entire *why* statement in a business strategy, but it is an important aspect.

Operating model: If the business model represents the *what* and *why*, then an operating model represents the *how* and *who* for operationalizing the business strategy. The operating model defines the ways in which people work together to accomplish the large goals outlined in the business strategy. Operating models are often described as the people, process, and technology behind the business strategy. In the article on the Cloud Adoption Framework operating model, this concept is explained in detail.

Cloud adoption: As stated above, digital transformation is an important aspect of the customer experience and the business model. Likewise, cloud adoption is an important aspect of any operating model. Cloud adoption is a strong enabler to deliver the right technologies and processes required to successfully deliver on the modern operating model.

Cloud adoption is *what we do* to realize the business value. The operating model represents *who we are* and how we function on a daily basis while cloud adoption is being delivered.

Take action

Use the operating model provided by the Cloud Adoption Framework to develop operational maturity.

Next steps

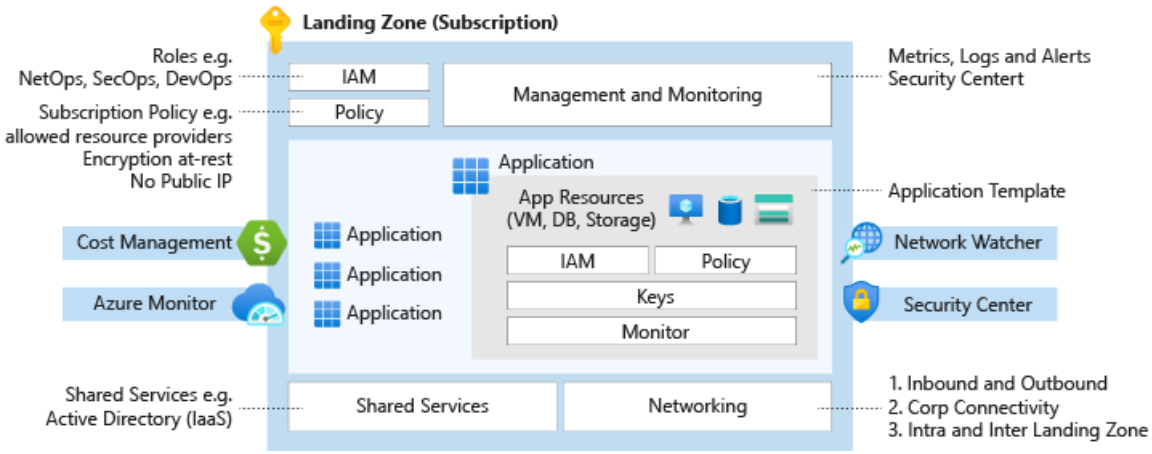
Continue to the next section of the Cloud Adoption Framework. Learn how a landing zone provides the basic building block of any cloud adoption environment.

Use the operating model

What is an Azure landing zone?

8/25/2021 • 2 minutes to read • Edit Online

Azure landing zones are the output of a multisubscription Azure environment that accounts for scale, security governance, networking, and identity. Azure landing zones enable application migration, modernization, and innovation at enterprise-scale in Azure. These zones consider all platform resources that are required to support the customer's application portfolio and don't differentiate between infrastructure as a service or platform as a service.



A landing zone is an environment for hosting your workloads, preprovisioned through code. Watch the following video to learn more.

Scalable and modular

No single solution fits all technical environments. A few Azure landing zone implementation options can help you meet the deployment and operations needs of your growing cloud portfolio.

- **Scalable:** All Azure landing zones support cloud adoption at scale by providing repeatable environments, with consistent configuration and controls, regardless of the workloads or Azure resources deployed to each landing zone instance.
- **Modular:** All Azure landing zones provide a modular approach to building out your environment, based on a common set of design areas. Each design area can be easily extended to support the distinct needs of various technology platforms like Azure SQL Database, Azure Kubernetes Service, and Azure Virtual Desktop.

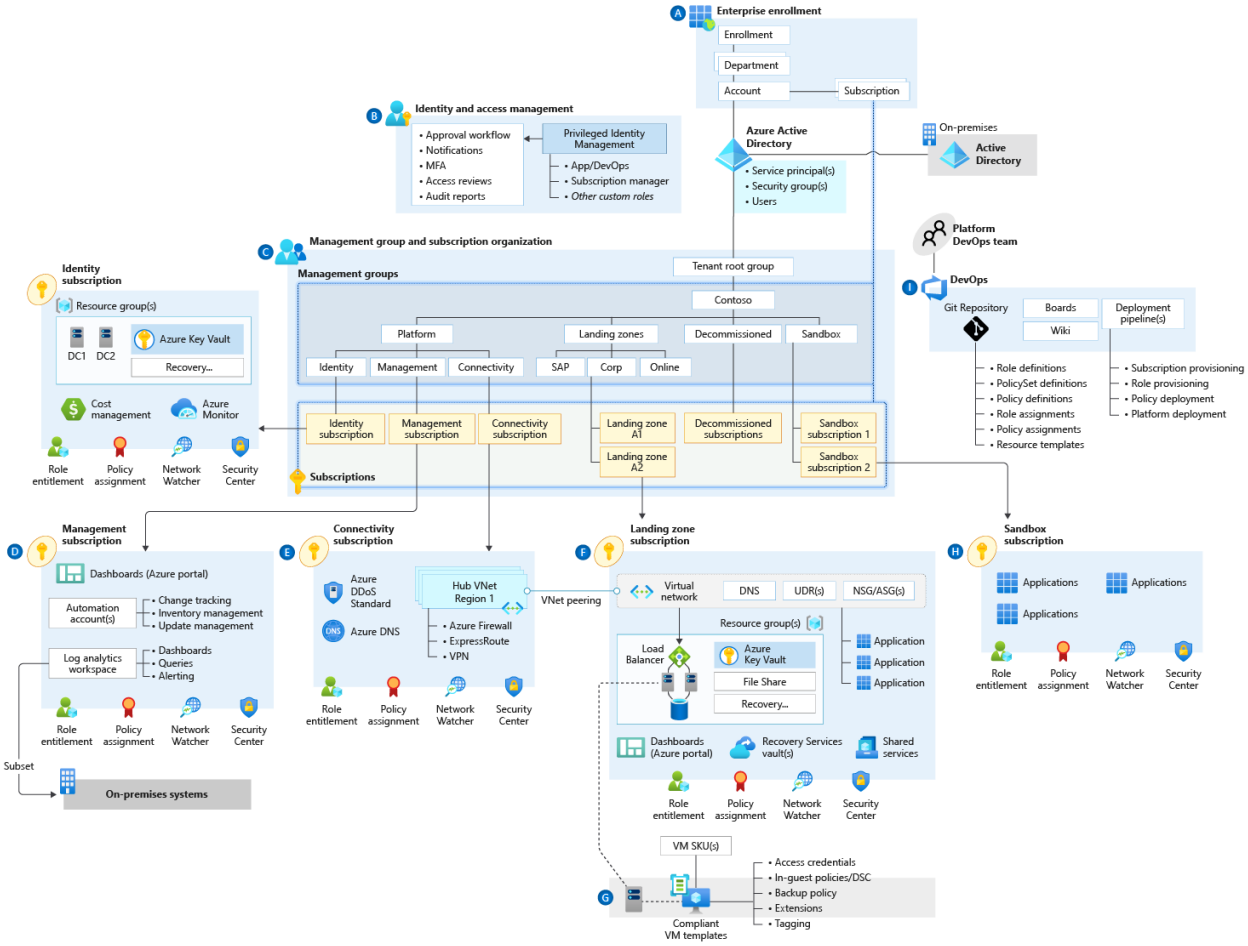
Whether you're looking to deploy your first production application to Azure or you're operating a complex portfolio of tech platforms and workloads, the Azure landing zone implementation options can be tailored to your needs.

Azure landing zone conceptual architecture

For many organizations, the Azure landing zone conceptual architecture below represents the destination in their cloud adoption journey. It's a mature, scaled-out target architecture intended to help organizations operate successful cloud environments that drive their business while maintaining best practices for security and governance.

This conceptual architecture represents scale and maturity decisions based on a wealth of lessons learned and feedback from customers who have adopted Azure as part of their digital estate.

While your specific implementation might vary, as a result of specific business decisions or existing investments in tools that need to persist in your cloud environment, this conceptual architecture will help set a direction for the overall approach your organization takes to designing and implementing a landing zone.



Azure landing zone accelerator

For organizations where this conceptual architecture fits with the operating model and resource structure they plan to use, there's a ready-made deployment experience called the **Azure landing zone accelerator**.

The accelerator is an Azure-portal-based deployment that will provide a full implementation of the conceptual architecture, along with opinionated configurations for key components such as management groups and policies.

Deploy to Azure

There are other deployment options available, some that deliver the full architecture using third-party deployment technologies, and others that start from a smaller footprint. For more information, see Implementation options.

Next steps

Organizations may be at different stages of their cloud journey when reviewing this guidance. As a result, the actions and recommendations required to progress toward the outcome detailed above may vary. To understand best next actions in relation to where you are in your cloud adoption, review the journey to the target architecture content.

Journey to the target architecture

When you're choosing the right Azure landing zone implementation option, you should understand the Azure landing zone design areas.

Review design areas

Journey toward the target architecture

7/22/2021 • 2 minutes to read • Edit Online

Adopting cloud technologies is a journey. Business priorities and the need to bring new technologies online in order to unlock capabilities or features will influence the speed at which an organization deploys and scales out a cloud environment.

Over time, the organization iterates and matures the deployed technologies, processes, and skilling needed to progress toward that destination. However, this doesn't mean you arrive at the destination immediately. The journey takes time, which varies depending on the size of the organization, the current technical footprint, and the skilling maturity within technical teams.

On-ramps

Consider an analogy of a trip along a freeway. There might be multiple *on-ramps* you can use to join the freeway, but the *destination* is the same.



These on-ramps represent where organizations are today in their cloud adoption plans and the specific guidance needed to continue to develop the cloud environment.

ON-RAMP	DESCRIPTION	FURTHER GUIDANCE
Start	For organizations who are at the beginning of their cloud adoption journey (also referred to as *greenfield*) who are looking to implement a new cloud environment, based on best practices and proven architectural patterns. Start with the Azure landing zone conceptual architecture to understand the recommended end state. Next, explore each of the design areas to understand the key themes that comprise the considerations and decisions needed in order to design and implement the landing zone that best fits your requirements	What is an Azure landing zone Azure landing zone design areas

ON-RAMP	DESCRIPTION	FURTHER GUIDANCE
Align	For organizations who have an existing environment that needs modification to align to the Azure landing zone target architecture and best practices (also referred to as *brownfield*). Use the transition from brownfield guidance in order to understand the decision points and technical approach to refactoring environments to align with the guidance in the Ready methodology	Refactor a landing zone
Enhance	For environments that are already in line with best practices but the organization is looking to add additional controls or features. Explore the articles that explain the considerations as part of enhancing the key ongoing processes for cloud environments, such as management, governance and security.	Enhance guidance for management Enhance guidance for governance Enhance guidance for security

Next steps

To understand the decisions you need to make when implementing an Azure landing zone, explore the Azure landing zone design areas for specific technical topics to consider.

Azure landing zone design areas

Azure landing zone design areas

7/22/2021 • 2 minutes to read • Edit Online

Each Azure landing zone implementation option provides a deployment approach and defined design principles. Before choosing an implementation option, use this article to gain an understanding of the design areas listed in the following table.

> **NOTE**
> These design areas describe what you should consider prior to deploying a landing zone. Use it as a simple reference. Explore each of the design areas in this section for further guidance on each of the key themes for landing zone design.

Environment design areas

Regardless of the deployment option, you should carefully consider each design area. Your decisions affect the platform foundation on which each landing zone depends.

DESIGN AREAS	OBJECTIVE	RELEVANT METHODOLOGIES
Azure billing and Active Directory tenant	Proper tenant creation, enrollment, and billing setup are important early steps.	Ready
Identity and access management	Identity and access management is a primary security boundary in the public cloud. It's the foundation for any secure and fully compliant architecture.	Ready
Network topology and connectivity	Networking and connectivity decisions are an equally important foundational aspect of any cloud architecture.	Ready
Resource organization	As cloud adoption scales, considerations for subscription design and management group hierarchy have an impact on governance, operations management, and adoption patterns.	Govern

Compliance design areas

Security, governance, and compliance are key topics when designing and building an Azure environment, not only to start from strong foundations, but also to ensure solid ongoing processes and controls are in place. Alongside the controls that help maintain and demonstrate compliance, the tools and processes implemented for managing environments play an important role in detecting and responding to issues that require action.

DESIGN AREAS	OBJECTIVE	RELEVANT METHODOLOGIES
Governance	Automate auditing and enforcement of governance policies.	Govern

DESIGN AREAS	OBJECTIVE	RELEVANT METHODOLOGIES
Security	Implement controls and process to protect your cloud environments	Secure
Management	For stable, ongoing operations in the cloud, a management baseline is required to provide visibility, operations compliance, and protect and recover capabilities.	Manage
Platform automation and DevOps	Align the best tools and templates to deploy your landing zones and supporting resources.	Ready

Next steps

You can implement these design areas over time so that you can grow into your cloud operating model. Review the methodologies related to each of the design areas to understand in more detail the considerations and decisions required to implement a landing zone.

Alternately, there are rich, opinionated implementation options that start with a defined position on each design area.

With an understanding of the modular design areas, the next step is to choose the landing zone implementation option that best aligns with your cloud adoption plan and requirements.

Choose an implementation option

Choose the landing zone for your organization

8/9/2021 • 6 minutes to read • Edit Online

There are different approaches to implementing landing zones in the Cloud Adoption Framework. The right approach for your organization has the necessary services to support your business applications without extra overhead to manage. Starting with an implementation that doesn't meet your needs can waste your time and effort.

Microsoft offers two implementation options for landing zones:

- Start small and expand
- Enterprise-scale

Watch the following 15-minute video to learn more about how to choose the Azure landing zone implementation options that best fit your needs.

You might also consider third-party implementations. Our partners have many implementations available through their services. For more information, see Evaluate a Microsoft partner's Azure landing zone.

Overview of landing zone options

The following table summarizes considerations for each landing zone implementation approach.

Start small and expand

Enterprise-scale

Initial considerations

Operating model alignment

Centralized operations

Enterprise operations

Baseline reference architecture

Offers a simple starting point to build your own solution with minimal subscriptions, which you scale only as needed.

Offers an entire Azure tenant reference regardless of your scale-point, which includes cloud-native operations.

Adoption plan considerations

Provides long-term self-sufficiency

Requires Cloud Adoption Framework Govern and Manage methodologies to achieve long-term self-sufficiency.

Enterprise-scale architecture landing zones approach and architecture prepare your organization for long-term self-sufficiency. Provides reserved instances to get started.

Enables adoption velocity across the organization

Quickly enable low-risk adoption. Build toward security governance and compliance over time.

Start with security governance and compliance to enable compliant adoption sooner.

Achieves operational excellence

Requires Cloud Adoption Framework Govern and Manage methodologies to achieve operational excellence.

Enables operational excellence with autonomy for platform and application teams built on policy-driven governance and management.

Compliance considerations

Path to attain security governance and compliance

Iterative approach. Requires Govern and Manage methodologies to support sensitive data or mission critical workloads.

Enterprise-scale architecture includes designs for governance, security segmentation, and separation of duties. This approach empowers teams to act within appropriate landing zones.

Associated risks while building out security governance and compliance

There is a risk of extensive refactoring or even redeployment to attain required needs.

There is a risk of enabling cloud-native operations products that may not align to your operating model.

Deployment considerations

Best practices from cloud provider?

More best practices are required to be added using Cloud Adoption Framework methodologies to apply security governance and compliance.

Enterprise-scale includes Azure best practices and is the target technical state for your Azure environment.

All critical services are present and properly configured according to recommended best practices for identity/access management, governance, security, network, and logging.

Partial. Some resources are deployed. Additional offerings aligned to Cloud Adoption Framework methodologies required to apply best practices to support security governance and compliance.

Enterprise-scale architecture is the target technical state recommendation for your Azure environment that aligns with the Azure platform roadmap.

Automation capabilities like infrastructure as code (IaC) and Azure DevOps

Azure Resource Manager, Azure Policy, and Azure Blueprints can be used to create your own continuous integration/continuous development pipeline.

Azure Resource Manager, Azure Policy, and GitHub/Azure DevOps. CI/CD pipeline options are included in the reference implementation guidance.

Timeline considerations

Timeline to adopt or migrate a low-risk workload:

3 to 10 days

3 to 10 days

Timeline to achieve security governance and compliance requirements for all workloads:

Four to six months

Six to eight weeks

Initial considerations

Which operating model better describes your organization? Be aware of not just how your organization looks now but what you expect and want it to be in three months to a year or further.

- Centralized operations: In this small environment, centralized teams for IT operations, security, and other roles manage production and workloads.
- Enterprise operations: In this typically larger or industry specialized environments, enterprise operations have a stable, steady state that's managed centrally.

Centralized operations favor a start-small-and-expand approach. Enterprise operations might be better addressed with enterprise-scale.

Do you need a baseline architecture or environment? The start-small-and-expand approach offers a simple starting point where you can build your own solution. The enterprise-scale approach provides an environment for the entire Azure tenant, which includes cloud-native operations.

For more information about operations types, see Compare common cloud operating models.

Adoption plan considerations

The following considerations are key to your adoption plan for either type:

- Long-term self-sufficiency
- Adoption velocity across the organization
- Operational excellence

Enterprise-scale provides long-term self-sufficiency and operational excellence immediately. Enterprise-scale also helps to accelerate compliance adoption across your organization. The enterprise-scale approach builds out a foundation for you. Enterprise-scale includes the guardrails around security, identity, and network. The approach includes CI/CD pipeline options for DevOps and automation.

If you start small and expand, there are ways to achieve self-sufficiency, adoption velocity, and operational excellence. Use the Govern and Manage methodologies of the Cloud Adoption Framework to iteratively build those pieces into the landing zone solution. Use the eight areas of design, Cloud Adoption Framework enterprise-scale design guidelines, to iteratively improve your design.

To learn more about operational excellence, see Deliver operational excellence during digital transformation.

Compliance considerations

Consider the following issues around compliance for your organization:

- Path to reach security governance and compliance
- Risks while building out security governance and compliance

Your organization might need to have a particular workload or application that needs to be compliant in a short amount of time, and this requirement could affect your choice.

Start small and expand is an iterative approach to compliance. Use the Cloud Adoption Framework Govern and Manage methodologies to support sensitive data or critical workloads. For more information, see the Govern methodology for the cloud and IT management and operations in the cloud.

The enterprise-scale architecture includes designs for segmentation and separation to support compliance goals and a service enablement framework to determine how to achieve appropriate levels of governance, security, and compliance.

If possible, identify low-risk workloads to implement first. This technique helps you build infrastructure and skills over time. You can add the Govern and Manage methodologies as you understand how the cloud works.

Deployment considerations

Deploying your landing zone or landing zones raises several considerations for choosing an implementation:

- Best practices from cloud provider.
- All critical services are present and properly configured according to recommended best practices for identity/access management, governance, security, network, and logging.
- Automation capabilities like IaC and Azure DevOps.

Both implementation offer best practices. Start small and expand lets you add best practices using Cloud Adoption Framework methodologies to apply security governance and compliance.

Enterprise-scale comes with all critical services configured. Start small and expand comes with some resources deployed.

For more information, see Best practices for Azure readiness.

Both methodologies offer automation capabilities.

- **Start small and expand:** ARM templates, Azure Policy, and Azure Blueprints. Can create your own CI/CD pipeline.
- **Enterprise-scale:** ARM templates, Azure Policy, GitHub/Azure DevOps, and CI/CD pipeline options are included in the reference implementation guidance.

Start small and expand uses ARM templates, Azure Policy, and Azure Blueprints.

- CAF Foundation blueprint
- CAF Migration landing zone blueprint

Enterprise-scale uses ARM templates, Azure Policy, and offers three reference implementation and different deployments.

- Enterprise-scale foundation
- Enterprise-scale Virtual WAN
- Enterprise-scale hub and spoke

Whether you implement start small and expand or enterprise-scale, you can use templates and a portal-based experience. You can include IaC later in the process. Explore this IaC overview for more information.

Timeline considerations

Landing zone options take different amounts of time to implement. There are two types of timelines:

- A timeline to adopt or migrate a low-risk workload.
- A timeline to reach security governance and compliance requirements for all workloads.

With a start-small-and-expand approach, you can have a low-risk workload up and running in 3 to 10 days. For workloads with high security governance and compliance requirements, it could take four to six months.

For an enterprise-scale implementation, you can still adopt a low-risk workload in 3 to 10 days, but more elaborate workloads can be ready in six to eight weeks.

Next steps

Start small and expand

Start with enterprise-scale

Landing zone implementation options

8/25/2021 • 3 minutes to read • Edit Online

Azure landing zones provide cloud adoption teams with a well-managed environment for their workloads. Follow the landing zone design areas guidance to take advantage of these capabilities.

Each of the following implementation options is designed for a specific set of operating model dependencies to support your nonfunctional requirements. Each implementation option includes distinct automation approaches. When available, reference architectures and reference implementations are included to accelerate your cloud adoption journey. While each option is mapped to a different operating model, they share the same design areas. The difference is how you choose to implement them.

Platform development velocity

In addition to the recommended design areas, your platform development velocity (how fast your platform team can develop the required skills) is a key factor when choosing the best implementation option. Consider two primary modes:

Start with enterprise scale: Use this mode if your business requirements necessitate a rich initial implementation of landing zones with fully integrated governance, security, and operations from the start. With this approach, you can use either the Azure portal or infrastructure-as-code to set up and configure your environment. You can also transition between the Azure portal and infrastructure as code (recommended) when your organization is ready. To use the recommended infrastructure-as-code approach, your organization will require skills in Azure Resource Manager templates and GitHub.

Start small and expand: Use this mode if it's more important to develop these skills and commit to your decisions as you learn more about the cloud. In this approach, the landing zones only focus on implementing the basic landing zones considerations required to start cloud adoption. As your adoption expands, modules in the Govern and Manage methodologies will build on top of your initial landing zones. The design principles of any Azure landing zone outline the specific design areas that will require refactoring over time.

Implementation options

The following table describes some of the implementation options for landing zones and the variables that might drive your decision.

IMPLEMENTATION OPTION	DESCRIPTION	DEPLOYMENT VELOCITY	DEEPER DESIGN PRINCIPLES	DEPLOYMENT INSTRUCTIONS
CAF Migration landing zone blueprint	Deploys the basic foundation for migrating low risk assets.	Start small	Design principles	Deploy
CAF Foundation blueprint	Adds the minimum tools need to begin developing a governance strategy.	Start small	Design principles	Deploy

IMPLEMENTATION OPTION	DESCRIPTION	DEPLOYMENT VELOCITY	DEEPER DESIGN PRINCIPLES	DEPLOYMENT INSTRUCTIONS
CAF enterprise-scale landing zone (hybrid connectivity with Virtual WAN)	Deploys an enterprise-ready platform foundation with all the necessary shared services to support the full IT portfolio, including hybrid connectivity (Virtual WAN).	Enterprise-scale	Design principles	Deploy
CAF enterprise-scale landing zone (hybrid connectivity with hub and spoke)	Deploys an enterprise-ready platform foundation with all the necessary shared services to support the full IT portfolio, including hybrid connectivity (hub and spoke).	Enterprise-scale	Design principles	Deploy
CAF enterprise-scale landing zone (foundation)	Deploys an enterprise-ready platform foundation with all the necessary shared services to support the full IT portfolio, where connectivity can be added later as needed.	Enterprise-scale	Design principles	Deploy
CAF enterprise-scale landing zone (small and midsize enterprise)	This reference implementation is meant for organizations that do not have a large IT team and do not require fine grained administration delegation models.	Enterprise-scale	Design principles	Deploy

IMPLEMENTATION OPTION	DESCRIPTION	DEPLOYMENT VELOCITY	DEEPER DESIGN PRINCIPLES	DEPLOYMENT INSTRUCTIONS
Terraform module for Cloud Adoption Framework enterprise-scale	Deploys an enterprise-ready platform foundation using Terraform. Use this option when managing your platform using Terraform and need to accelerate delivery of the recommended resource hierarchy and governance model. This module is officially verified on the Terraform registry. Shared services, network connectivity, and application workloads can be integrated into your deployment or managed independently.	Enterprise-scale	Design principles	Deploy
CAF Terraform modules	Third-party path for multicloud operating models. This path can limit Azure-first operating models.	Start small	Design principles	Deploy
Partner landing zones	Partners who provide offerings aligned to the Ready methodology of the Cloud Adoption Framework can provide their own customized implementation option.	Variable	Design principles	Find a partner

The following table looks at some of these implementation options from a slightly different perspective to guide more technical decision processes.

IMPLEMENTATION OPTION	HUB	SPOKE	DEPLOYMENT TECHNOLOGY	DEPLOYMENT INSTRUCTIONS
Cloud Adoption Framework enterprise-scale landing zone	Included	Included	Azure Resource Manager templates, Azure portal, Azure Policy and GitHub	Deploy
CAF Migration landing zone blueprint	Refactor required	Included	Azure Resource Manager templates, Azure portal, and Azure Blueprints	Deploy

IMPLEMENTATION OPTION	HUB	SPOKE	DEPLOYMENT TECHNOLOGY	DEPLOYMENT INSTRUCTIONS
CAF Terraform modules	Included in virtual datacenter module	Included	Terraform	Deploy
Terraform module for Cloud Adoption Framework enterprise-scale	Excluded	Excluded	Terraform	Deploy

Next steps

To proceed, choose one of the implementation options shown in the preceding tables. Each option includes a link to deployment instructions and the specific design principles that guide implementation.

Start with Cloud Adoption Framework enterprise-scale landing zones

8/25/2021 • 2 minutes to read • Edit Online

The enterprise-scale architecture represents the strategic design path and target technical state for your Azure environment. It will continue to evolve alongside the Azure platform and is defined by the various design decisions that your organization must make to map your Azure journey.

Not all enterprises adopt Azure in the same way, so the Cloud Adoption Framework for Azure enterprise-scale landing zone architecture varies between customers. The technical considerations and design recommendations of the enterprise-scale architecture might lead to different trade-offs based on your organization's scenario. Some variation is expected, but if you follow the core recommendations, the resulting target architecture will set your organization on a path to sustainable scale.

Prescriptive guidance

The enterprise-scale architecture provides prescriptive guidance coupled with best practices for your Azure control plane. It follows design principles across the critical design areas for an organization's Azure environment.

Qualifiers: Should I start with enterprise-scale?

The enterprise-scale architecture is modular by design. It allows you to start with a foundational landing zone control plane that support your application portfolios, no matter whether the applications are being migrated or are newly developed and deployed to Azure. The architecture can scale alongside your business requirements regardless of scale point.

Start with a Cloud Adoption Framework enterprise-scale landing zone

The enterprise-scale approach to construct landing zones includes three sets of assets to support cloud teams:

- Design guidelines: Guide to the critical decisions that drive the design of the Cloud Adoption Framework for Azure enterprise-scale landing zone.
- Architecture: Conceptual reference architecture that demonstrates design areas and best practices.
- Implementations: Azure Resource Manager template of the architecture to accelerate adoption.

Community

This guide is developed largely by Microsoft architects and the broader Cloud Solutions Unit technical community. This community actively advances this guide to share lessons learned during enterprise-scale adoption efforts.

This guide shares the same design principles as the standard Ready methodology. It expands on those principles to integrate subjects such as governance and security earlier in the planning process. Expanding the standard process is necessary because of a few natural assumptions that can be made when an adoption effort requires large-scale enterprise change.

Next steps

Implement a Cloud Adoption Framework enterprise-scale landing zone

Cloud Adoption Framework enterprise-scale landing zone architecture

8/25/2021 • 5 minutes to read • Edit Online

Enterprise-scale is an architectural approach and a reference implementation that enables effective construction and operationalization of landing zones on Azure, at scale. This approach aligns with the Azure roadmap and the Cloud Adoption Framework for Azure.

Architecture overview

The Cloud Adoption Framework enterprise-scale landing zone architecture represents the strategic design path and target technical state for an organization's Azure environment. It will continue to evolve alongside the Azure platform and is defined by the various design decisions that your organization must make to map your Azure journey.

Not all enterprises adopt Azure the same way, so the Cloud Adoption Framework enterprise-scale landing zone architecture varies between customers. The technical considerations and design recommendations in this guide might yield different trade-offs based on your organization's scenario. Some variation is expected, but if you follow the core recommendations, the resulting target architecture will set your organization on a path to sustainable scale.

Landing zone in enterprise-scale

Azure landing zones are the output of a multisubscription Azure environment that accounts for scale, security, governance, networking, and identity. Azure landing zones enable application migrations and greenfield development at enterprise-scale in Azure. These zones consider all platform resources that are required to support the customer's application portfolio and don't differentiate between infrastructure as a service or platform as a service.

An example is how city utilities such as water, gas, and electricity are accessible before new homes are constructed. In this context, the network, identity and access management, policies, management, and monitoring are shared utility services that must be readily available to help streamline the application migration process before it begins.

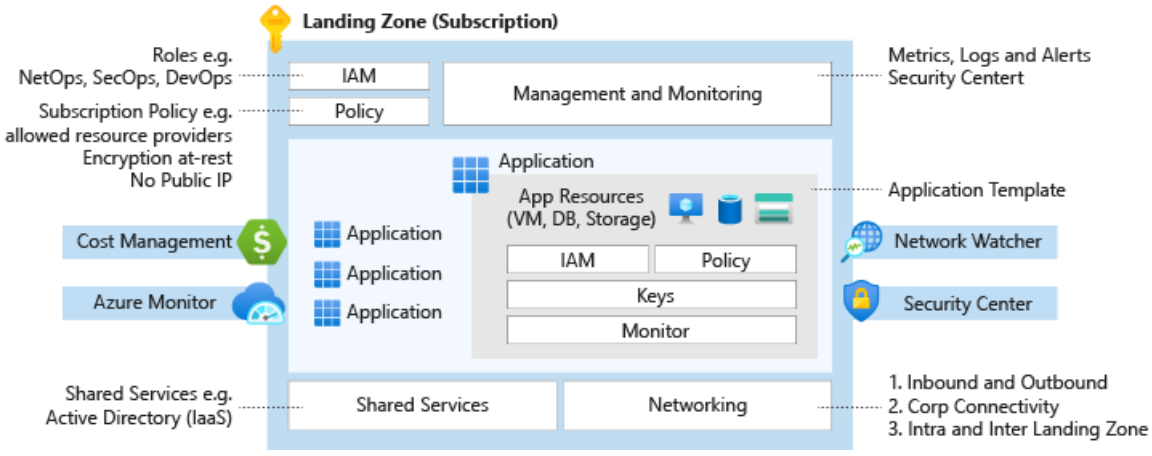


Figure 1: Landing zone design.

High-level architecture

An enterprise-scale architecture is defined by a set of design considerations and recommendations across eight critical design areas with two recommended network topologies: an enterprise-scale architecture based on an Azure Virtual WAN network topology (Figure 2) or based on a traditional Azure network topology that's on the hub-and-spoke architecture (Figure 3).

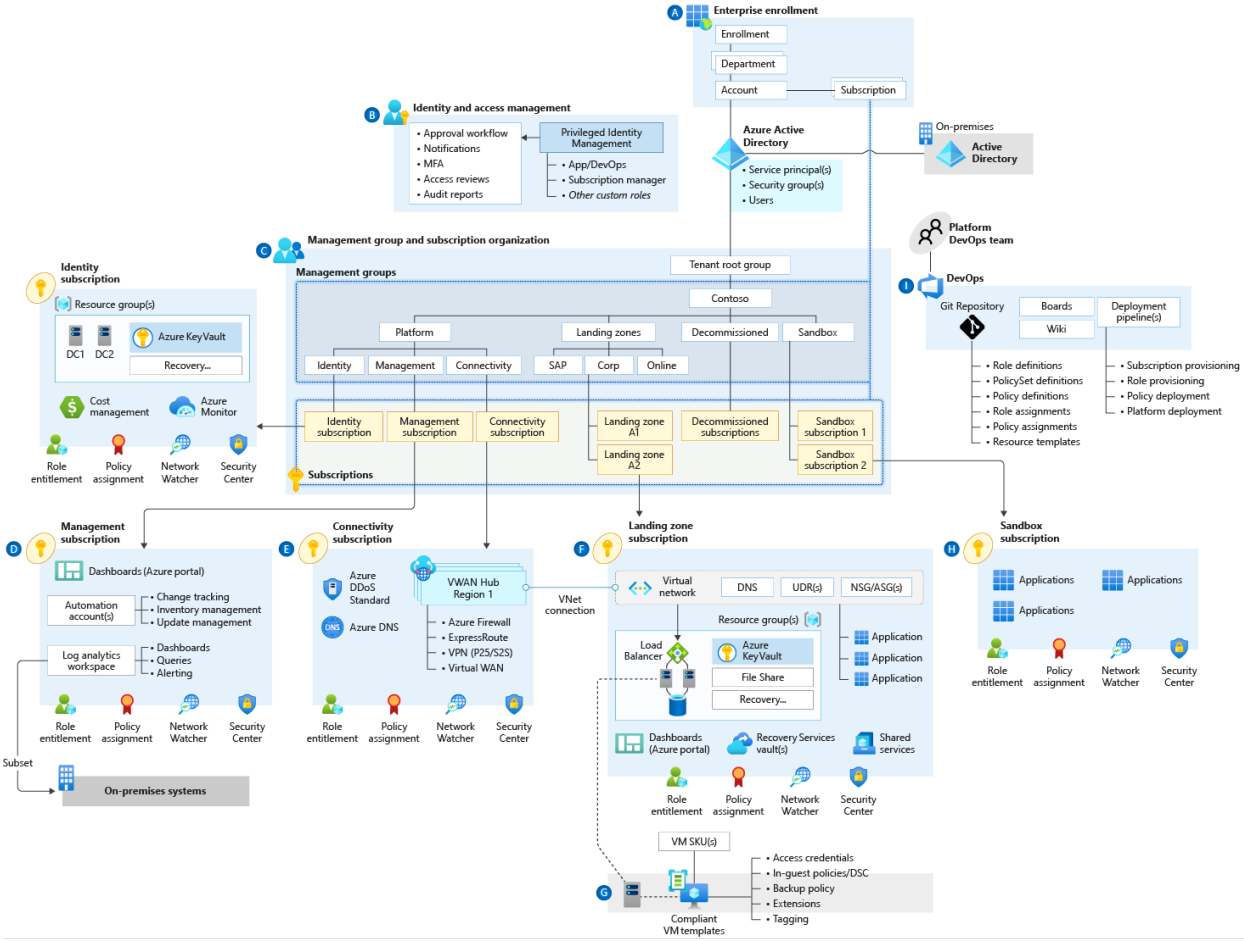


Figure 2: Cloud Adoption Framework enterprise-scale landing zone architecture based on an Azure Virtual WAN network topology. The connectivity subscription uses a Virtual WAN hub.

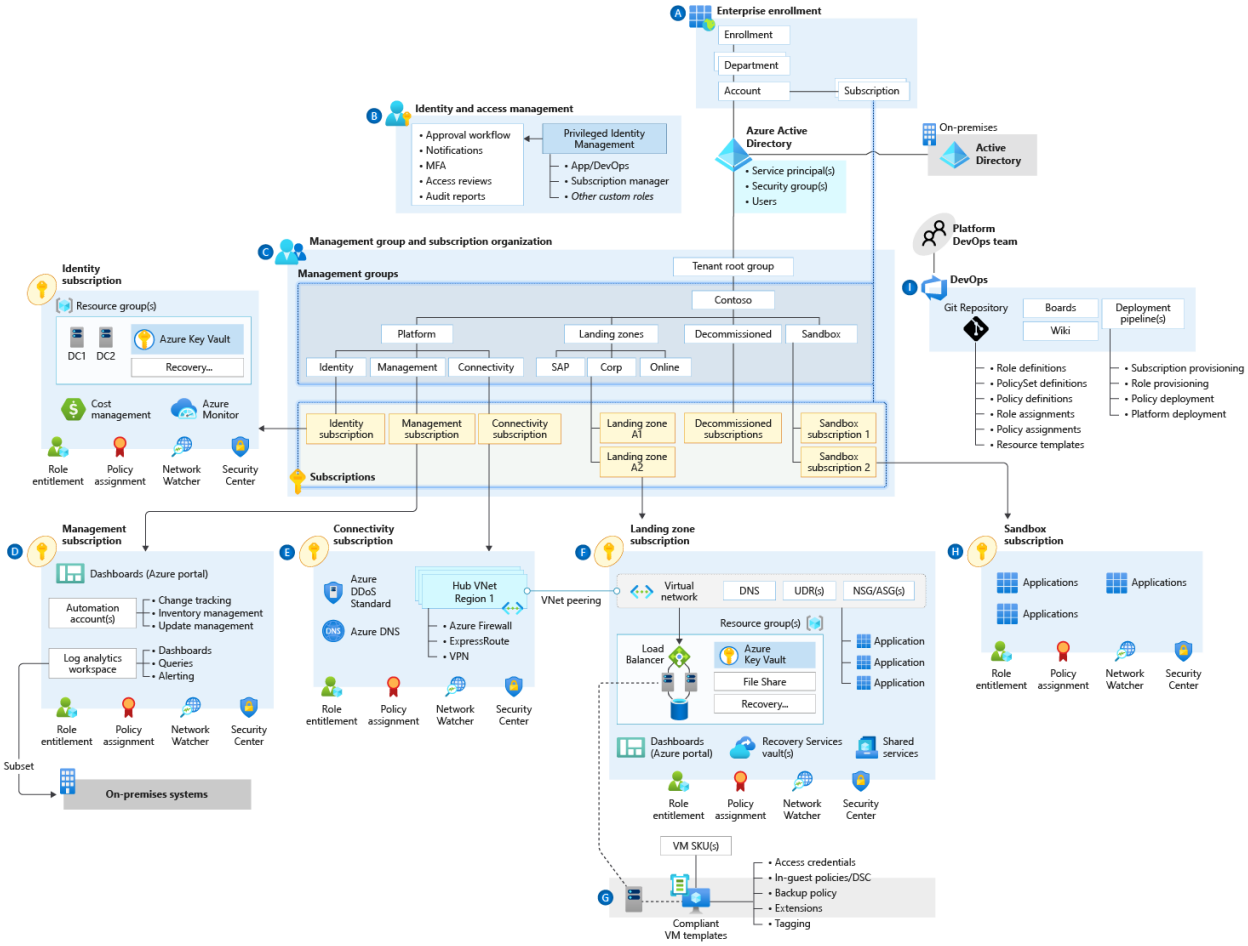


Figure 3: Cloud Adoption Framework enterprise-scale landing zone architecture based on a traditional Azure networking topology. The connectivity subscription uses a hub VNet.

Download the PDF or Visio files that contain the enterprise-scale architecture diagrams based on the Virtual WAN (PDF) network topology or a traditional Azure network topology based on the hub-and-spoke (PDF) architecture. A Visio file containing both the Virtual WAN and the hub-and-spoke architecture diagram can be downloaded as a Visio diagram (VSDX).

> **IMPORTANT**
>
> It's possible to implement an enterprise-scale landing zone architecture with a single subscription for **platform resources** if the team managing platform resources isn't concerned with separating duties. With scalability and security in mind, it's strongly recommended to have dedicated subscriptions for platform resources (Figures 2 and 3).

Figures 2 and 3 reference the enterprise-scale critical design areas, which are outlined by the letters A through I:

A Enterprise Agreement (EA) enrollment and Azure Active Directory (Azure AD) tenants. An Enterprise Agreement (EA) enrollment represents the commercial relationship between Microsoft and how your organization uses Azure. It provides the basis for billing across all your subscriptions and affects administration of your digital estate. Your EA enrollment is managed via the Azure EA portal. An enrollment often represents an organization's hierarchy, which includes departments, accounts, and subscriptions. An Azure AD tenant provides identity and access management, which is an important part of your security posture. An Azure AD tenant ensures that authenticated and authorized users have access to only the resources for which they have access permissions.

B Identity and access management. Azure AD design and integration must be built to ensure both server and user authentication. Azure role-based access control (Azure RBAC) must be modeled and deployed to enforce separation of duties and the required entitlements for platform operation and management. Key management must be designed and deployed to ensure secure access to resources and support operations such as rotation

and recovery. Ultimately, access roles are assigned to application owners at the control and data planes to create and manage resources autonomously.

C Management group and subscription organization. Management group structures within an Azure AD tenant support organizational mapping and must be considered thoroughly when an organization plans Azure adoption at scale. Subscriptions are a unit of management, billing, and scale within Azure. They play a critical role when you're designing for large-scale Azure adoption. This critical design area helps you capture subscription requirements and design target subscriptions based on critical factors. These factors are environment type, ownership and governance model, organizational structure, and application portfolios.

D Management and monitoring. Platform-level holistic (horizontal) resource monitoring and alerting must be designed, deployed, and integrated. Operational tasks such as patching and backup must also be defined and streamlined. Security operations, monitoring, and logging must be designed and integrated with both resources on Azure and existing on-premises systems. All subscription activity logs that capture control plane operations across resources should be streamed into Log Analytics to make them available for query and analysis, subject to Azure RBAC permissions.

E Network topology and connectivity. The end-to-end network topology must be built and deployed across Azure regions and on-premises environments to ensure north-south and east-west connectivity between platform deployments. Required services and resources such as firewalls and network virtual appliances must be identified, deployed, and configured throughout network security design to ensure that security requirements are fully met.

F, G, H Business continuity and disaster recovery and Security governance and compliance. Holistic and landing-zone-specific policies must be identified, described, built, and deployed onto the target Azure platform to ensure corporate, regulatory, and line-of-business controls are in place. Ultimately, policies should be used to guarantee the compliance of applications and underlying resources without any abstraction provisioning or administration capability.

I Platform automation and DevOps. An end-to-end DevOps experience with robust software development life cycle practices must be designed, built, and deployed to ensure a safe, repeatable, and consistent delivery of infrastructure-as-code artifacts. Such artifacts are to be developed, tested, and deployed by using dedicated integration, release, and deployment pipelines with strong source control and traceability.

Next steps

Customize implementation of this architecture by using the Cloud Adoption Framework enterprise-scale design guidelines.

Design guidelines

Cloud Adoption Framework enterprise-scale design principles

8/25/2021 • 2 minutes to read • Edit Online

The enterprise-scale architecture prescribed in this guidance is based on the design principles described here. These principles serve as a compass for subsequent design decisions across critical technical domains. Familiarize yourself with these principles to better understand their impact and the trade-offs associated with nonadherence.

Subscription democratization

Subscriptions should be used as a unit of management and scale aligned with business needs and priorities to support business areas and portfolio owners to accelerate application migrations and new application development. Subscriptions should be provided to business units to support the design, development, and testing of new workloads and migration of workloads.

Policy-driven governance

Azure Policy should be used to provide guardrails and ensure continued compliance with your organization's platform, along with the applications deployed onto it. Azure Policy also provides application owners with sufficient freedom and a secure unhindered path to the cloud.

Single control and management plane

Enterprise-scale architecture shouldn't consider any abstraction layers, such as customer-developed portals or tooling. It should provide a consistent experience for both AppOps (centrally managed operation teams) and DevOps (dedicated application operation teams). Azure provides a unified and consistent control plane across all Azure resources and provisioning channels subject to role-based access and policy-driven controls. Azure can be used to establish a standardized set of policies and controls for governing the entire enterprise estate.

Application-centric and archetype-neutral

Enterprise-scale architecture should focus on application-centric migrations and development rather than pure infrastructure lift-and-shift migrations, such as moving virtual machines. It shouldn't differentiate between old and new applications, infrastructure as a service, or platform as a service applications. Ultimately, it should provide a safe and secure foundation for all application types to be deployed onto your Azure platform.

Align Azure-native design and roadmaps

The enterprise-scale architecture approach advocates using Azure-native platform services and capabilities whenever possible. This approach should align with Azure platform roadmaps to ensure that new capabilities are available within your environments. Azure platform roadmaps should help to inform the migration strategy and enterprise-scale trajectory.

Recommendations

Be prepared to trade off functionality because it's unlikely that everything will be required on day one. Use preview services and take dependencies on service roadmaps to remove technical blockers.

Cloud Adoption Framework enterprise-scale design guidelines

8/25/2021 • 2 minutes to read • Edit Online

This article and the articles series that follows outline how the enterprise-scale architecture provides an opinionated position on each of the Azure landing zone design areas. This series provides a step-by-step set of design guidelines that can be followed to implement the design principles embodied in the enterprise-scale solution.

The core of enterprise-scale architecture contains a critical design path comprised of fundamental design topics with heavily interrelated and dependent design decisions. This repo provides design guidance across these architecturally significant technical domains to support the critical design decisions that must occur to define the enterprise-scale architecture. For each of the considered domains, review the provided considerations and recommendations and use them to structure and drive designs within each area.

For example, you might ask how many subscriptions are needed for your estate. You can review subscription organization and governance and use the outlined recommendations to drive subscription decisions.

Critical design areas

The following eight critical design areas can help you translate your requirements to Azure constructs and capabilities. These design areas can help you address the mismatch between on-premises and cloud-design infrastructure, which typically creates dissonance and friction between the enterprise-scale definition and Azure adoption.

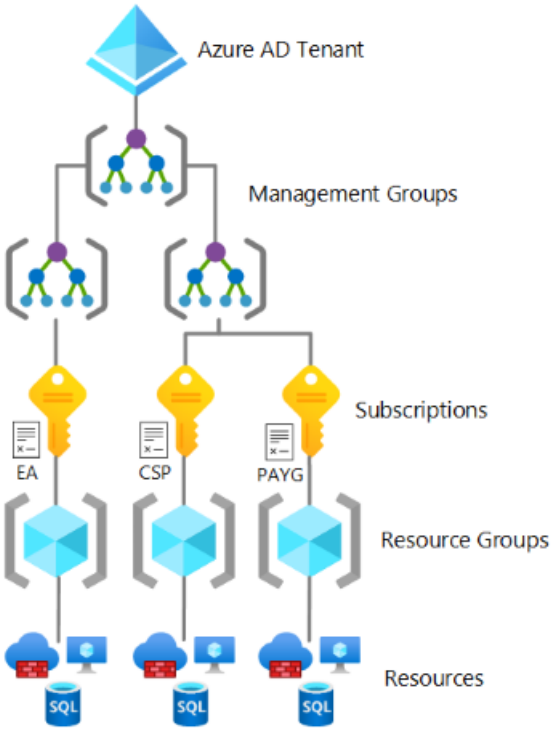
The impact of decisions made within these critical areas will reverberate across enterprise-scale architecture and influence other decisions. Familiarize yourself with these eight areas to better understand the consequences of encompassed decisions, which could later produce trade-offs within related areas.

- Enterprise Agreement (EA) enrollment and Azure Active Directory tenants
- Identity and access management
- Management group and subscription organization
- Network topology and connectivity
- Management and monitoring
- Business continuity and disaster recovery
- Security governance and compliance
- Platform automation and DevOps

Enterprise Agreement enrollment and Azure Active Directory tenants

9/1/2021 • 10 minutes to read • Edit Online

The Azure service presents a range of active subscription offers, and customers can use these offers at the same time to gain flexible billing options. Example subscriptions include Enterprise Agreement (Enterprise Agreement) Support, Microsoft Customer Agreement, Cloud Service Provider, and others.

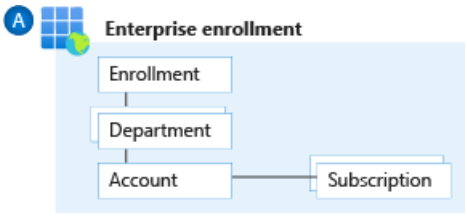


Enterprise-scale architecture supports subscriptions from any Azure offer. Subscriptions should exist within one Azure Active Directory (Azure AD) tenant to then relocate into the management group hierarchy within that tenant. They can then be managed by the various controls with enterprise-scale platforms like Azure Policy and role-based access control (RBAC).

> **NOTE**
> Enterprise-scale architecture is only scoped and deployed to one Azure AD tenant; however, billing options can span across multiple Azure AD tenants. For example, an Enterprise Agreement enrollment supports Azure subscriptions across different Azure AD tenants.

Plan for Enterprise Agreement enrollment

Enterprise Agreement enrollment represents the commercial relationship between Microsoft and how your organization uses Azure. It provides billing foundation for your subscriptions and how your digital estate is administered. The Azure enterprise portal (https://ea.azure.com) helps you to manage your Enterprise Agreement enrollment. An enrollment often represents an organization's hierarchy, including departments, accounts, and subscriptions. This hierarchy represents cost centers within an organization.



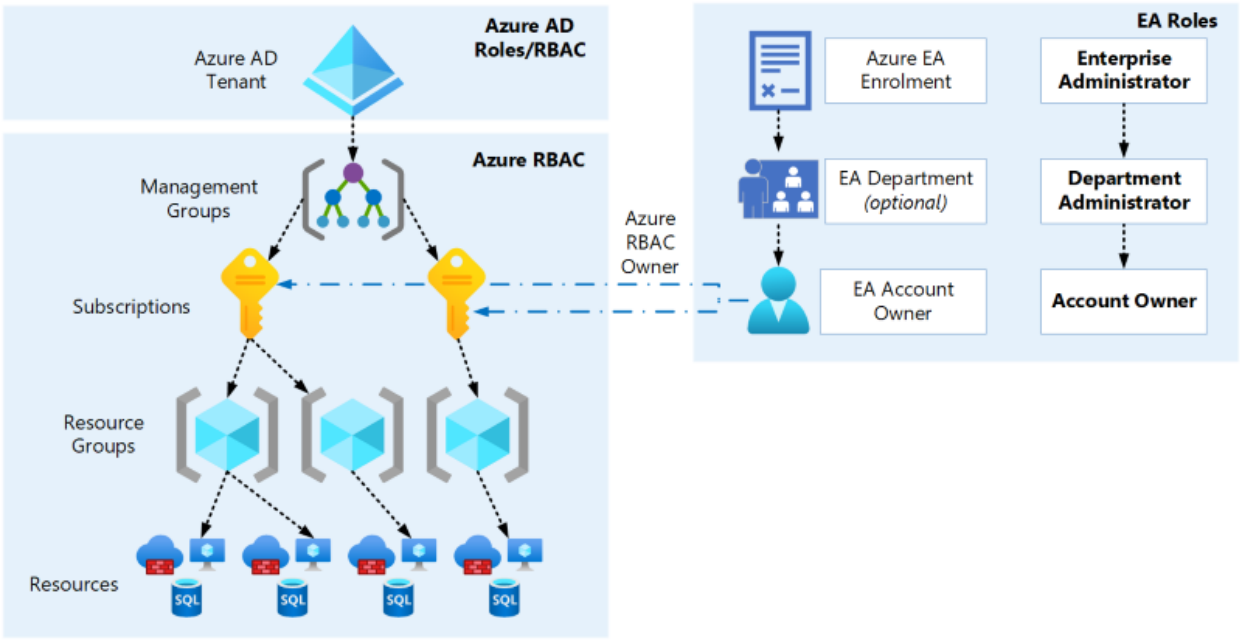
- Departments help to segment costs into logical groupings and set a budget or quota at the department level. The quota isn't firmly enforced; it's used for reporting purposes.
- Accounts are organizational units in the Azure enterprise portal. They can be used to manage subscriptions and access reports.
- Subscriptions are the smallest units in the Azure enterprise portal. They're containers for Azure services that are managed by a service administrator. This is where your organization deploys Azure services.
- Enterprise Agreement enrollment roles link users with their functional role. These roles are:
  - Enterprise administrator
  - Department administrator
  - Account owner
  - Service administrator
  - Notification contact

How an Enterprise Agreement enrollment relates to Azure AD and Azure RBAC

When your organization uses an Enterprise Agreement enrollment for Azure subscriptions, it's important to understand the various authentication and authorization boundaries and the relationship between these boundaries.

There is an inherent trust relationship between Azure subscriptions and an Azure AD tenant, which is described further in Associate or add an Azure subscription to your Azure AD tenant. An Enterprise Agreement enrollment can also use an Azure AD tenant as an identity provider, depending on the authentication level set on the enrollment and which option was selected when the enrollment account owner was created. However, apart from the account owner, Enterprise Agreement enrollment roles don't provide access to Azure AD or the Azure subscriptions within that enrollment.

For example, a finance user is granted an enterprise administrator role on the Enterprise Agreement enrollment. They're a standard user without elevated permissions or roles assigned to them in Azure AD or on any Azure management group, subscription, resource group, or resource. The finance user can only perform the roles listed at Managing Azure Enterprise Agreement roles and can't access the Azure subscriptions on the enrollment. The only Enterprise Agreement role with access to Azure subscriptions is the account owner because this permission was granted when the subscription was created.



Design considerations:

- The enrollment provides a hierarchical organizational structure to govern how subscriptions are managed. See Managing Azure Enterprise Agreement roles.
- A range of administrators can be assigned to a single enrollment.
- Each subscription should have a designated account owner.
- Each account owner is a subscription owner for any subscriptions provisioned under that account.
- A subscription can belong to only one account at a time.
- A specific set of criteria can be used to determine if a subscription should be suspended.
- Departments and accounts can filter enrollment billing and usage reports.
- Review Programmatically create Azure Enterprise Agreement subscriptions with the latest APIs for more information about Enterprise Agreement subscription limitations.

Design recommendations:

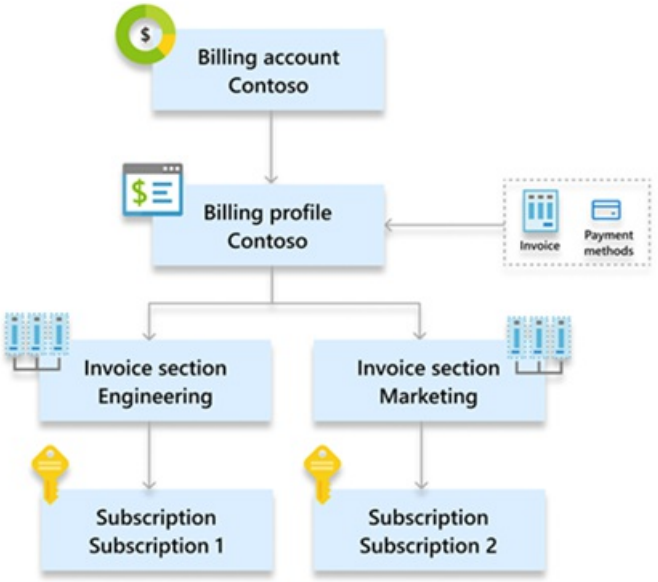
- Only use the authentication type `Work or school account` for all account types. Avoid using the `Microsoft account (MSA)` account type.
- Set up a Notification Contact email address to ensure notifications are sent to an appropriate group mailbox.
- An organization can have various structures, including functional, divisional, geographic, matrix, or team structures. Using departments and accounts to map your organization's structure to your enrollment hierarchy can help with separating billing.
- Use Azure Cost Management reports and views, which can use Azure metadata (for example, tags and location) to explore and analyze your organization's costs.
- Restrict and minimize the number of account owners within the enrollment to limit administrator access to subscriptions and associated Azure resources.
- Assign a budget for each department and account, and establish an alert associated with the budget.
- Create a new department for IT if business domains have independent IT capabilities.

- If you use multiple Azure AD tenants, verify that the account owner is associated with the same tenant as where subscriptions for the account are provisioned.
- For development/testing (dev/test) workloads, use the Enterprise Dev/Test offer, where available. Ensure you comply with the terms of use.
- Don't ignore notification emails sent to the notification account email address. Microsoft sends important Enterprise Agreement prompts to this account.
- Don't move or rename an Enterprise Agreement account in Azure AD.
- Periodically audit the Azure enterprise portal to review who has access, and when possible, avoid using a Microsoft account.
- Enable both **DA View Charges** and **AO View Charges** on every Enterprise Agreement enrollment to allow users with the correct permissions to view Azure cost management data.

Plan for the Microsoft Customer Agreement service

The Microsoft Customer Agreement is a recent and modern Azure service and commerce platform. It represents the commercial relationship between Microsoft and how your organization uses Azure. The agreement enables a streamlined, electronic transaction in an 11-page agreement that doesn't expire. It provides a billing foundation for your subscriptions and affects how your digital estate is administered. You can manage your agreement is managed in the Azure portal.

The Microsoft Customer Agreement often represents an organization's hierarchy, which consists of billing profiles, invoice sections, and subscriptions. This hierarchy represents cost centers within an organization.



> **IMPORTANT**
> If migrating from an Enterprise Agreement to a Microsoft Customer Agreement, please review the following articles:
>
> - Complete Enterprise Agreement tasks in your billing account for a Microsoft Customer Agreement
> - Set up your billing account for a Microsoft Customer Agreement

Design considerations:

- The agreement provides a hierarchical organizational structure to govern how subscriptions are managed. See Organize costs by customizing your billing account.

- An agreement billing account is managed by a single Azure AD tenant. However, subscriptions across different Azure AD tenants are supported by a single agreement. See How tenants and subscriptions relate to billing account and Manage subscriptions under multiple tenants in a single Microsoft Customer Agreement.
- New Azure subscriptions provisioned with an agreement are associated with the Azure AD tenant in which the agreement billing account is located.
- Agreements use the RBAC model. Multiple users can be assigned with the required roles at the same scopes (for example, billing account, billing profile, and invoice section). These billing roles and assignments are outside of standard Azure RBAC roles and assignments. They can't be assigned at a management group or resource group scope.
- A subscription can belong to only one invoice section at any time. Subscriptions can only be moved between invoice sections within the same billing profile.
- An optional purchase order number can be set up on a billing profile.
- A specific set of criteria can be used to determine if a subscription should be suspended.
- Before you provision more billing profiles, review the potential impact to charges and reservations.
- Use Azure Cost Management reports and views, which explore and analyze your organization's costs with Azure metadata.

Design recommendations:

- Set up a Notification Contact email address on the agreement billing account to ensure notifications are sent to an appropriate group mailbox.
- Assign a budget for each invoice section or billing profile, and establish an alert associated with the budget.
- An organization can have a variety of structures, such as functional, divisional, geographic, matrix, or team. Use organizational structures to map your organization to your agreement hierarchy. Invoice sections are suitable for most scenarios.
- If your business domain has independent IT capabilities, create a new invoice section for IT.
- Don't ignore notifications sent to the Contact email address. Microsoft sends important prompts to this address.
- Periodically audit the agreement billing RBAC role assignments to review who has access.
- For development/testing (dev/test) workloads, use the Enterprise Dev/Test offer, where available. Ensure you comply with the terms of use.

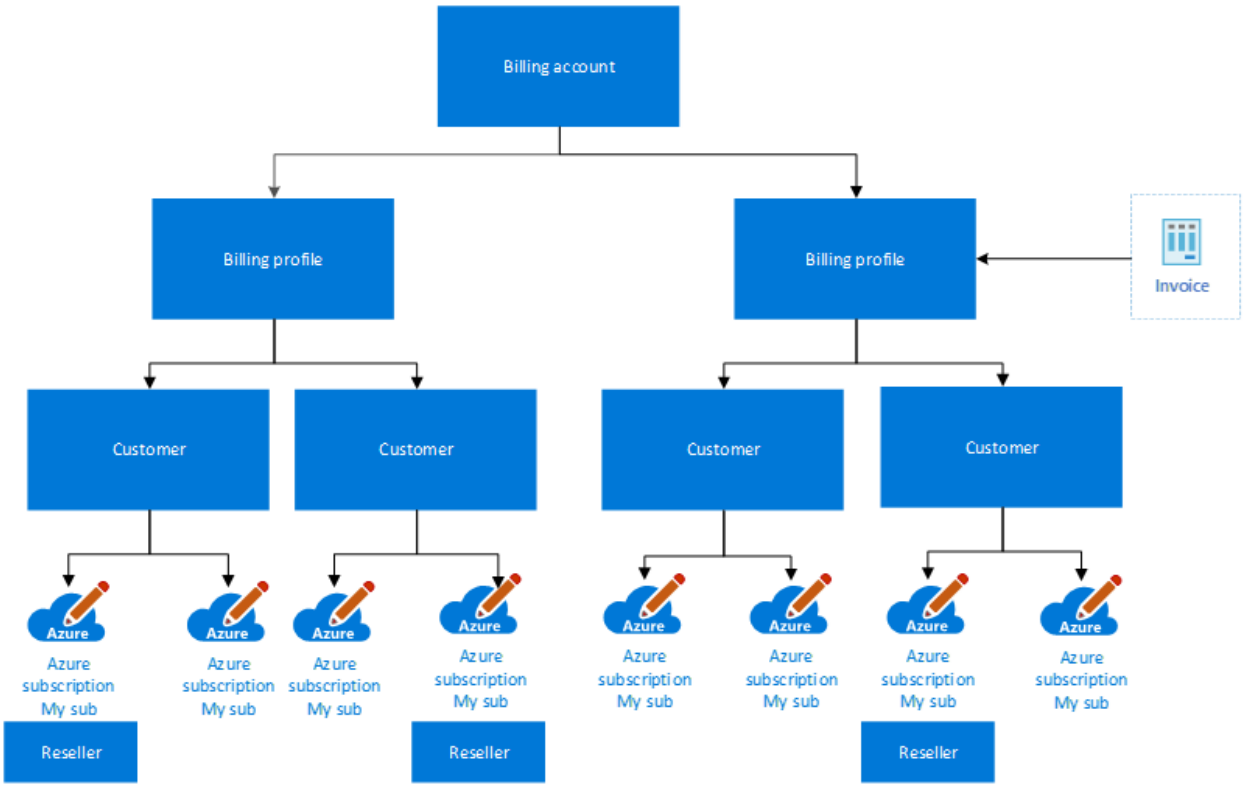
Plan for the Cloud Solution Provider service

The Cloud Solution Provider (CSP) service gives Microsoft partners access to Microsoft cloud services within one platform. It supports partners to:

- Own the customer lifecycle and end-to-end relationship.
- Set pricing, terms, and directly bill customers.
- Directly provision and manage subscriptions.
- Attach services that add value.
- Be the customer's first point of contact for support.

Azure in CSP is an Azure plan with various subscriptions that are hosted by the partner's Microsoft Partner

Agreement (MPA). The MPA is similar to the Microsoft Customer Agreement; both are hosted on the modern commerce platform and use a simplified purchase agreement.



IMPORTANT

The partner CSP completely manages an MPA.

Design considerations:

- A CSP reseller relationship must exist between the partner and each Azure AD tenant in which the customer wants to provision a new Azure plan and CSP subscriptions.
- Only the partner can provision an Azure plan and CSP subscriptions.
- A specific set of criteria can be used to determine if a subscription should be suspended; a partner can also suspend a subscription.
- The partner can allow customers to view their Azure usage fees on a per customer basis. For more information, see Enable the policy to view Azure usage charges. Partners can also use other tools to provide customers with access to their charges.
- By default, only the partner Azure reservations can only be purchased by the partner for their customer. However, the **Customer Permissions** feature grants customers permission to purchase Azure reservations from their CSP.

Design recommendations:

- Work with your CSP partner to ensure that Azure Lighthouse is used for Administer on Behalf of (AOBO) access for most support scenarios. See Azure Lighthouse and the Cloud Solution Provider program.
- Work with your CSP partner to understand how to create support cases and escalation processes.
- Discuss how to create self-service subscriptions with your CSP partner.
- Use Azure Cost Management reports and views, which can use Azure metadata (for example, tags and location) to explore and analyze your organization's costs.

Define Azure AD tenants

An Azure AD tenant provides identity and access management, which is an important part of your security posture. An Azure AD tenant ensures that authenticated and authorized users only access the resources to which they have permissions. Azure AD provides these services to applications and services deployed in and outside of Azure (such as on-premises or third-party cloud providers).

Azure AD is also used by software as a service (SaaS) applications such as Microsoft 365 and Azure Marketplace. Organizations already using on-premises Azure AD can integrate it with their current infrastructure and extend cloud authentication. Each Azure AD directory has one or more domains. A directory can have many subscriptions associated with it but only one Azure AD tenant.

Ask basic security questions during the Azure AD design phase, such as how your organization manages credentials and how it controls human, application, and programmatic access.

Design considerations:

- Multiple Azure AD tenants can function in the same enrollment.
- Azure Lighthouse only supports delegation at the subscription and resource group scopes.

Design recommendations:

- Base using Azure single sign-on on the selected planning topology.
- If your organization doesn't have an identity infrastructure, start by implementing an Azure-AD-only identity deployment. Deployment with Azure AD Domain Services and Microsoft Enterprise Mobility + Security provides end-to-end protection for SaaS applications, enterprise applications, and devices.
- Azure AD multifactor authentication provides another layer of security and authentication. For more security, also enforce conditional access policies for all privileged accounts.
- Plan for emergency access or break-glass accounts to prevent tenant-wide account lockout.
- Use Azure AD Privileged Identity Management to manage identities and access.
- Avoid creating multiple Azure AD tenants. For further information, see Testing approach for enterprise scale and Cloud Adoption Framework Azure best practices guidance to standardize on a single directory and identity.
- Use Azure Lighthouse to grant third parties/partners access to Azure resources in customer Azure AD tenants and centralized access to Azure resources in multitenant Azure AD architectures.

Next steps

Identity and access management

Identity and access management

8/25/2021 • 7 minutes to read • Edit Online

Identity provides the basis of a large percentage of security assurance. It enables access based on identity authentication and authorization controls in cloud services to protect data and resources and to decide which requests should be permitted.

Identity and access management (IAM) is boundary security in the public cloud. It must be treated as the foundation of any secure and fully compliant public cloud architecture. Azure offers a comprehensive set of services, tools, and reference architectures to enable organizations to make highly secure, operationally efficient environments as outlined here.

This section examines design considerations and recommendations related to IAM in an enterprise environment.

Why we need identity and access management

The technological landscape in the enterprise is becoming complex and heterogenous. To manage compliance and security for this environment, IAM enables the right individuals to access the right resources at the right time for the right reasons.

Plan for identity and access management

Enterprise organizations typically follow a least-privileged approach to operational access. This model should be expanded to consider Azure through Azure Active Directory (Azure AD), Azure role-based access control (Azure RBAC), and custom role definitions. It's critical to plan how to govern control- and data-plane access to resources in Azure. Any design for IAM and Azure RBAC must meet regulatory, security, and operational requirements before it can be accepted.

Identity and access management is a multistep process that involves careful planning for identity integration and other security considerations, such as blocking legacy authentication and planning for modern passwords. Staging planning also involves selection of business-to-business or business-to-consumer identity and access management. While these requirements vary, there are common design considerations and recommendations to consider for an enterprise landing zone.

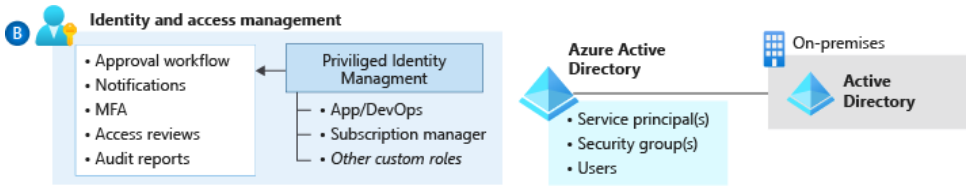


Figure 1: Identity and access management.

Design considerations:

- There are limits around the number of custom roles and role assignments that must be considered when you lay down a framework around IAM and governance. For more information, see Azure RBAC service limits.
- There's a limit of 2,000 role assignments per subscription.
- There's a limit of 500 role assignments per management group.
- Centralized versus federated resource ownership:
  - Shared resources or any aspect of the environment that implements or enforces a security boundary, such as the network, must be managed centrally. This requirement is part of many regulatory frameworks. It's standard practice for any organization that grants or denies access to confidential or critical business resources.
  - Managing application resources that don't violate security boundaries or other aspects required to maintain security and compliance can be delegated to application teams. Allowing users to provision resources within a securely managed environment allows organizations to take advantage of the agile nature of the cloud while preventing the violation of any critical security or governance boundary.
  - Depending on the definition of the centralized or federated resource ownership, custom roles might differ. The custom roles for the centralized resource ownership are limited and might need additional rights depending on the responsibility model. For example, in some organizations a NetOps role might only need to manage and configure global connectivity. But in other organizations that need a

more centralized approach, the NetOps role needs to be enriched with more allowed actions like creating peering between the hub and the spokes.

Design recommendations:

- Use Azure RBAC to manage data-plane access to resources, where possible. Examples are Azure Key Vault, a storage account, or a SQL database.
- Deploy Azure AD conditional-access policies for any user with rights to Azure environments. Doing so provides another mechanism to help protect a controlled Azure environment from unauthorized access.
- Enforce multi-factor authentication for any user with rights to the Azure environments. Multi-factor authentication enforcement is a requirement of many compliance frameworks. It greatly lowers the risk of credential theft and unauthorized access.
- Use Azure AD Privileged Identity Management (PIM) to establish zero standing access and least privilege. Map your organization's roles to the minimum level of access needed. Azure AD PIM can either be an extension of existing tools and processes, use Azure native tools as outlined, or use both as needed.
- Use Azure-AD-only groups for Azure control-plane resources in Azure AD PIM when you grant access to resources.
  - Add on-premises groups to the Azure-AD-only group if a group management system is already in place.
- Use Azure AD PIM access reviews to periodically validate resource entitlements. Access reviews are part of many compliance frameworks. As a result, many organizations will already have a process in place to address this requirement.
- Integrate Azure AD logs with the platform-central Log Analytics workspace. It allows for a single source of truth around log and monitoring data in Azure, which gives organizations cloud-native options to meet requirements around log collection and retention.
- If any data sovereignty requirements exist, custom user policies can be deployed to enforce them.
- Use custom role definitions within the Azure AD tenant while you consider the following key roles:

ROLE	USAGE	ACTIONS	NO ACTIONS
Azure platform owner (such as the built-in Owner role)	Management group and subscription lifecycle management	`*`	
Network management (NetOps)	Platform-wide global connectivity management: Virtual networks, UDRs, NSGs, NVAs, VPN, Azure ExpressRoute, and others	`*/read`, `Microsoft.Network/*`, `Microsoft.Resources/deployments/*`, `Microsoft.Support/*`	
Security operations (SecOps)	Security administrator role with a horizontal view across the entire Azure estate and the Azure Key Vault purge policy	`*/read`, `*/register/action`, `Microsoft.KeyVault/locations/deletedVaults/purge/action`, `Microsoft.PolicyInsights/*`, `Microsoft.Authorization/policyAssignments/*`, `Microsoft.Authorization/policyDefinitions/*`, `Microsoft.Authorization/policyExemptions/*`, `Microsoft.Authorization/policySetDefinitions/*`, `Microsoft.Insights/alertRules/*`, `Microsoft.Resources/deployments/*`, `Microsoft.Security/*`, `Microsoft.Support/*`	

ROLE	USAGE	ACTIONS	NO ACTIONS
Subscription owner	Delegated role for subscription owner derived from subscription Owner role	`*`	`Microsoft.Authorization/*/write`, `Microsoft.Network/vpnGateways/*`, `Microsoft.Network/expressRouteCircuits/*`, `Microsoft.Network/routeTables/write`, `Microsoft.Network/vpnSites/*`
Application owners (DevOps/AppOps)	Contributor role granted for application/operations team at resource group level	`*`	`Microsoft.Authorization/*/write`, `Microsoft.Network/publicIPAddresses/write`, `Microsoft.Network/virtualNetworks/write`, `Microsoft.KeyVault/locations/deletedVaults/pu`

- Use Azure Security Center just-in-time access for all infrastructure as a service (IaaS) resources to enable network-level protection for ephemeral user access to IaaS virtual machines.
- Use Azure AD managed identities for Azure resources to avoid authentication based on user names and passwords. Because many security breaches of public cloud resources originate with credential theft embedded in code or other text sources, enforcing managed identities for programmatic access greatly reduces the risk of credential theft.
- Use privileged identities for automation runbooks that require elevated access permissions. Automated workflows that violate critical security boundaries should be governed by the same tools and policies users of equivalent privilege are.
- Don't add users directly to Azure resource scopes. Instead add users to defined roles, which are then assigned to resource scopes. Direct user assignments circumvent centralized management, greatly increasing the management required to prevent unauthorized access to restricted data.

Plan for authentication inside a landing zone

A critical design decision that an enterprise organization must make when adopting Azure is whether to extend an existing on-premises identity domain into Azure or to create a brand new one. Requirements for authentication inside the landing zone should be thoroughly assessed and incorporated into plans to deploy Active Directory Domain Services (AD DS) in Windows Server, Azure AD Domain Services (Azure AD DS), or both. Most Azure environments will use at least Azure AD for Azure fabric authentication and AD DS local host authentication and group policy management.

Design considerations:

- Consider centralized and delegated responsibilities to manage resources deployed inside the landing zone.
- Applications that rely on domain services and use older protocols can use Azure AD DS.
- There is a difference between Azure AD, Azure AD DS, and AD DS running on Windows Server. Evaluate your application needs, and understand and document the authentication provider that each one will be using. Plan accordingly for all applications.

Design recommendations:

- Use centralized and delegated responsibilities to manage resources deployed inside the landing zone based on role and security requirements.
- Privileged operations such as creating service principal objects, registering applications in Azure AD, and procuring and handling certificates or wildcard certificates require special permissions. Consider which users will be handling such requests and how to secure and monitor their accounts with the degree of diligence required.
- If an organization has a scenario where an application that uses integrated Windows authentication must be accessed remotely through Azure AD, consider using Azure AD Application Proxy.
- Evaluate the compatibility of workloads for AD DS on Windows Server and for Azure AD DS.
- Ensure your network design allows resources that require AD DS on Windows Server for local authentication and management to access the appropriate domain controllers.
    - For AD DS on Windows Server, consider shared services environments that offer local authentication and host management in a larger enterprise-wide network context.

- Deploy Azure AD DS within the primary region because this service can only be projected into one subscription. The Azure AD DS managed domain can be expanded to further regions with replica sets.
- Use managed identities instead of service principals for authentication to Azure services. This approach reduces exposure to credential theft.

Management group and subscription organization

8/18/2021 • 29 minutes to read • Edit Online

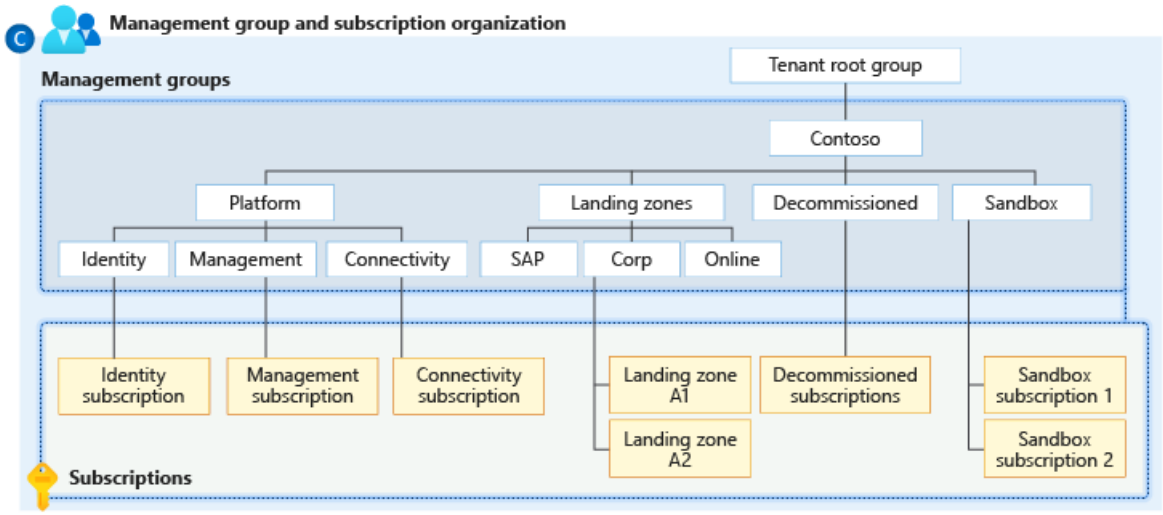


Figure 1: Management group hierarchy.

Define a management group hierarchy

Management group structures within an Azure Active Directory (Azure AD) tenant support organizational mapping. Consider your management group structure thoroughly as your organization plans Azure adoption at scale.

Design considerations:

- Management groups can be used to aggregate policy and initiative assignments via Azure Policy.
- A management group tree can support up to six levels of depth. This limit doesn't include the tenant root level or the subscription level.
- Any principal (user or service principal) within an Azure AD tenant can create new management groups, since Azure role-based access control (RBAC) authorization for management group operations isn't enabled by default.
- All new subscriptions will be placed under the root management group by default.

Design recommendations:

- Keep the management group hierarchy reasonably flat, with no more than three to four levels ideally. This restriction reduces management overhead and complexity.
- Avoid duplicating your organizational structure into a deeply nested management group hierarchy. Management groups should be used for policy assignment versus billing purposes. This approach necessitates using management groups for their intended purpose in enterprise-scale architecture, which is providing Azure policies for workloads that require the same type of security and compliance under the same management group level.
- Create management groups under your root-level management group to represent the types of workloads (archetypes) that you'll host and ones based on their security, compliance, connectivity, and feature needs. This grouping structure allows you to have a set of Azure policies applied at the management group level for all workloads that require the same security, compliance, connectivity, and

feature settings.

- Use resource tags, which can be enforced or appended through Azure Policy, to query and horizontally navigate across the management group hierarchy. Then you can group resources for search needs without having to use a complex management group hierarchy.
- Create a top-level sandbox management group to allow users to immediately experiment with Azure. Users can then experiment with resources that might not yet be allowed in production environments. The sandbox provides isolation from your development, test, and production environments.
- Use a dedicated service principal name (SPN) to execute management group management operations, subscription management operations, and role assignment. Using an SPN reduces the number of users who have elevated rights and follows least-privilege guidelines.
- Assign the User Access Administrator role at the root management group scope (`/`) to grant the SPN just mentioned access at the root level. After the SPN is granted permissions, the User Access Administrator role can be safely removed. In this way, only the SPN is part of the User Access Administrator role.
- Assign the Contributor role to the SPN previously mentioned at the root management group scope (`/`), which allows tenant-level operations. This permission level ensures that the SPN can be used to deploy and manage resources to any subscription within your organization.
- Create a `Platform` management group under the root management group to support common platform policy and Azure role assignment. This grouping structure ensures that different policies can be applied to the subscriptions used for your Azure foundation. It also ensures that the billing for common resources is centralized in one set of foundational subscriptions.
- Limit the number of Azure Policy assignments made at the root management group scope (`/`). This limitation minimizes debugging inherited policies in lower-level management groups.
- Use the policies available for enterprise-scale landing zones to enforce compliance requirements either at management group or subscription scope. Refer to guidance in the Policy-driven governance section to learn more about the governance requirements that can be addressed.
- Ensure that only privileged users can operate management groups in the tenant by enabling Azure RBAC authorization in the management group hierarchy settings (by default, all users are authorized to create their own management groups under the root management group).
- Configure a default, dedicated management group for new subscriptions to ensure that no subscriptions are placed under the root management group. This is especially important if there are users eligible for MSDN or Visual Studio benefits and subscriptions. A good candidate for this type of management group is a `Sandbox` management group.

> **NOTE**
> For testing guidance for enterprise-scale deployments, see Testing approach for enterprise-scale.

Subscription organization and governance

Subscriptions are a unit of management, billing, and scale within Azure. They play a critical role when you're designing for large-scale Azure adoption. This section helps you capture subscription requirements and design target subscriptions based on critical factors. These factors are environment type, ownership and governance model, organizational structure, and application portfolios.

Design considerations:

- Subscriptions serve as boundaries for assigning Azure policies. For example, secure workloads such as Payment Card Industry (PCI) workloads typically require additional policies to achieve compliance. Instead of using a management group to group workloads that require PCI compliance, you can achieve the same isolation with a subscription. This way, you don't have too many management groups with a small number of subscriptions.
- Subscriptions serve as a scale unit so that component workloads can scale within the platform subscription limits. Make sure to consider subscription resource limits during your workload design sessions.
- Subscriptions provide a management boundary for governance and isolation, which clearly separates concerns.
- There's a manual process, planned future automation, that can be conducted to limit an Azure AD tenant to use only Enterprise Agreement enrollment subscriptions. This process prevents creation of Microsoft Developer Network subscriptions at the root management group scope.

Design recommendations:

- Treat subscriptions as a democratized unit of management aligned with business needs and priorities.
- Make subscription owners aware of their roles and responsibilities:
  - Perform an access review in Azure AD Privileged Identity Management quarterly or twice a year to ensure that privileges don't proliferate as users move within the customer organization.
  - Take full ownership of budget spending and resource utilization.
  - Ensure policy compliance and remediate when necessary.
- Use the following principles when identifying requirements for new subscriptions:
  - **Scale limits:** Subscriptions serve as a scale unit for component workloads to scale within platform subscription limits. For example, large, specialized workloads such as high-performance computing, IoT, and SAP are all better suited to use separate subscriptions to avoid limits (such as a limit of 50 Azure Data Factory integrations).
  - **Management boundary:** Subscriptions provide a management boundary for governance and isolation, which allows for a clear separation of concerns. For example, different environments such as development, test, and production are often isolated from a management perspective.
  - **Policy boundary:** Subscriptions serve as a boundary for the assignment of Azure policies. For example, secure workloads such as PCI typically require additional policies to achieve compliance. This additional overhead doesn't need to be considered holistically if a separate subscription is used. Similarly, development environments might have more relaxed policy requirements relative to production environments.
  - **Target network topology:** Virtual networks can't be shared across subscriptions, but they can connect with different technologies such as virtual network peering or Azure ExpressRoute. Consider which workloads must communicate with each other when you decide whether a new subscription is required.
- Group subscriptions together under management groups aligned within the management group structure and policy requirements at scale. Grouping ensures that subscriptions with the same set of policies and Azure role assignments can inherit them from a management group, which avoids duplicate assignments.
- Establish a dedicated management subscription in the `Platform` management group to support global management capabilities such as Azure Monitor Log Analytics workspaces and Azure Automation runbooks.
- Establish a dedicated identity subscription in the `Platform` management group to host Windows Server

Active Directory domain controllers, when necessary.

- Establish a dedicated connectivity subscription in the `Platform` management group to host an Azure Virtual WAN hub, private Domain Name System (DNS), ExpressRoute circuit, and other networking resources. A dedicated subscription ensures that all foundation network resources are billed together and isolated from other workloads.
- Avoid a rigid subscription model, and opt instead for a set of flexible criteria to group subscriptions across the organization. This flexibility ensures that as your organization's structure and workload composition changes, you can create new subscription groups instead of using a fixed set of existing subscriptions. One size doesn't fit all for subscriptions. What works for one business unit might not work for another. Some applications might coexist within the same landing zone subscription while others might require their own subscription.

Configure subscription quota and capacity

Each Azure region contains a finite number of resources. When you consider an enterprise-scale Azure adoption that involves large resource quantities, ensure that sufficient capacity and SKUs are available and that the attained capacity can be understood and monitored.

Design considerations:

- Consider limits and quotas within the Azure platform for each service that your workloads require.
- Consider the availability of required SKUs within chosen Azure regions. For example, new features might be available only in certain regions. The availability of certain SKUs for given resources such as VMs might be different from one region to another.
- Consider that subscription quotas aren't capacity guarantees and are applied on a per-region basis.

Design recommendations:

- Use subscriptions as scale units, and scale out resources and subscriptions as required. Your workload can then use the required resources for scaling out, when needed, without hitting subscription limits in the Azure platform.
- Use reserved instances to prioritize reserved capacity in required regions. Then your workload will have the required capacity even when there's a high demand for that resource in a specific region.
- Establish a dashboard with custom views to monitor used capacity levels. Set up alerts if capacity utilization is reaching critical levels (for example, CPU utilization of 90 percent).
- Raise support requests for quota increase as a part of subscription provisioning (for example, total available VM cores within a subscription). This approach ensures your quota limits are set before your workloads require going over the default limits.
- Ensure required services and features are available within the chosen deployment regions.

Configure subscription tenant transfer restrictions

Each Azure subscription is linked to a single Azure AD tenant, this acts as the identity provider (IdP) for the Azure subscription which it will trust to authenticate users, services and devices.

However, the Azure AD tenant that an Azure subscription is linked to can be changed by a user with the required permissions that also exists in more than one Azure AD tenant, again, with the required permissions. This process is detailed in the following articles:

- Associate or add an Azure subscription to your Azure Active Directory tenant

- Transfer an Azure subscription to a different Azure AD directory

> **NOTE**
> For Azure cloud solution provider (CSP) subscriptions, transferring to another Azure AD tenant is not supported.

When this is considered in the context of enterprise-scale, you may decide to restrict and prevent users from being able to either transfer Azure subscriptions out of your organization's Azure AD tenant or transfer Azure subscriptions into your organization's Azure AD tenant. This can be done by following the process detailed in Manage Azure subscription policies.

As part of the subscription policy you can configure, you can also provide a list of exempted users who are permitted to bypass what's set in the policy.

> **IMPORTANT**
> This is not an Azure Policy.

Design considerations:

- Consider whether users with Visual Studio/MSDN Azure subscriptions should be allowed to transfer their subscription to or from the Azure AD tenant.
- These settings are only configurable by users with the Azure AD Global Administrator role assigned and they must have elevated their access to be able to change the policy.
  - Only individual user accounts can be specified as exempted users. Azure AD groups are not supported at this time.
- All users with access to Azure will be able to view the policy defined for the Azure AD tenant.
  - They will not be able to view the exempted users.
  - They will be able to view the global administrators within the Azure AD tenant.
- Azure subscriptions transferred into an Azure AD tenant will be placed into the default management group for that tenant.
- Define a process to allow Azure subscriptions to be transferred to or from an Azure AD tenant, if it's deemed necessary by the application team and approved by the organization.

Design recommendations:

- Prevent all users, by default, from being able to transfer Azure subscriptions to or from the Azure AD tenant by configuring the following settings:
  - **Subscription leaving Azure AD directory** set to `Permit no one`.
  - **Subscription entering Azure AD directory** set to `Permit no one`.
- Configure a limited set of exempted users.
  - This would ideally be members of an Azure PlatformOps (platform operations) team.
  - Include break-glass accounts in the list of exempted users.

Establish cost management

Cost transparency across a technical estate is a critical management challenge faced by every large enterprise

organization. This section explores key aspects associated with how cost transparency can be achieved across large Azure environments.

Design considerations:

- There could be a need for chargeback models where shared platform as a service (PaaS) resources are concerned, such as Azure App Service Environment and Azure Kubernetes Service, which might need to be shared to achieve higher density.
- Use a shutdown schedule for nonproduction workloads to optimize costs.
- Use Azure Advisor to check recommendations for optimizing costs.

Design recommendations:

- Use Azure Cost Management + Billing to aggregate costs. Make it available to application owners.
- Use Azure resource tags categorize costs and group resources. Using tags allows you to have a chargeback mechanism for workloads that share a subscription or for a given workload that spans across multiple subscriptions.

Policy-driven governance

Organizations can use enterprise-scale Azure policies to enforce the following governance requirements:

- **Prevent public IP-based services:**

  Most Azure platform as a service (PaaS) services are created with a public IP address assigned to each service. This option can help developers who need to start using these services quickly.

  Public endpoint accelerates the learning curve and supports developing pilots and small-scale proof of concept (POC) implementations, but developers sometimes overlook their public IP addresses when transitioning pilots/POCs to production-ready enterprise applications.

  Security risks can increase when production workloads use public IPs without proper security measures. One type of threat is an actor using a public IP as a gateway to launch an attack. To minimize security risks, many enterprise compliance policies don't allow public IPs.

  There's a custom policy that targets a scope and prevents a public IP address from being created there. Enterprises can also use this policy to create virtual machines (VMs) without public IPs. Similarly, there's a custom policy initiative/policy set that also helps enterprises to prevent Azure services from being created with public IP addresses.

- **Collect audit and log information:**

  A lack of auditing and diagnostics information at a granular level can affect operational practices. Incomplete audit information makes it difficult to correlate logs from multiple Azure services and create cohesive debugging.

  Once Azure services are provisioned, they should provide detailed information about how they interact with the Azure platform. This information can be broadly divided into logs and metrics, and each Azure service can be further grouped into its subcomponents. Ror example, an Azure public IP resource with `DDoSProtectionNotifications`, `DDoSMitigationReports`, and `DDoSMitigationFlowLogs` as its subcomponents. Collecting diagnostic information at these subcategories can also help organizations to enhance auditing and debugging.

  A custom Azure Policy initiative is available to help enterprises gather logs and metrics at a deeper level for each Azure service. This initiative includes a policy for every Azure service, and the policies collect key log categories and metrics automatically.

- **Provide comprehensive security for SQL databases:**

  SQL databases are a common Azure service in most Azure deployments. Unfortunately, they're also target for malicious activity both inside and outside an enterprise. A custom Azure Policy initiative for SQL databases helps organizations to apply the following key governance practices:

  - **Encrypt SQL data at rest:**

    SQL databases and their backups are vulnerable to malicious actors. Since it's simple to restore SQL databases from database files or backups, malicious actors can gain access to this data if a proper defense system isn't in place.

    One of the first steps in building a SQL database defense strategy is ensuring that a SQL database is encrypted at rest. Azure SQL Database transparent data encryption (TDE) encrypts database data at rest without changing an application's code and presents obstacles for malicious actors trying access the data, even if it's compromised. As SQL Database deployments increase within an enterprise, it's important to create them with TDE. There's a custom policy to ensure TDE for SQL databases.

  - **Optimize alerts for suspicious activity:**

    Bad actors aim to access and exploit business-critical SQL databases. When enterprises don't acknowledge these attempts, their risk of not detecting and resolving these incidents increases. In a worst-case scenario, an enterprise might not know whether its SQL database has been compromised.

    SQL Database lets enterprises set up security alerts that report suspicious Microsoft SQL Server activity. These alerts reach a preconfigured email address and optionally, Azure subscription admins and owners. They can expose malicious activities like SQL injection attacks, brute-force attacks, and more.

    A custom policy is available to enable security alerts on SQL databases. Security alerts provide detailed information about every incident, which is visible in the Azure portal or emails from when alerts are triggered.

  - **Examine an audit trail of operations:**

    A business-critical SQL database can operate daily with a range of data manipulation, control, and definition language commands. Without clear controls and insights into these operational activities, it can be challenging to distinguish between legitimate and suspicious operations.

    Enabling SQL auditing can help enterprises to gather important information about all database activities, and SQL auditing helps enterprises to create and analyze an audit trail of database events. This is also part of many industry/regional regulatory compliance requirements.

    Enterprises can use a custom policy to enforce SQL database auditing. This policy audits and reports on key database events like ownership, role membership, or schema changes, successful and failed logins, and more. Enterprises can use this policy's audit trail to gain insights about database operations and comply with industry or regional regulatory requirements.

- **Evaluate proven best practices:**

  A SQL database can experience a lot of schema, permission, and configuration changes throughout its lifecycle, and these changes can deviate from best practices. In turn, excessive permissions, orphaned roles, and other configurational drifts can be exploited by malicious actors.

  Microsoft best practices for SQL databases can help enterprises to assess their SQL databases, and SQL Database has a built-in vulnerability assessment service to assist. A vulnerability assessment scans and identifies database- and server-level security risks, and it offers remediation tasks that can fix

vulnerabilities.

A custom policy ensures that SQL databases are configured with vulnerability assessments. Assessment scans run periodically, reports are stored in an Azure Storage account, and a predefined email address shares the results for reporting.

- **Protect secrets from being deleted intentionally or accidentally:**

  Azure Key Vault stores confidential information like keys, certificates, passwords, and more. A malicious user can abuse the service by deleting the secrets that it stores, and a standard user could accidentally delete the sensitive stored within. Without proper provisions, malicious or accidental deletion in Azure Key Vault could harm the business.

  The **soft-delete** feature in Azure Key Vault can protect enterprises against intentional or accidental deletion. With **soft-delete** enabled, deleted keys are retained for a predefined time period. If the delete operation was intended, then key content can be deleted until another purge, typically by a user with more access privileges. If the delete operation was an accident, then the deleted keys can be restored within the time defined.

  Azure offers a custom policy to ensure that **soft-delete** is enabled by default with Azure Key Vault. This policy provides an extra security layer if Azure Key Vault content is deleted maliciously, and enterprises gain more control if content is deleted accidentally.

- **Enforce Azure Application Gateway Web Application Firewall:**

  Web applications running on Azure are potential targets for malicious attacks. The top 10 web application security risks like injection, cross-site scripting, and others exploit web application vulnerabilities, and a successful attack can cost an organization its resources and reputation.

  Azure Application Gateway Web Application Firewall (WAF) uses the Open Web Application Security Project, OWASP, Core Rule Set 3.1, 3.0, or 2.2 to protect web applications from common attacks. WAF policies in Application Gateway can be accessed **Prevention** or **Detection** mode.

  Azure offers a custom policy to help prevent potential misconfiguration in Application Gateway; it creates a WAF for Application Gateway by default. The WAF protects Azure web applications using Application Gateway.

- **Prevent IP forwarding on VMs:**

  IP forwarding supports an Azure VM to route its traffic to other destinations. Unless specifically required, this routing can expose a VM and other unintended networks with a public IP address as a router.

  Azure provides an option to configure IP forwarding on virtual machines (VMs) with tools like firewalls, load balancers, and others deployed via Azure Marketplace. Any application can use these services can use them in Azure Marketplace transactions.

  Outside of specific needs, IP forwarding on VMs can be a security liability. Azure offers a custom policy to prevent VMs acting as IP forwarding routers, and this policy is applied at the landing zone scope. Focusing on VMs in landing zones should be the final destination for user requests; any routing should be implemented in the connectivity subscription.

- **Enforce centralized DNS record management:**

  Private Azure DNS zones help to create and manage DNS records for Azure resources. Without controls for how these zones proliferate, management and network connectivity debugging issues can result. In hybrid environments where on-premises sites need to connect to Azure resources, fragmented DNS zones can create duplicate DNS records and maintenance challenges.

  Enterprises can deploy private Azure DNS zones centrally to manage DNS records clearly. Azure Virtual

Network can link with private zones to help run domain controllers, which can streamline connectivity from on-premises sites. Azure services that support Azure Private Link and Private Endpoint can use centrally managed private DNS zones without creating them during each application deployment.

Azure offers a custom policy that can prevent the creation of a private DNS zone in the scope during which it's applied. Enterprises can view this policy's compliance status even when the policy enforcement is disabled. This policy helps to streamline connectivity between on-premises sites and access to Azure PaaS services using Private Link or Private Endpoint.

- **Enforce network traffic control:**

  An Azure virtual network can be divided into multiple subnets. Since network access controls between these subnets don't exist by default, this can result in unsolicited network traffic in a subnet.

  Azure network security groups (NSGs) help to filter incoming and outgoing subnet traffic, and stateful packet inspections can allow or deny network traffic. Resources inside subnets can only receive traffic from allowed IP address ranges.

  Azure offers a custom policy that pairs every subnet with an NSG. A combination of subnet and an NSG ensures that a default set of rules controls traffic to and from a subnet. Depending on their needs, enterprises can also add or modify rules to control traffic further.

- **Use Azure Security Center to detect and protect against security threats:**

  An Azure subscription can hold a range of resources like VMs, container images, an more, and these resources are exposed to risks like malware/unwanted software installation, uncontrolled access to management ports on a VM, and others. With security attacks becoming more sophisticated and a limited-supply of experienced security professionals, detecting security vulnerabilities and protecting workloads is extremely challenging.

  Azure Security Center is the Azure native security management system that assesses Azure resources' security posture against security best practices. It helps to detect and prevent threats against data and application services, and with multiple integration points, it can be deployed quickly.

  Azure offers a custom policy that pairs Azure subscriptions with Security Center, helping subscriptions to quickly start receiving Security Center threat detection and protection. This policy automatically covers key Azure services like VMs, storage accounts, and seven others with Security Center. If deviation from security best practice occurs, enterprises benefit from continuous security assessments and actionable recommendations.

- **Protect against ransomware attacks and data loss:**

  The increasing frequency of ransomware and intrusion attacks present another concern for enterprises. A successful ransomware attack can disrupt business-critical processes and applications, and attackers have held enterprises hostage for large amounts of money.

  Azure Backup protects Azure virtual machine data from being destroyed maliciously or accidentally. Backups are easy to configure and scale, and Azure Recovery Services backs up the data for simple management and protection.

  Azure offers a custom policy that protects virtual machines by configuring them with Azure Backup. This policy automatically provisions an Azure Recovery Services vault and creates backup container for every Azure VM created.

- **Protect against distributed denial of service attacks:**

  Publicly reachable Azure resources are vulnerable to distributed denial-of-service (DDoS) attacks. These attacks can affect an application's availability to its intended users, and prolonged attacks can exhaust all available resources and create downtime for business-critical applications.

Azure DDoS protection defends Azure resources against DDoS attacks by continuously monitoring incoming traffic to identify potential threats. During an active attack, enterprises can benefit from working with the Microsoft DDoS rapid response team.

Azure offers a custom policy that automatically provisions an Azure DDoS Protection Standard plan on all Azure subscriptions in its scope. This policy allows enterprises to select the Azure regions that will be covered by the service.

- **Auto-provision Private Link or Private Endpoint with private DNS zones:**

  One enterprise maintenance challenge is how to create private DNS zones for every application that needs access to Azure PaaS services. Azure Private Link and Private Endpoint use private IP addresses to provide access to Azure platform as a service (PaaS) services, and private DNS zones resolve DNS records. Private DNS zone groups use categorizes from Azure services like blob, queue, table, and SQL to group Private Link connections and use one private DNS zone per service.

  Enterprises can also create central private DNS zones, and custom Azure policies can automatically connect Private Link or Private Endpoint with private DNS zones for Azure services.

- **Manage firewall rules centrally:**

  Fragmented firewall rules can lead to uncontrolled and ambiguous network traffic paths. Continuous changes to firewalls rules for every firewall instance make it difficult to assess network security posture, and multiple rules make it difficult to distinguish between a centrally managed basic set of rules and workload-specific network path rules.

  Azure Firewall policies to help organizations define a minimum set of rules that apply throughout their organization. Application-specific policies can inherit basic rules to create hierarchical rules that meet enterprise- and application-specific firewall requirements. When rules are configured through policies, they can be managed and monitored centrally.

  Azure offers a custom policy that helps enterprises to define Azure Firewall policies centrally. Enterprises control defining the rules and priorities that to meet their network traffic routing requirements. Depending on their needs, enterprises can define firewall policies centrally and apply them to either Azure Virtual WAN or hub-and-spoke network topology.

- **Provision hub-and-spoke network topology:**

  As more workloads start to get deployed in Azure, they begin using a common set of services such as firewalls, VPN gateways, and others. If not carefully planned, common services can replicate per application deployment, creating unnecessary costs and operational overhead. In scenarios where on-premises connectivity is needed from Azure and connectivity is established per application deployment, network topology becomes more difficult to maintain.

  Azure hub-and-spoke network topology helps to streamline needs for network connectivity. A hub virtual network (VNet) can host shared services while spoke VNets host application-specific Azure resources. Hub-and-spoke VNets use VNet peering to connect with each other, and the network topology promotes clean network design, easier management, and optimized costs.

  Azure offers a custom policy that uses Azure Firewall and gateways from VPNs and ExpressRoute to provisions hub VNets. Enterprises can configure all options for firewalls and gateways as part of the policy assignment. This policy simplifies the process to deploy Azure hub-and-spoke network topology.

  Another custom policy prevents two VNets from peering with each other to instead communicate with each other via a hub VNet. Configuring VNets to communicate with each other through with hubs makes it possible to control and monitor network connections. Network topology is simplified from a maintenance perspective as well.

- **Provision default configurations for Azure Monitor:**

  The inability to identify and visualize the relationship between the Azure platform, its services, and applications can lead to an outage or undetected and degraded performance. Operations or support teams could miss opportunity to correct specific conditions, and an Azure application might not scale itself to respond to a surge or slump in the demand.

  Azure Monitor Logs and Log Analytics workspaces use alerts to help enterprises understand and resolve critical conditions. They use dashboards, workbooks, and Microsoft Power BI to support enterprises to visualize and interact with a robust set of log information. Enterprises can use Azure Monitor Logs and Log Analytics workspaces together to configure VM autoscaling and automatically add or remove extra instances.

  Azure offers a custom policy to configure Azure Monitor Logs with Log Analytics workspaces. This policy deploys prepackaged dashboard reports from Azure Monitor solutions for specific Azure services like Azure SQL Database or Azure AD. It also configures data sources from Linux or Windows VM performance metrics with Azure Monitor.

- **Enable log storage and queries:**

  If not carefully planned, log information from multiple Azure sources can become unmanageable, as capturing, storing, and managing logs can consume resources, time, and costs. Identifying trends or patterns over a long period of time and a large number of logs can also become challenging. Log Analytics queries use alerts and interactive reports to help enterprises to efficiently store and manage logs from multiple sources.

  Azure offers a custom policy that creates a Log Analytics workspace to serve as a repository that stores data logs. An Azure Automation account is created, and it links to a Log Analytics workspace to automate tasks or deploy Azure Monitor solutions that could depend on those workspaces. The policy also helps to configure log retention periods, Azure regions, and other properties.

- **Provision logging for Azure Arc enabled servers:**

  With IT estates spanning across multiple clouds, on-premises sites, and edge locations, enterprises could struggle to manage and govern servers that are scattered across environments and geographic locations. The idea of using a range of products to monitor these servers could be overwhelming, and assigning servers to multiple environments under one unified access and identity management solution can be challenging to set up and manage.

  Azure Arc simplifies how resources like servers, Kubernetes clusters, and data services are governed and managed across heterogeneous. By projecting hybrid resources as native Azure resources, Azure Arc provides a single control pane for managing native and hybrid resources, bringing native and hybrid resources under a unified role-based-access-control solution.

  Azure offers two custom policies that can help enterprises to set up Log Analytics agents on Azure Arc enabled Linux and Windows servers. A Log Analytics workspace is also configured to store and manage logs. When assigned successfully, the policy identifies the name of servers within its scope and assigns it to a Log Analytics agent.

- **Enforce collecting network traffic logs:**

  Even though virtual network and subnets provide logical private network boundaries, it's still necessary to monitor network traffic in Azure. Without proper monitoring, enterprise networks are vulnerable to malicious or unknown traffic from compromised IP addresses. Without an understanding of the current traffic, it can be challenging to provision extra capacity for increasing in network traffic.

  Azure Network Watcher helps enterprises to monitor and repair network issue for infrastructure as a service (IaaS) services in Azure. With this service, NSG flow logs provide a way to capture information

about network traffic. Enterprises can benefit from traffic analysis and patterns, forecast future capacity needs, and enforce compliance with corporate governance policies.

Azure offers a custom policy for setting up NSG flow logs in Network Watcher. Here, a storage account is provisioned as a repository to store NSG flow logs. This policy also configuring the retention period to store the NSG flow logs.

- **Provision an at-scale network connectivity solution:**

  Enterprise network connectivity requirements can be complex. Constant requests for adding new sites, devices, and users to an ever-expanding network are challenging to provision and manage, and network bandwidth and throughput demands from multiple touchpoints in an enterprise can be demanding.

  Azure Virtual WAN is an enterprise-level network service that helps organizations to resolve connectivity challenges. The service provides higher aggregate throughput with network connectivity, optimal routing over the Azure backbone, and a unified Azure management experience.

  Azure offers a custom policy to set up Virtual WAN and provision a virtual hub within the service. Enterprises can deploy virtual hubs to act as a central point for connections from multiple sources and destinations. ExpressRoute, Azure VPN Gateway, and Azure Firewall are also provisioned to address network connectivity requirements.

- **Back up virtual machines:**

  As cloud adoption increases, enterprise face ensuring that Azure workloads are backed up. In conventional IT support models where application development and IT operations are managed by separate teams, the ownership of VM backups can become unclear. Missing backup process can also create costly consequences during intended or unintended scenarios that need VM backups for restoring workloads.

  Azure Backup provides a seamless, simple, and integrated option for backing up VMs running in Azure or on-premises sites. Azure Backup uses cloud-scale storage and frees enterprises from constantly procuring and managing storage for backups. Azure Backup provides a repository to store at-rest and transitional data securely.

  Azure offers a custom policy that automatically configures Azure Backup protection for Windows and Linux VMs. An Azure Recovery Services vault is configured for storing backups, securely storing data and using **soft-delete** to protect backups from being deleted maliciously. A default backup policy is created and assigned with preconfigured values for backup schedules, retention periods, and more.

- **Provision connectivity between VNets:**

  It's common for enterprises to scatter workloads across multiple subscriptions or VNets. In turn, critical business applications can struggle with exchanging data without dedicated and secure network connectivity. An internet-based network connection will can offer inconsistent network bandwidth and performance, and potentially high network latency can negatively impact the user experience.

  Virtual network peering provides connects two VNets over the Microsoft backbone network. Azure peering supports high-bandwidth, low-latency network connectivity where data between separate Azure subscriptions, tenants, or Azure regions can be exchanged securely.

  Azure offers a custom policy provides with a template for setting up virtual network peering. A network layout in an Azure Resource Manager template definition can be passed as a parameter. This policy will spin up VNets and configure VNet peering between them, plus dependencies like NSGs, user-define routing, and others.

- **Enforce Windows VMs to join Azure AD domains:**

  Enterprises have been using domain-joined VMs for a consistent management experience. When

operations like corporate password policies, central authentication, and others are created as part of domain policies, a VM that doesn't join the domain is exposed to risks like weak passwords and an inability to connect with corporate devices, applications, and other limitations. When deployed on VMs that aren't joined to the domain, legacy applications relying on authentication protocols like NTLM or Kerberos could face authentication issues.

Azure provides managed and unmanaged solutions for implementing domain services. With self-managed Azure AD DS services, enterprises gain the same complete control of setups, configurations, and operations as with on-premises environment. The service frees enterprises of management overhead while providing essential domain services.

Azure offers a custom policy that helps new Windows VMs to automatically join domains. The `JsonADDomainExtension` extension deployed on the VM uses configuration settings like the username, domain, OU path, and other settings to ensure that the VM joins the specified domain. This policy uses Azure Key Vault to manage confidential domain username and password information.

Network topology and connectivity overview

3/6/2021 • 2 minutes to read • Edit Online

This series of articles examines key design considerations and best practices surrounding networking and connectivity to, from, and within Microsoft Azure.

Plan for IP addressing

It's vital that your organization plans for IP addressing in Azure to ensure that the IP address space doesn't overlap across on-premises locations and Azure regions. This section provides guidance on planning IP addressing for a hybrid implementation

Configure DNS and name resolution for on-premises and Azure resources

Domain Name System (DNS) is a critical design topic in the overall enterprise-scale architecture. Some organizations might want to use their existing investments in DNS. Others might see cloud adoption as an opportunity to modernize their internal DNS infrastructure and use native Azure capabilities. This section explores guidance on planning DNS and name resolution for hybrid implementations.

Define an Azure network topology

Network topology is a critical element of the enterprise-scale architecture because it defines how applications can communicate with each other. This section explores technologies and topology approaches for Azure deployments. It focuses on two core approaches: topologies based on Azure Virtual WAN, and traditional topologies.

Virtual WAN network topology

This section explores the option for implementing an Azure Virtual WAN network topology.

Traditional Azure networking topology

This section explores the option for implementing a traditional Azure networking topology.

Connectivity to Azure

This section expands on the network topology to consider recommended models for connecting on-premises locations to Azure.

Private Link and DNS integration at scale

This section describes how to integrate Azure Private Link for PaaS services with Azure Private DNS zones in hub and spoke network architectures.

Connectivity to Azure PaaS services

Building on the previous connectivity sections, this section explores recommended connectivity approaches for using Azure PaaS services.

Plan for inbound and outbound internet connectivity

This section describes recommended connectivity models for inbound and outbound connectivity to and from the public internet.

Plan for application delivery

This section explores key recommendations to deliver internal-facing and external-facing applications in a secure, highly scalable, and highly available way.

Plan for landing zone network segmentation

This section explores key recommendations to deliver highly secure internal network segmentation within a landing zone to drive a network zero-trust implementation.

Define network encryption requirements

This section explores key recommendations to achieve network encryption between on-premises and Azure as well as across Azure regions.

Plan for traffic inspection

In many industries, organizations require that traffic in Azure is mirrored to a network packet collector for deep inspection and analysis. This requirement typically focuses on inbound and outbound internet traffic. This section explores key considerations and recommended approaches for mirroring or tapping traffic within Azure Virtual Network.

Connectivity to other cloud providers

This section provides different connectivity approaches to integrate an Azure enterprise-scale landing zone architecture to other cloud providers.

Plan for IP addressing

3/6/2021 • 2 minutes to read • Edit Online

It's vital that your organization plans for IP addressing in Azure to ensure that the IP address space doesn't overlap across on-premises locations and Azure regions.

Design considerations:

- Overlapping IP address spaces across on-premises and Azure regions will create major contention challenges.
- You can add address space after you create a virtual network. This process requires an outage if the virtual network is already connected to another virtual network via virtual network peering because the peering must be deleted and re-created.
- Azure reserves five IP addresses within each subnet. Factor in those addresses when you're sizing virtual networks and encompassed subnets.
- Some Azure services require dedicated subnets. These services include Azure Firewall and Azure VPN Gateway.
- You can delegate subnets to certain services to create instances of a service within the subnet.

Design recommendations:

- Plan for non-overlapping IP address spaces across Azure regions and on-premises locations well in advance.
- Use IP addresses from the address allocation for private internets (RFC 1918).
- For environments that have limited availability of private IP addresses (RFC 1918), consider using IPv6.
- Don't create unnecessarily large virtual networks (for example, `/16`) to ensure that IP address space isn't wasted.
- Don't create virtual networks without planning the required address space in advance. Adding address space will cause an outage after a virtual network is connected via virtual network peering.
- Don't use public IP addresses for virtual networks, especially if the public IP addresses don't belong to your organization.

DNS for on-premises and Azure resources

6/18/2021 • 2 minutes to read • Edit Online

Domain Name System (DNS) is a critical design topic in the overall enterprise-scale architecture. Some organizations might want to use their existing investments in DNS. Others might see cloud adoption as an opportunity to modernize their internal DNS infrastructure and use native Azure capabilities.

Design considerations:

- You can use a DNS resolver in conjunction with Azure Private DNS for cross-premises name resolution.
- You might require the use of existing DNS solutions across on-premises and Azure.
- The maximum number of private DNS zones to which a virtual network can link with auto-registration is one.
- Familiarize yourself with Azure Private DNS zone limits.

Design recommendations:

- For environments where name resolution in Azure is all that's required, use Azure Private DNS zones for resolution. Create a delegated zone for name resolution (such as `azure.contoso.com`). Enable auto-registration for Azure Private DNS to automatically manage the lifecycle of the DNS records for the virtual machines deployed within a virtual network.
- For environments where name resolution across Azure and on-premises is required, use existing DNS infrastructure (for example, Active Directory integrated DNS) deployed onto at least two virtual machines (VMs). Configure DNS settings in virtual networks to use those DNS servers.
- For environments with Azure Firewall, evaluate using it as DNS proxy.
- Use a virtual machine as a resolver for cross-premises DNS resolution with Azure Private DNS.
- You can still link an Azure Private DNS zones to the virtual networks and use DNS servers as hybrid resolvers with conditional forwarding to on-premises DNS names, such as `corporate.contoso.com`, by using on-premises DNS servers. You can configure on-premises servers with conditional forwarders to resolver VMs in Azure for the Azure Private DNS zone (for example, `azure.contoso.com`).
- Special workloads that require and deploy their own DNS (such as Red Hat OpenShift) should use their preferred DNS solution.
- Create the Azure Private DNS zones within a global connectivity subscription. The Azure Private DNS zones that should be created include the zones required for accessing Azure PaaS services via a private endpoint (for example, `privatelink.database.windows.net` or `privatelink.blob.core.windows.net`).

Private Link and DNS integration at scale

6/7/2021 • 11 minutes to read • Edit Online

This article describes how to integrate Azure Private Link for PaaS services with Azure Private DNS zones in hub and spoke network architectures.

Introduction

Many customers build their network infrastructure in Azure using the hub and spoke network architecture, where:

- Networking shared services (such as network virtual appliances, ExpressRoute/VPN gateways, or DNS servers) are deployed in the **hub** virtual network (VNet)
- **Spoke** VNets consume those shared services via VNet peering.

In hub and spoke network architectures, application owners are typically provided with an Azure subscription, which includes a VNet (*a spoke*) connected to the *hub* VNet. In this architecture, they can deploy their virtual machines and have private connectivity to other VNets or to on-premises networks via ExpressRoute or VPN.

Internet-outbound connectivity is provided via a central network virtual appliance (NVA) such as Azure Firewall.

Many application teams build their solutions using a combination of Azure IaaS and PaaS resources. Some Azure PaaS services (such as SQL Managed Instance) can be deployed in customer VNets. As a result, traffic will stay private within the Azure network and would be fully routable from on-premises.

However, some Azure PaaS services (such as Azure Storage or Azure Cosmos DB) cannot be deployed in a customer's VNets and are accessible over their public endpoint. In some instances, this can cause a contention with a customer's security policies, as corporate traffic might not allow the deployment or accessing of corporate resources (such as a SQL database) over public endpoints.

Azure Private Link allows access to a list of Azure services over private endpoints, but it requires that those private endpoint records are registered in a corresponding private DNS zone.

This article describes how application teams can deploy Azure PaaS services in their subscriptions which are only accessible over private endpoints.

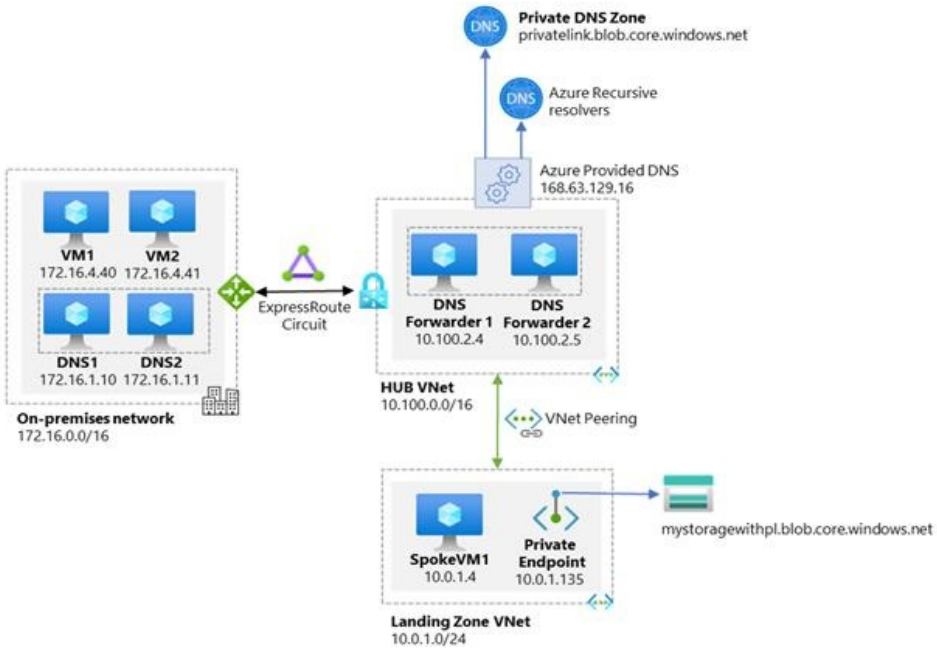
This article will also describe how application teams can ensure that services are automatically integrated with private DNS zones via Azure Private DNS. Removing the need for manual creation or deletion of records in DNS.

Private Link and DNS integration in hub and spoke network architectures

Private DNS zones are typically hosted centrally in the same Azure subscription where the hub VNet is deployed. This central hosting practice is driven by cross-premises DNS name resolution and other needs for central DNS resolution such as Active Directory. In most cases, only networking/identity admins have permissions to manage DNS records in these zones.

Application teams do have permissions to create Azure resource in their own subscription. They do not have any permissions in the central networking connectivity subscription, which includes managing DNS records in the private DNS zones. This access limitation means they do not have the possibility to create the DNS records required when deploying Azure PaaS services with private endpoints.

The following diagram shows a typical high-level architecture for enterprise environments with central DNS resolution and where name resolution for Private Link resources is done via Azure Private DNS:



From the previous diagram, it is important to highlight that:

- On-premises DNS servers have conditional forwarders configured for each private endpoint public DNS zone, pointing to the DNS servers 10.100.2.4 and 10.100.2.5 hosted in the hub VNet.
- The DNS servers 10.100.2.4 and 10.100.2.5 hosted in the hub VNet use the Azure-provided DNS resolver (168.63.129.16) as a forwarder.
- The hub VNet must be linked to the Private DNS zone names for Azure services (such as privatelink.blob.core.windows.net, as shown in the picture).
- All Azure VNets use the DNS servers hosted in the hub VNet (10.100.2.4 and 10.100.2.5) as the primary and secondary DNS servers.
- If the DNS servers 10.100.2.4 and 10.100.2.5 are not authoritative for customer's corporate domains (for example, Active Directory domain names), they should have conditional forwarders for the customer's corporate domains, pointing to the on-premises DNS Servers (172.16.1.10 & 172.16.1.11) or DNS servers deployed in Azure which are authoritative for such zones.

Please note that while the previous diagram depicts a single hub and spoke architecture, this guidance is applicable to scenarios where there are multiple hub and spoke networks across multiple Azure regions. In this case, the hub VNets in all regions should be linked to the same Azure Private DNS zones.

There are two conditions that must be true to allow application teams the freedom to create any required Azure PaaS resources in their subscription:

- Central networking and/or central platform team must ensure that application teams can only deploy and access Azure PaaS services via private endpoints.
- Central networking and/or central platform teams must ensure that whenever private endpoints are created, the corresponding records are automatically created in the centralized private DNS zone that matches the service created.
  - DNS record needs to follow the lifecycle of the private endpoint and automatically remove the DNS record when the private endpoint is deleted.

The following sections describe how application teams can enable these conditions by using Azure Policy. We will use Azure Storage as the Azure service that application teams need to deploy in our example below, but the

same principle can be applied to most Azure services that support Private Link.

Configuration required by platform team

Create private DNS zones

Create private DNS zones in the central connectivity subscription for the supported Private Link services as per documentation.

In our case, as we will use **Storage account with blob** as our example, it translates to us creating a `privatelink.blob.core.windows.net` private DNS zone in the connectivity subscription.

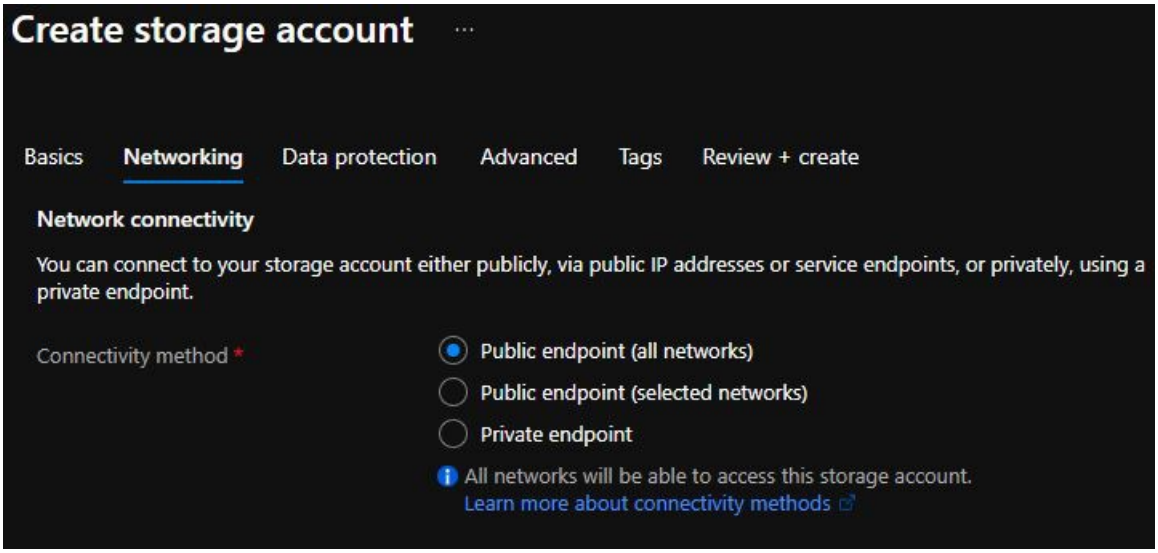


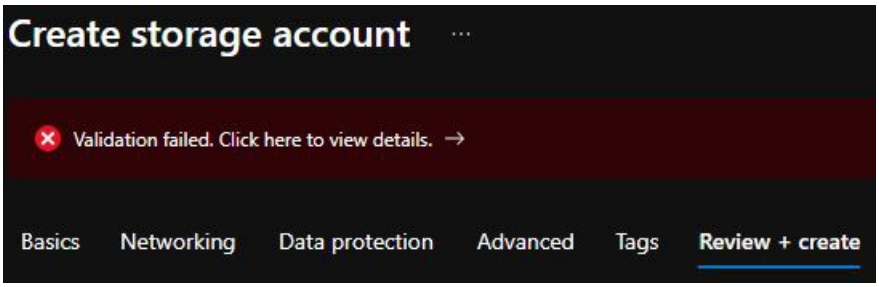
Policy definitions

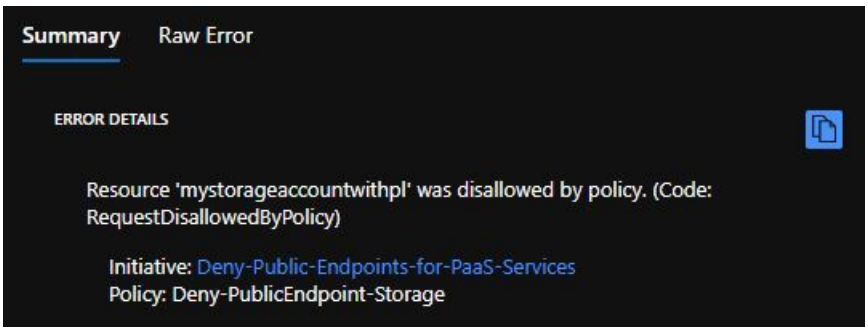
In addition to the private DNS zones, we also need to create a set of custom Azure Policy definitions to enforce the use of private endpoints and automate the DNS record creation in the DNS zone we created:

1. `Deny` public endpoint for PaaS services policy.

   This policy will prevent users from creating Azure PaaS services with public endpoints and give them an error message if private endpoint is not selected at resource creation.







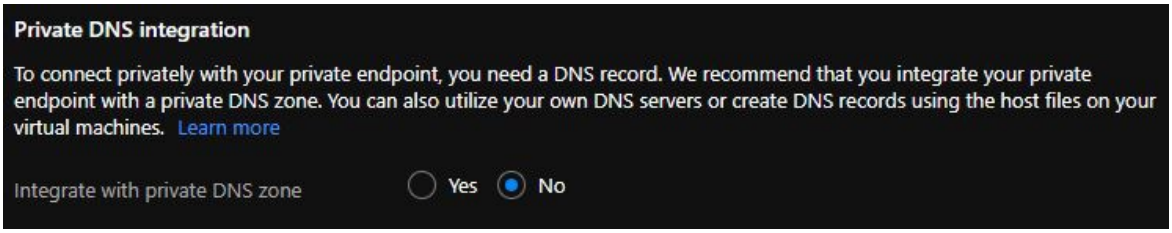
The exact policy rule may differ between PaaS services. For Azure Storage accounts, we look at the **networkAcls.defaultAction** property that defines whether requests from public network are allowed or not. In our case, we will set a condition to deny the creation of the **Microsoft.Storage/storageAccounts** resource type if the property **networkAcls.defaultAction** is not `Deny`. This policy definition is listed below:

```
{
  "mode": "All",
  "policyRule": {
    "if": {
      "allOf": [
        {
          "field": "type",
          "equals": "Microsoft.Storage/storageAccounts"
        },
        {
          "field": "Microsoft.Storage/storageAccounts/networkAcls.defaultAction",
          "notequals": "Deny"
        }
      ]
    },
    "then": {
      "effect": "Deny"
    }
  }
}
```

2. `Deny` creation of a private DNS zone with the `privatelink` prefix policy.

   Since we use a centralized DNS architecture with a conditional forwarder and private DNS zones hosted in the subscriptions managed by the platform team, we need to prevent the application teams owners from creating their own Private Link private DNS zones and linking services into their subscriptions.

   To accomplish this, we need to ensure that when application teams create a private endpoint, the option to `Integrate with private DNS zone` must be set to `No` when using the Azure portal.



   If `Yes` is selected, Azure Policy will prevent the creation of the private endpoint. In our policy definition, we will deny the creation of the **Microsoft.Network/privateDnsZones** resource type if the zone has the `privatelink` prefix. This policy definition is described below:

```
{
  "description": "This policy restricts creation of private DNS zones with the `privatelink` prefix",
  "displayName": "Deny-PrivateDNSZone-PrivateLink",
  "mode": "All",
  "parameters": null,
  "policyRule": {
    "if": {
      "allOf": [
        {
          "field": "type",
          "equals": "Microsoft.Network/privateDnsZones"
        },
        {
          "field": "name",
          "like": "privatelink*"
        }
      ]
    },
    "then": {
      "effect": "Deny"
    }
  }
}
```

3. `DeployIfNotExists` policy to automatically create the required DNS record in the central private DNS zone.

   This policy will be triggered if a private endpoint resource is created with a service-specific `groupId`. The `groupId` is the ID of the group obtained from the remote resource (service) that this private endpoint should connect to. We then trigger a deployment of a `privateDNSZoneGroup` within the private endpoint, which is used to associate the private endpoint with our private DNS zone. For our example, the `groupId` for Azure Storage blobs is `blob` (`groupId` for other Azure services can be found on this article, under the **Subresource** column). When policy finds that `groupId` in the private endpoint created, it will deploy a `privateDNSZoneGroup` within the private endpoint, and it will be linked to the private DNS zone resource ID that is specified as parameter. For our example, the private DNS zone resource ID would be:

   ```
   /subscriptions/<subscription-id>/resourceGroups/<resourceGroupName>/providers/Microsoft.Network/privateDnsZones/privatelink.blob.core.windows.net
   ```

   This policy definition is listed below:

   ```
   {
     "mode": "Indexed",
     "policyRule": {
       "if": {
         "allOf": [
           {
             "field": "type",
             "equals": "Microsoft.Network/privateEndpoints"
           },
           {
             "count": {
               "field":
   "Microsoft.Network/privateEndpoints/privateLinkServiceConnections[*].groupIds[*]",
               "where": {
                 "field":
   "Microsoft.Network/privateEndpoints/privateLinkServiceConnections[*].groupIds[*]",
                 "equals": "blob"
               }
             },
             "greaterOrEquals": 1
           }
         ]
   ```

```
    },
    "then": {
      "effect": "deployIfNotExists",
      "details": {
        "type": "Microsoft.Network/privateEndpoints/privateDnsZoneGroups",
        "roleDefinitionIds": [
          "/providers/Microsoft.Authorization/roleDefinitions/b24988ac-6180-42a0-ab88-20f7382dd24c"
        ],
        "existenceCondition": {
          "allOf": [
            {
              "field":
"Microsoft.Network/privateEndpoints/privateDnsZoneGroups/privateDnsZoneConfigs[*].privateDnsZoneId",
              "equals": "[parameters('privateDnsZoneId')]"
            }
          ]
        },
        "deployment": {
          "properties": {
            "mode": "incremental",
            "template": {
              "$schema": "https://schema.management.azure.com/schemas/2019-04-
01/deploymentTemplate.json#",
              "contentVersion": "1.0.0.0",
              "parameters": {
                "privateDnsZoneId": {
                  "type": "string"
                },
                "privateEndpointName": {
                  "type": "string"
                },
                "location": {
                  "type": "string"
                }
              },
              "resources": [
                {
                  "name": "[concat(parameters('privateEndpointName'), '/deployedByPolicy')]",
                  "type": "Microsoft.Network/privateEndpoints/privateDnsZoneGroups",
                  "apiVersion": "2020-03-01",
                  "location": "[parameters('location')]",
                  "properties": {
                    "privateDnsZoneConfigs": [
                      {
                        "name": "storageBlob-privateDnsZone",
                        "properties": {
                          "privateDnsZoneId": "[parameters('privateDnsZoneId')]"
                        }
                      }
                    ]
                  }
                }
              ]
            },
            "parameters": {
              "privateDnsZoneId": {
                "value": "[parameters('privateDnsZoneId')]"
              },
              "privateEndpointName": {
                "value": "[field('name')]"
              },
              "location": {
                "value": "[field('location')]"
              }
            }
          }
        }
      }
    }
```

```
    },
    "parameters": {
      "privateDnsZoneId": {
        "type": "String",
        "metadata": {
          "displayName": "privateDnsZoneId",
          "strongType": "Microsoft.Network/privateDnsZones"
        }
      }
    }
  }
```

Policy assignments

Once policy definitions have been deployed, assign the policies at the desired scope in your management group hierarchy. Ensure that the policy assignments target the Azure subscriptions that will be used by the application teams to deploy PaaS services with private endpoint access exclusively.

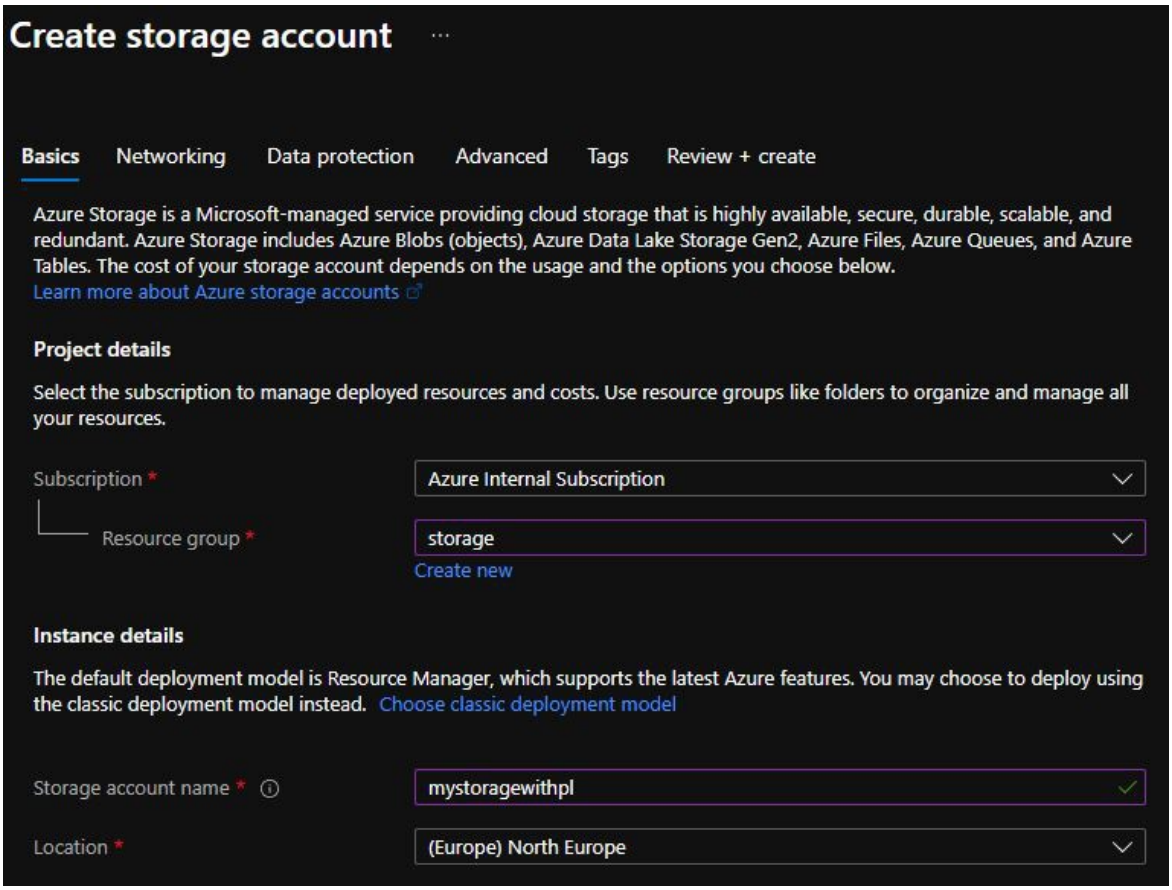
> **IMPORTANT**
>
> Remember to assign the Private DNS Zone Contributor role role in the subscription/resource group where the private DNS zones are hosted to the managed identity created by the `DeployIfNotExists` policy assignment that will be responsible to create and manage the private endpoint DNS record in the private DNS zone. This is because the private endpoint is located in the application owner Azure subscription, while the private DNS zone is located in a different subscription (such as central connectivity subscription).

Once the platform team finishes this configuration, Azure subscriptions from applications teams are ready for them to create Azure PaaS services with private endpoint access exclusively, and ensuring the DNS records for private endpoints are automatically registered (and removed once private endpoint is deleted) from corresponding private DNS zones.

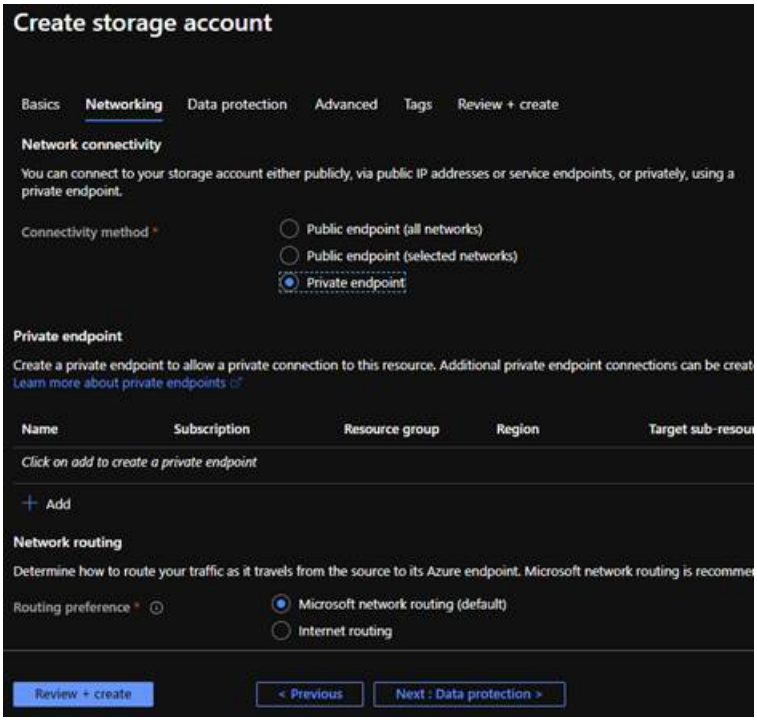
Application owner experience

Once the platform team has deployed the platform infrastructure components (private DNS zones and policies) described in the previous section, when an application owner tries to deploy an Azure PaaS service into the Azure subscription, the application owner will have the following experience, which will be the same if they do activities via the Azure portal or via other clients such as PowerShell or CLI, as their subscriptions are being governed by Azure policies.

1. Create a storage account through the Azure portal. In the basic tab, provide a name and your desired settings and click on **Next**.

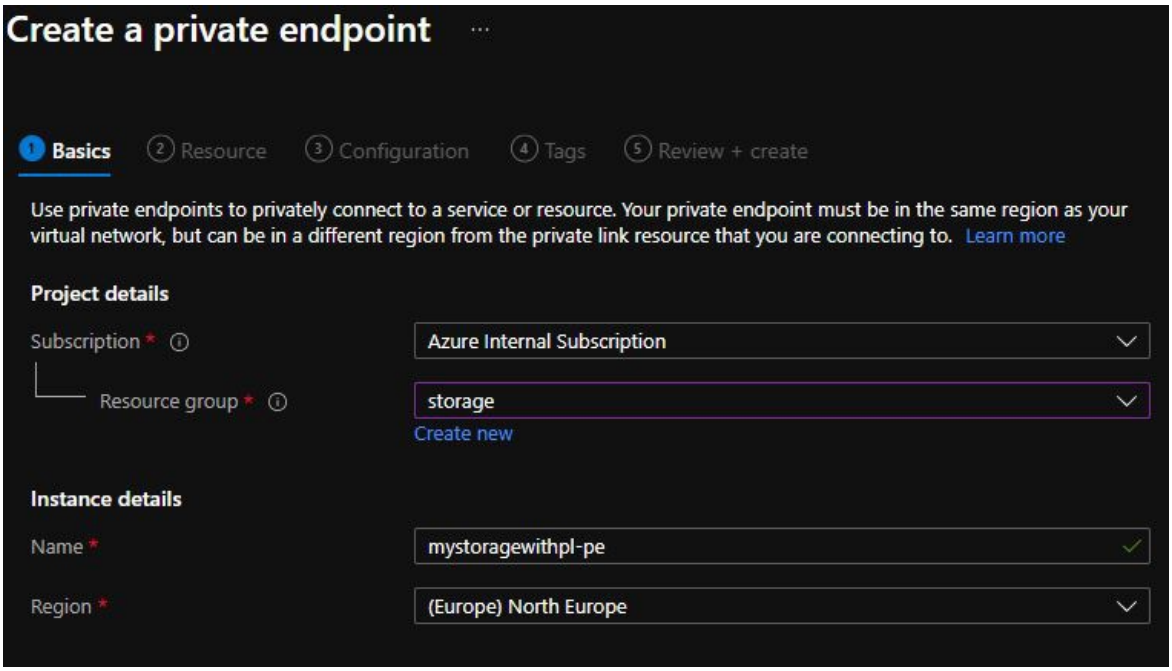


2. In the networking section, ensure **Private endpoint** is selected. If an option other than **Private endpoint** is selected, the Azure portal will not allow the creation of the storage account in the **Review + create** section of the deployment wizard, as policy is preventing the creation of this service if the public endpoint is enabled.

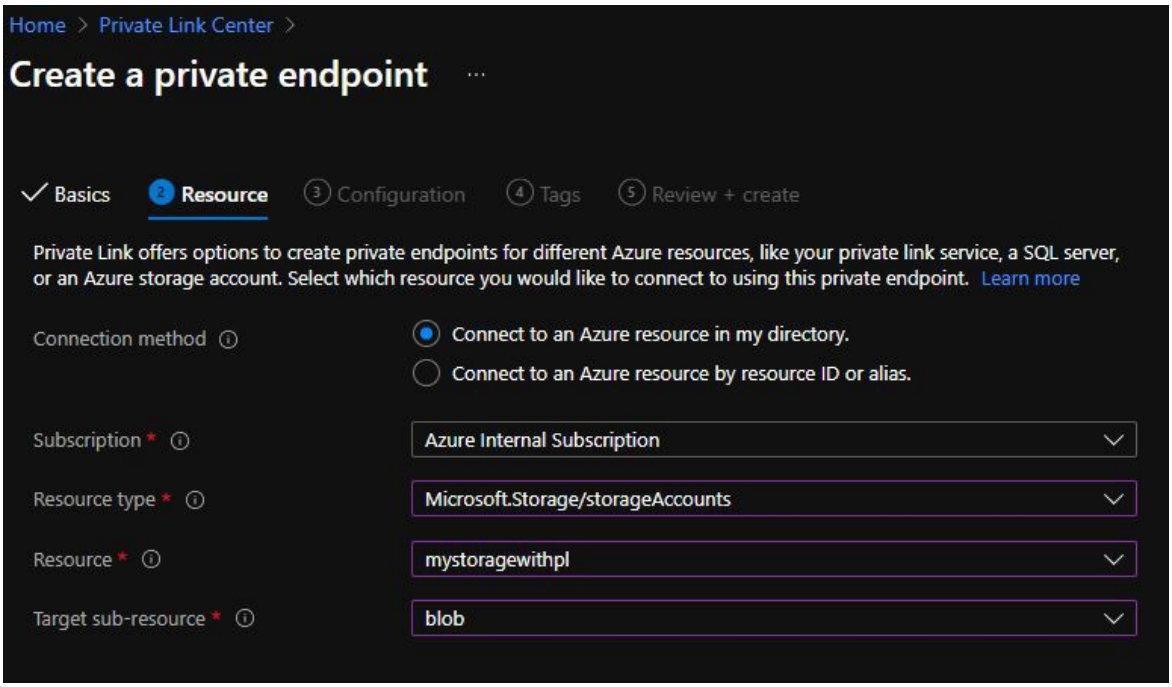


3. It is possible to create the private endpoint on this screen or it can be done after the storage account has been created. For this exercise, we will create the private endpoint after the storage account has been created. Click on **Review + create** and complete the creation of the storage account.

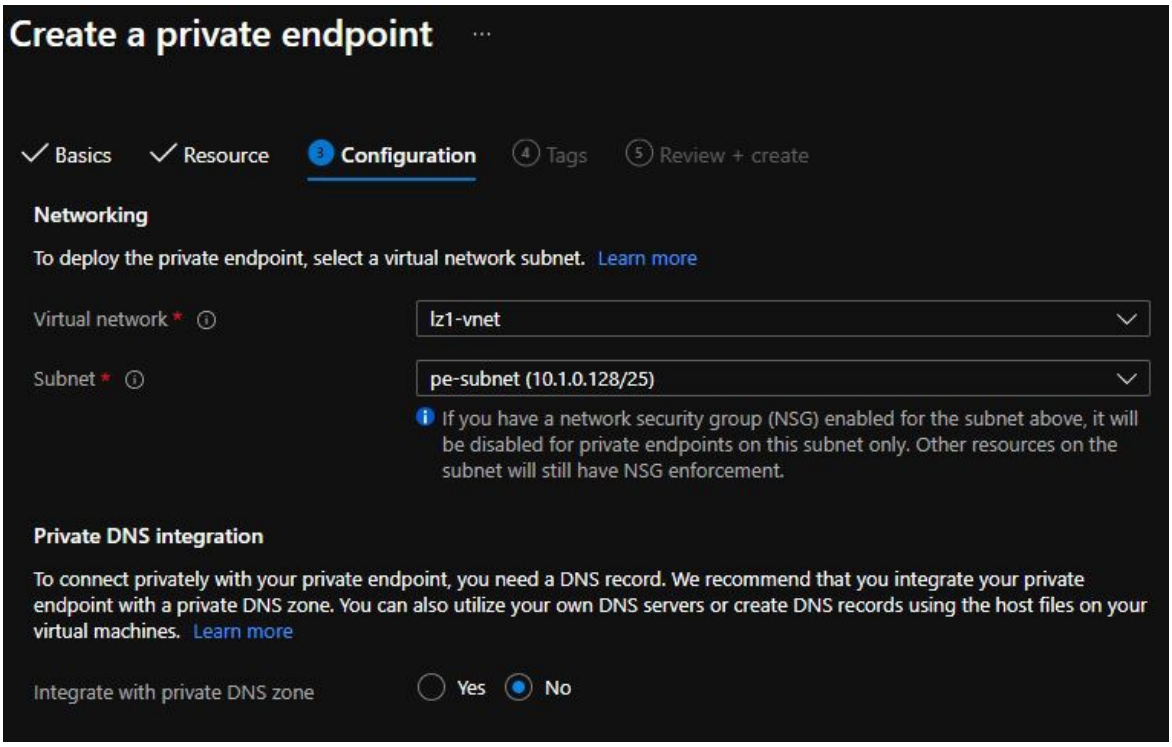
4. Once the storage account is created, create a private endpoint through the Azure portal



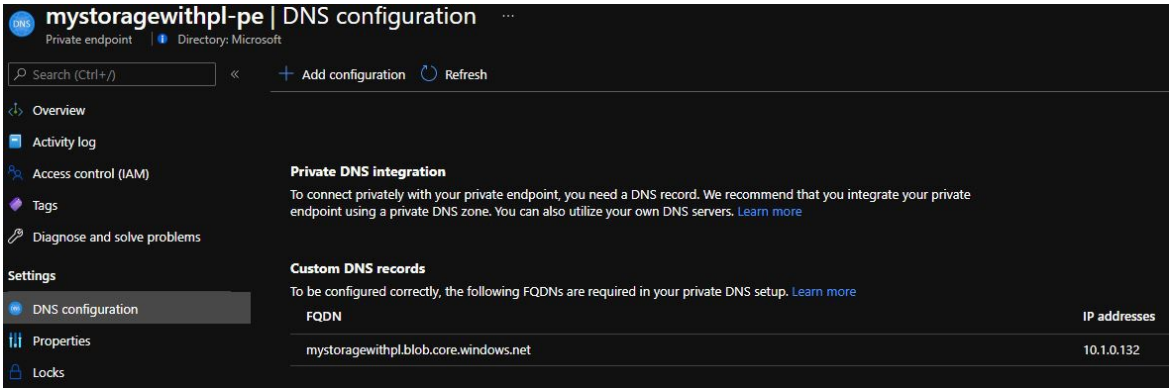
5. In the **Resource** section, locate the storage account created in the previous step, and for target subresource select **Blob**, and then select **Next**.



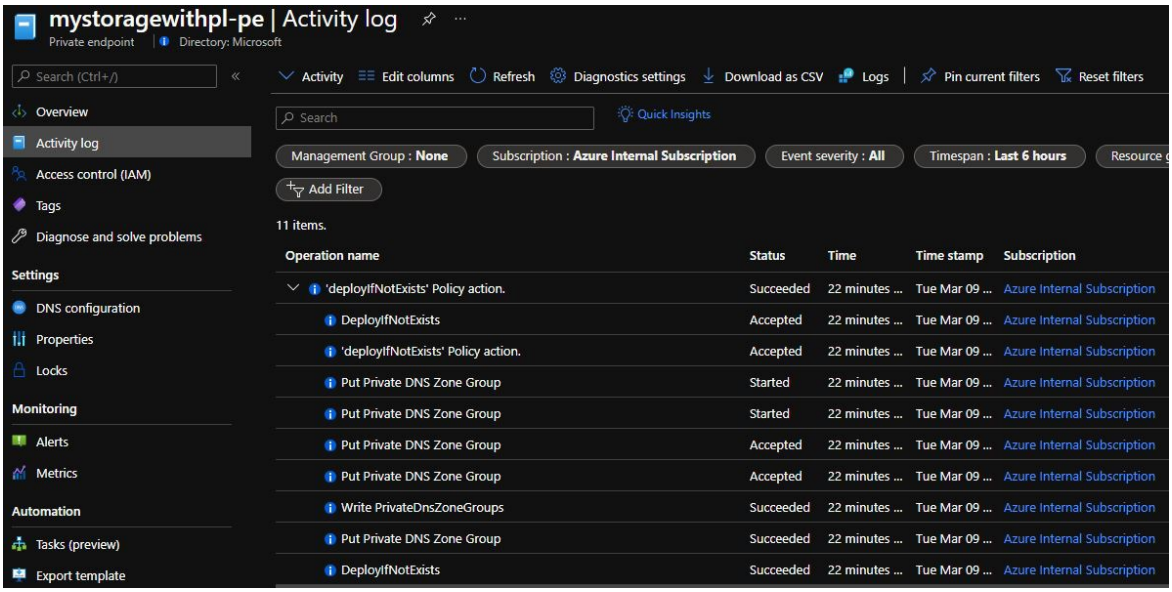
6. In the **Configuration** section, after selecting your VNet and subnet, ensure that **Integrate with private DNS zone** is set to **No**. Otherwise, the Azure portal will prevent the creation of the private endpoint, as Azure Policy will not allow the creation of a private DNS zone with the `privatelink` prefix.



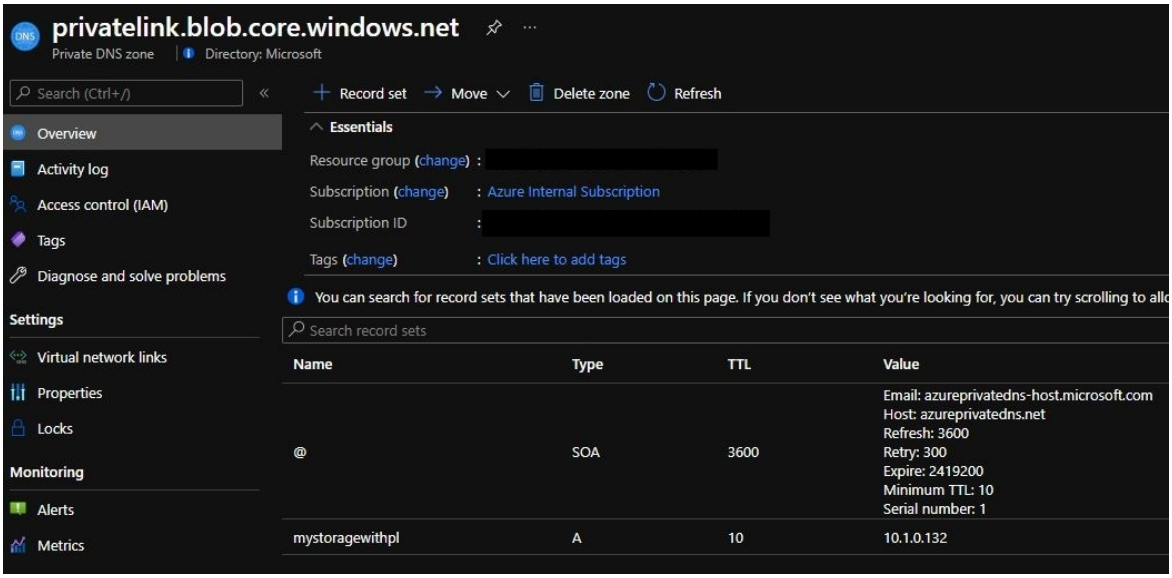
7. Select **Review + create**, and then select **Create** to deploy the private endpoint.
8. After a few minutes, the `DeployIfNotExists` policy will be triggered and the subsequent `dnsZoneGroup` deployment will add the required DNS records for the private endpoint in the centrally managed DNS zone.
9. Once the private endpoint has been created, select it, and review its FQDN and private IP:



10. Check the activity log for the resource group where the private endpoint was created, or you can check the activity log of the private endpoint itself. You will notice that after a few minutes, a `DeployIfNotExist` policy action is executed, which configures the DNS zone group on the private endpoint:



11. If the central networking team goes to the `privatelink.blob.core.windows.net` private DNS zone, they will confirm that the DNS record has been created for the private endpoint we created, and both the name and IP address match to the values within the private endpoint.



At this point, application teams can use the storage account via a private endpoint from any VNet in the hub and spoke network environment and from on-premises, as the DNS record has been automatically recorded in the private DNS zone.

If an application owner deletes the private endpoint, the corresponding records in the private DNS zone will automatically be removed.

Define an Azure network topology

3/6/2021 • 2 minutes to read • Edit Online

Network topology is a critical element of the enterprise-scale architecture because it defines how applications can communicate with each other. This section explores technologies and topology approaches for Azure deployments. It focuses on two core approaches: topologies based on Azure Virtual WAN, and traditional topologies.

Virtual WAN is used to meet large-scale interconnectivity requirements. Because it is a Microsoft-managed service, it also reduces overall network complexity and helps to modernize your organization's network. A Virtual WAN topology may be most appropriate if any of the following points meet your requirements:

- Your organization intends to deploy resources across several Azure regions and requires global connectivity between VNets in these Azure regions and multiple on-premises locations.
- Your organization intends to integrate a large-scale branch network directly in to Azure, either via a software-defined WAN (SD-WAN) deployment or requires more than 30 branch sites for native IPsec termination.
- You require transitive routing between VPN and ExpressRoute. E.g. Remote branches connected via Site-to-site VPN or remote users connected via Point-to-site VPN, require connectivity to an ExpressRoute connected DC, via Azure.

A traditional hub-and-spoke network topology helps you build customized secure large-scale networks in Azure with routing and security managed by the customer. A traditional topology may be most appropriate if any of the following points meet your requirements:

- Your organization intends to deploy resources across one or several Azure regions and while some traffic across Azure regions is expected (for example, traffic between two virtual networks across two different Azure regions), a full mesh network across all Azure regions is not required.
- You have a low number of remote or branch locations per region. That is, you need fewer than 30 IPsec Site-to-Site tunnels.
- You require full control and granularity for manually configuring your Azure network routing policy.

Virtual WAN network topology (Microsoft-managed)

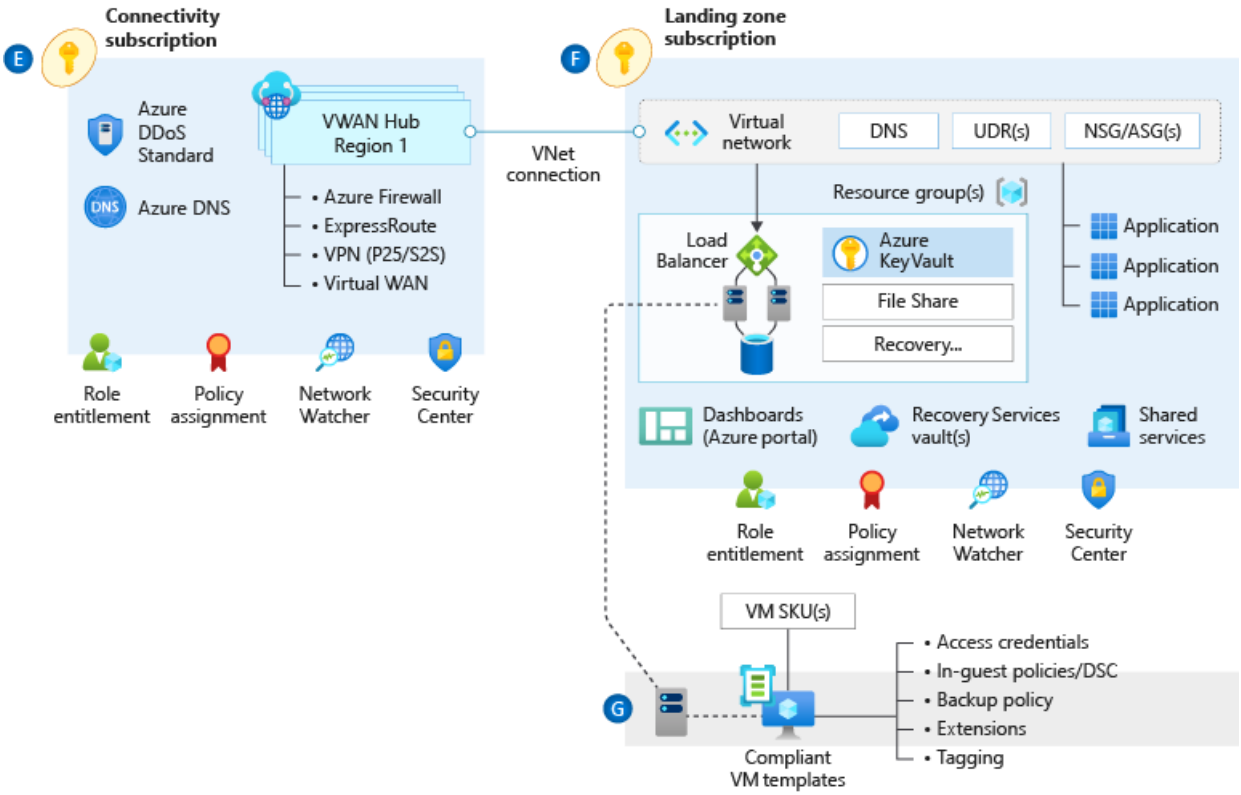


Figure 1: Virtual WAN network topology.

Traditional Azure networking topology

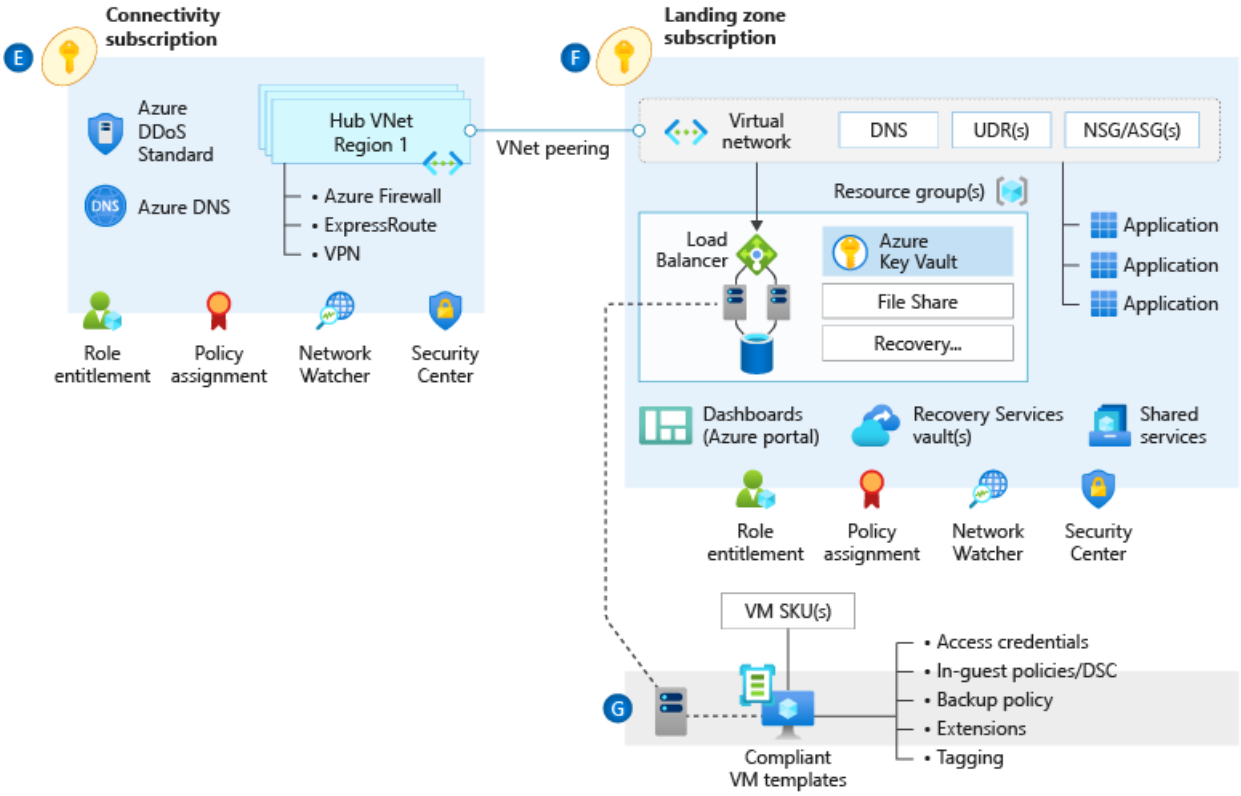


Figure 2: A traditional Azure network topology.

Virtual WAN network topology (Microsoft-managed)

7/9/2021 • 6 minutes to read • Edit Online

Explore key design considerations and recommendations surrounding virtual wide area networks (Virtual WAN) in Microsoft Azure.

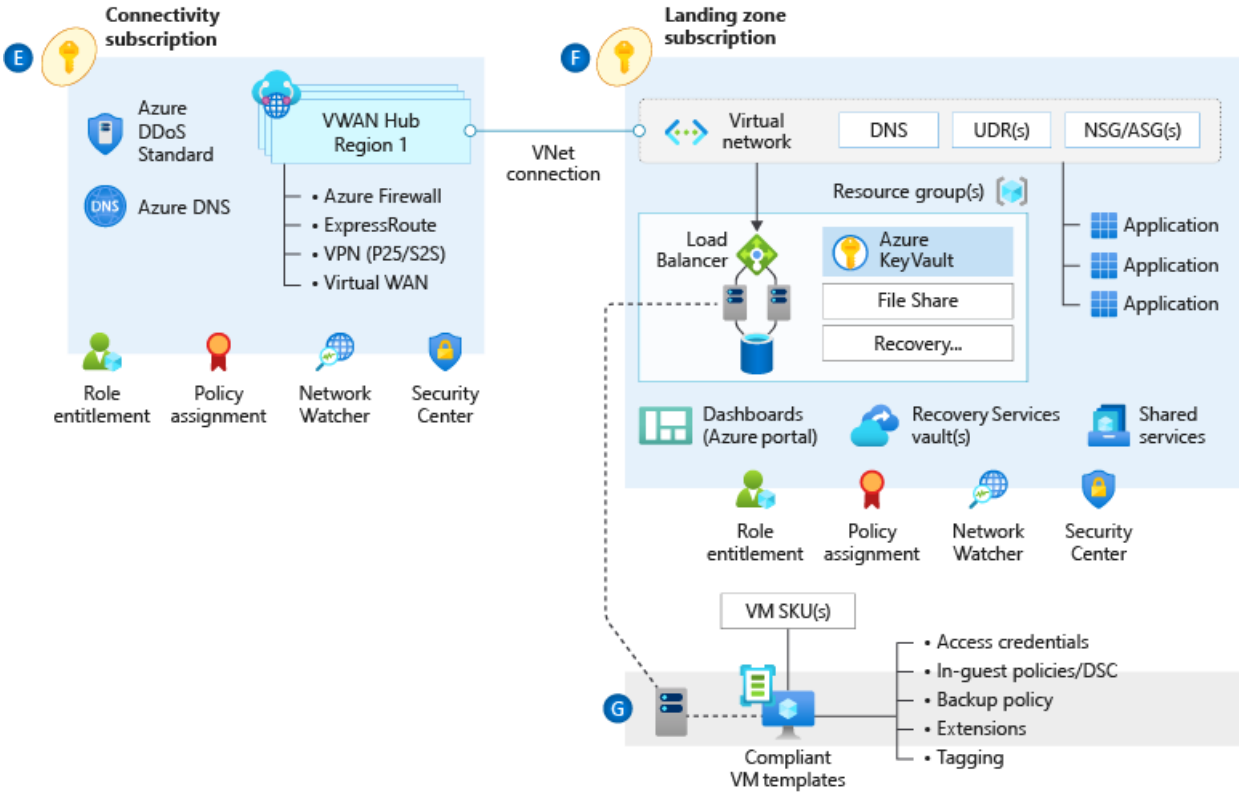


Figure 1: Virtual WAN network topology.

Design considerations:

- Azure Virtual WAN is a Microsoft-managed solution that provides end-to-end, global, and dynamic transit connectivity by default. Virtual WAN hubs eliminate the need to manually configure network connectivity. For example, you don't need to manage user-defined routes (UDR) or network virtual appliances (NVAs) to enable global transit connectivity.
- Virtual WAN simplifies end-to-end network connectivity in Azure, and to Azure from on-premises, by creating a hub-and-spoke network architecture. The architecture can easily scale to support multiple Azure regions and on-premises locations (any-to-any connectivity) as shown in the following figure:

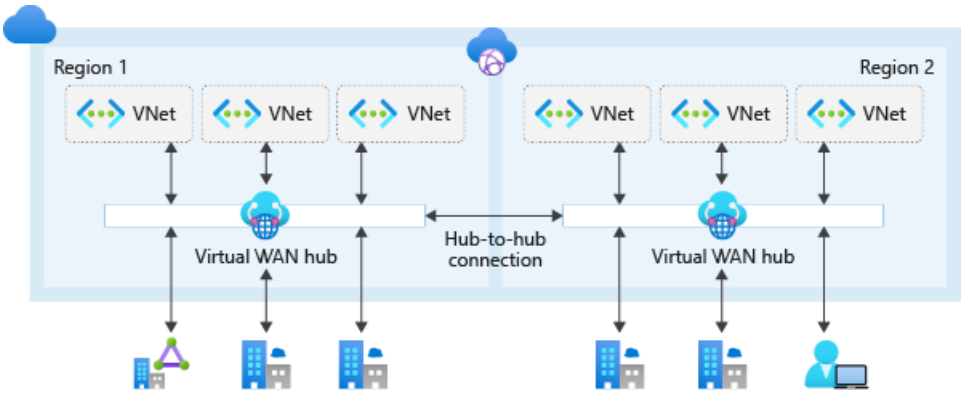


Figure 2: Global transit network with Virtual WAN.

- Virtual WAN any-to-any transitive connectivity supports the following paths (within the same region and across regions):
  - Virtual network to virtual network
  - Virtual network to branch
  - Branch to virtual network
  - Branch to branch
- Virtual WAN hubs are restricted to the deployment of Microsoft managed resources. The only resources that you can deploy within them are virtual network gateways (Point-to-Site VPN, Site-to-Site VPN, and Azure ExpressRoute), Azure Firewall via Firewall Manager, route tables and some network virtual appliances (NVA) for vendor-specific SD-WAN capabilities.
- Virtual WAN is bound to some Azure subscription limits for Virtual WAN.
- Network-to-network transitive connectivity (within a region and across regions via hub-to-hub) is now in general availability (GA).
- Transit connectivity between the virtual networks in Standard Virtual WAN is enabled due to the presence of a Microsoft-managed routing function in every virtual hub. Each hub supports an aggregate throughput of up to 50 Gbps for VNet-to-VNet traffic.
- A single Azure Virtual WAN hub can support a specific maximum number of VM workloads across all directly attached VNets, which are documented in Azure Virtual WAN limits.
- Virtual WAN integrates with various SD-WAN providers.
- Many managed service providers offer managed services for Virtual WAN.
- User VPN (Point-to-Site) gateways in Virtual WAN can scale up to 20-Gbps aggregated throughput and 10,000 client connections per virtual hub.
- Site-to-site VPN gateways in Virtual WAN can scale up to 20-Gbps aggregated throughput.
- ExpressRoute circuits using a Local, Standard, or Premium SKU can be connected to a Virtual WAN hub.
- ExpressRoute Standard or Premium circuits that are in locations supported by ExpressRoute Global Reach can connect to a Virtual WAN ExpressRoute gateway and enjoy all Virtual WAN transit capabilities (VPN-to-VPN, VPN, and ExpressRoute transit). ExpressRoute Standard or Premium circuits that are in locations not supported by Global Reach can connect to Azure resources, but cannot use Virtual WAN transit capabilities.
- ExpressRoute Local is supported with Azure Virtual WAN hubs if the spoke VNets connected to a Virtual WAN hub are in the same region as the Virtual WAN hub.
- Azure Firewall Manager, now generally available, allows the deployment of Azure Firewall in the Virtual

WAN hub, known as secured virtual hub. Review the Azure Firewall Manager overview of secured virtual hubs and the latest constraints.

- Virtual WAN hub-to-hub traffic via Azure Firewall is currently not supported when the Azure Firewall is deployed inside of the Virtual WAN hub itself (secured virtual hub). Several workarounds exist depending on your requirements, including placing the Azure Firewall in a spoke virtual network, or using NSGs for traffic filtering.
- The Virtual WAN portal experience requires that all Virtual WAN resources are deployed into the same resource group together.
- An Azure DDoS Protection standard protection plan can be shared across all VNets in a single Azure AD Tenant to protect resources with public IP addresses. For more information, see Azure DDoS Protection Standard.
    - At this time, Virtual WAN Secure Virtual Hubs don't support Azure DDoS standard protection plans. For more information, see Azure Firewall Manager known issues and hub virtual network and secured virtual hub comparison.
    - Only resources with Public IP addresses are covered by Azure DDoS Protection standard protection plans.
        - 100 public IP addresses are included in the cost of an Azure DDoS Protection standard protection plan across all protected VNets associated to the DDoS Protection plan. Any other public IP addresses over the 100 included with the plan, are charged separately. For more information on Azure DDoS Protection Standard Protection pricing, see the pricing page or the FAQ.
    - Review the supported resources of Azure DDoS Protection standard protection plans

Design recommendations:

- We recommend Virtual WAN for new large or global network deployments in Azure where you need global transit connectivity across Azure regions and on-premises locations. That way, you don't have to manually set up transitive routing for Azure networking.

  The following figure shows a sample global enterprise deployment with datacenters spread across Europe and the United States. The deployment also has many branch offices within both regions. The environment is globally connected via Azure Virtual WAN and ExpressRoute Global Reach.

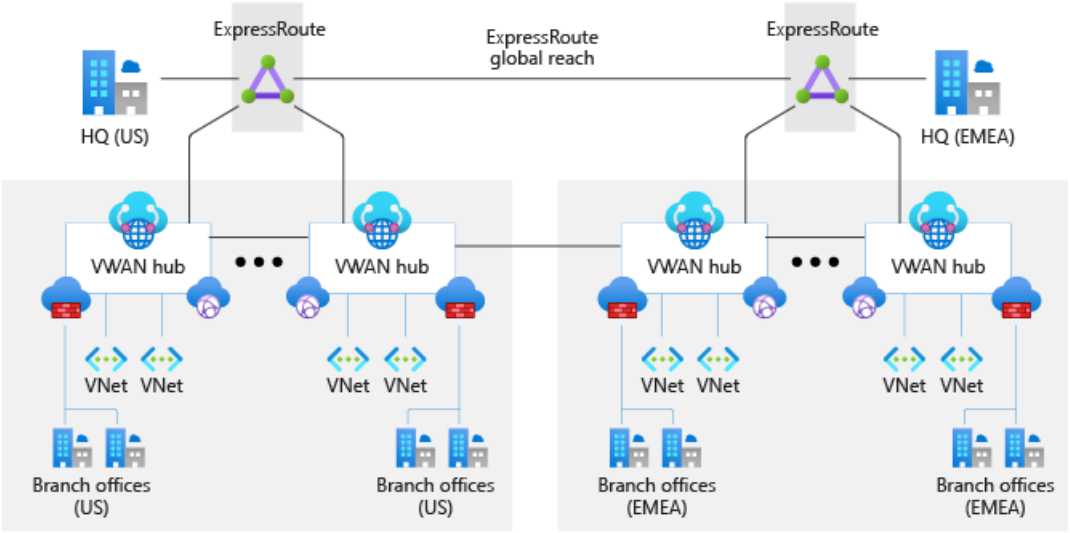


Figure 3: Sample network topology.

- Use a Virtual WAN hub per Azure region to connect multiple landing zones together across Azure regions

via a common global Azure Virtual WAN.

- Deploy all Virtual WAN resources into a single resource group in the connectivity subscription, including when you are deploying across multiple regions.
- Use virtual hub routing features to further segment traffic between VNets and branches.
- Connect Virtual WAN hubs to on-premises datacenters by using ExpressRoute.
- Deploy required shared services, like DNS servers, in a dedicated spoke virtual network. Customer deployed shared resources cannot be deployed inside the Virtual WAN hub itself.
- Connect branches and remote locations to the nearest Virtual WAN hub via Site-to-Site VPN, or enable branch connectivity to Virtual WAN via an SD-WAN partner solution.
- Connect users to the Virtual WAN hub via a Point-to-Site VPN.
- Follow the principle of "traffic in Azure stays in Azure" so that communication across resources in Azure occurs via the Microsoft backbone network, even when the resources are in different regions.
- For internet outbound protection and filtering, consider deploying Azure Firewall in the virtual hub.
- If partner NVAs are required for east/west or south/north traffic protection and filtering, as Azure Virtual WAN doesn't allow deploying such security NVAs in the virtual hub, evaluate whether deploying those NVAs to a separate spoke virtual network and configuring static routing as described in Scenario: Route traffic through an NVA will meet your requirements. As an alternative, consider a traditional network topology based on the hub and spoke model, as partner NVAs can be deployed in a regular hub virtual network.
- When you're deploying partner networking technologies and NVAs, follow the partner vendor's guidance to ensure there are no conflicting configurations with Azure networking.
- For brownfield scenarios where you're migrating from a hub-and-spoke network topology not based on Virtual WAN, see Migrate to Azure Virtual WAN.
- Create Azure Virtual WAN and Azure Firewall resources within the connectivity subscription.
- Don't create more than 500 virtual network connections per Virtual WAN virtual hub.
    - If more than 500 virtual network connections per Virtual WAN virtual hub are required, deploy an additional Virtual WAN virtual hub in the same region as part of the same Virtual WAN and same resource group.
- Plan your deployment carefully, and ensure that your network architecture is within the Azure Virtual WAN limits.
- Use insights in Azure Monitor for Virtual WAN (preview) to monitor the end-to-end topology of your Virtual WAN and status and key metrics.
- Deploy a single Azure DDoS standard protection plan in the connectivity subscription.
    - All landing zone and platform VNets should use this plan.

Traditional Azure networking topology

8/23/2021 • 7 minutes to read • Edit Online

Explore key design considerations and recommendation surrounding network topologies in Microsoft Azure.

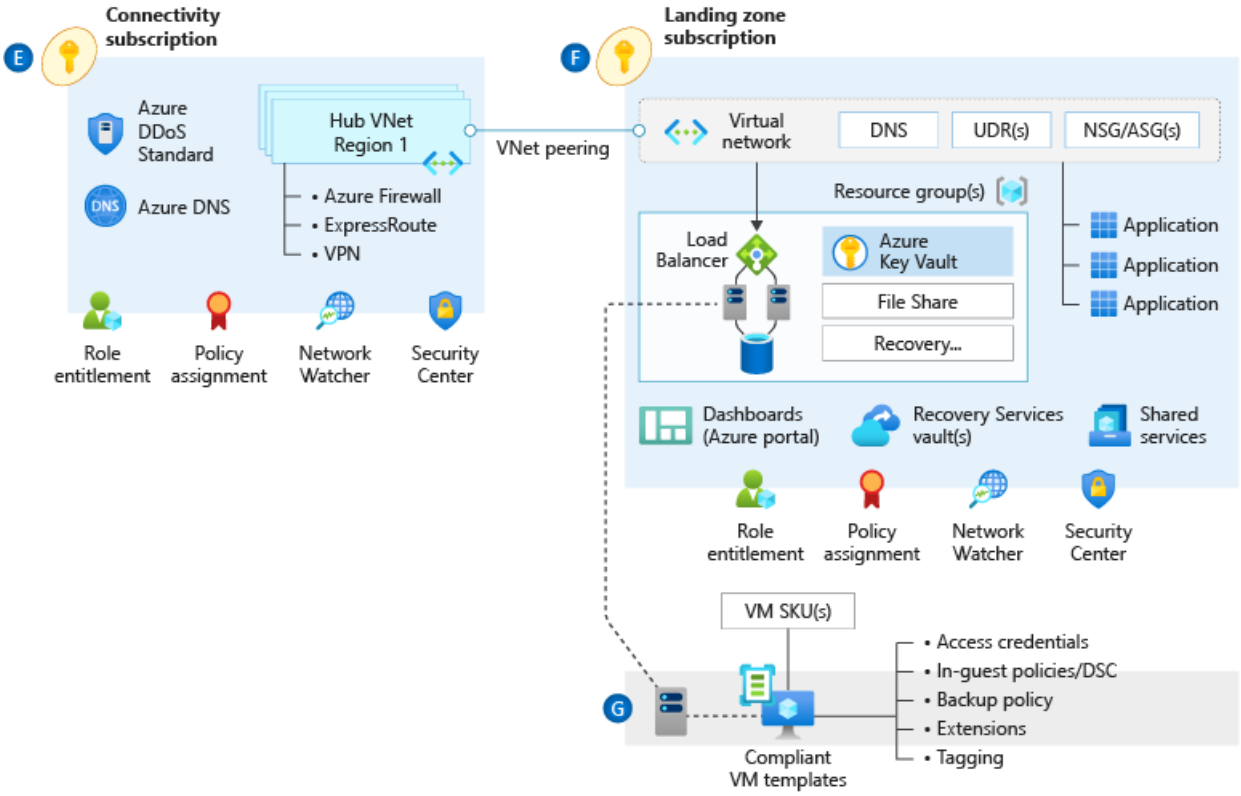


Figure 1: A traditional Azure network topology.

Design considerations:

- Various network topologies can connect multiple landing zone virtual networks. Examples are one large flat virtual network, multiple virtual networks connected with multiple Azure ExpressRoute circuits or connections, hub and spoke, full mesh, and hybrid.
- Virtual networks can't traverse subscription boundaries. However, you can achieve connectivity between virtual networks across different subscriptions by using virtual network peering, an ExpressRoute circuit, or VPN gateways.
- Virtual network peering is the preferred method to connect virtual networks in Azure. You can use virtual network peering to connect virtual networks in the same region, across different Azure regions, and across different Azure Active Directory (Azure AD) tenants.
- Virtual network peering and global virtual network peering aren't transitive. User-defined routes (UDRs) and network virtual appliances (NVAs) are required to enable a transit network. For more information, see Hub-spoke network topology in Azure.
- An Azure DDoS Protection standard protection plan can be shared across all virtual networks in a single Azure AD Tenant to protect resources with public IP addresses. For more information, see Azure DDoS Protection Standard.
    - Only resources with Public IP addresses are covered by Azure DDoS Protection standard protection plans.

    - 100 public IP addresses are included in the cost of an Azure DDoS Protection standard protection plan across all protected virtual networks associated to the DDoS Protection plan. Any other public IP addresses over the 100 included with the plan, are charged separately. For more information on Azure DDoS Protection Standard Protection pricing, see the pricing page or the FAQ.
    - Review the supported resources of Azure DDoS Protection standard protection plans
- You can use ExpressRoute circuits to establish connectivity across virtual networks within the same geopolitical region or by using the premium add-on for connectivity across geopolitical regions. Keep these points in mind:
    - Network-to-network traffic might experience more latency because traffic must hairpin at the Microsoft Enterprise Edge (MSEE) routers.
    - Bandwidth will be constrained to the ExpressRoute gateway SKU.
    - Deploy and manage UDRs if they need to be inspected or logged for traffic across virtual networks.
- VPN gateways with Border Gateway Protocol (BGP) are transitive within Azure and on-premises, but they don't provide transitive access to networks connected via ExpressRoute.
- When multiple ExpressRoute circuits are connected to the same virtual network, use connection weights and BGP techniques to ensure an optimal path for traffic between on-premises and Azure. For more information, see Optimize ExpressRoute routing.
- Using BGP metrics to influence ExpressRoute routing is a configuration change made outside of the Azure platform. Your organization or your connectivity provider must configure the on-premises routers accordingly.
- ExpressRoute circuits with premium add-ons provide global connectivity.
- ExpressRoute is bound to certain limits; there are a maximum number of ExpressRoute connections per ExpressRoute gateway and ExpressRoute private peering can identify a maximum number of routes from Azure to on-premises. See ExpressRoute limits for more information about these limits.
- A VPN gateway's maximum aggregated throughput is 10 gigabits per second. It supports up to 30 site-to-site or network-to-network tunnels.

Design recommendations:

- Consider a network design based on the traditional hub-and-spoke network topology for the following scenarios:
    - A network architecture deployed within a single Azure region.
    - A network architecture that spans multiple Azure regions, with no need for transitive connectivity between virtual networks for landing zones across regions.
    - A network architecture that spans multiple Azure regions, and global virtual network peering can be used to connect virtual networks across Azure regions.
    - There's no need for transitive connectivity between VPN and ExpressRoute connections.
    - The main hybrid connectivity method in place is ExpressRoute, and the number of VPN connections is less than 30 per VPN gateway.
    - There's a dependency on centralized NVAs and granular routing.
- For regional deployments, primarily use the hub-and-spoke topology. Use landing zone virtual networks that connect with virtual network peering to a central hub virtual network for cross-premises connectivity

via ExpressRoute, VPN for branch connectivity, spoke-to-spoke connectivity via NVAs and UDRs, and internet-outbound protection via Azure Firewall or other third-party NVA. The following diagram shows this topology. This configuration allows for appropriate traffic control to meet most requirements for segmentation and inspection.

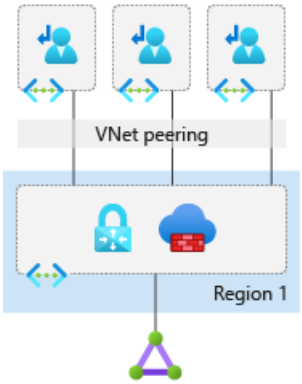


Figure 2: Hub-and-spoke network topology.

- Use the topology of multiple virtual networks connected with multiple ExpressRoute circuits when one of these conditions is true:
  - You need a high level of isolation.
  - You need dedicated ExpressRoute bandwidth for specific business units.
  - You've reached the maximum number of connections per ExpressRoute gateway (refer to the ExpressRoute limits article for the maximum number).

The following figure shows this topology.

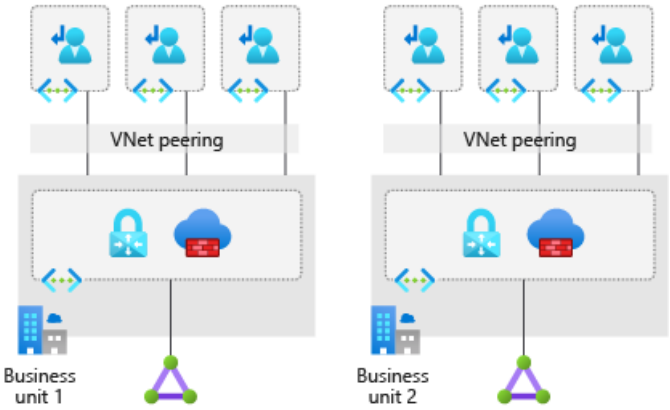


Figure 3: Multiple virtual networks connected with multiple ExpressRoute circuits.

- Deploy a set of minimal shared services, including ExpressRoute gateways, VPN gateways (as required), and Azure Firewall or partner NVAs (as required) in the central-hub virtual network. If necessary, also deploy Active Directory domain controllers and DNS servers.
- Deploy Azure Firewall or partner NVAs for east/west or south/north traffic protection and filtering, in the central-hub virtual network.
- When you're deploying partner networking technologies or NVAs, follow the partner vendor's guidance to ensure that:
  - The vendor supports deployment.
  - The guidance is designed for high availability and maximum performance.
  - There are no conflicting configurations with Azure networking.

- Don't deploy Layer-7 inbound NVAs, such as Azure Application Gateway, as a shared service in the central-hub virtual network. Instead, deploy them together with the application in their respective landing zones.
- Deploy a single Azure DDoS standard protection plan in the Connectivity subscription.
  - All Landing Zone and Platform virtual networks should use this plan.
- Use your existing network, multiprotocol label switching, and SD-WAN to connect branch locations with corporate headquarters. Transit in Azure between ExpressRoute and VPN gateways isn't supported.
- When you have hub-and-spoke networks in multiple Azure regions and a few landing zones need to connect across regions, use global virtual network peering to directly connect landing zone virtual networks that need to route traffic to each other. Depending on the communicating VM's SKU, global virtual network peering can provide high network throughput. Traffic between directly peered landing zone virtual networks will bypass NVAs within hub virtual networks. This would also be subject to limitations on global virtual network peering.
- When you have hub-and-spoke networks in multiple Azure regions and most landing zones need to connect across regions (or when using direct peering to bypass hub NVAs isn't compatible with your security requirements), use hub NVAs to connect hub virtual networks in each region to each other and route traffic across regions. Global virtual network peering or ExpressRoute circuits can help to connect Hub virtual networks in the following ways:
  - Global virtual network peering provides a low latency and high throughput connection but generates traffic fees.
  - Routing via ExpressRoute might lead to increased latency (due to MSEE hairpin) and throughput will be constrained to the ExpressRoute Gateway SKU.

The following figure shows both options:

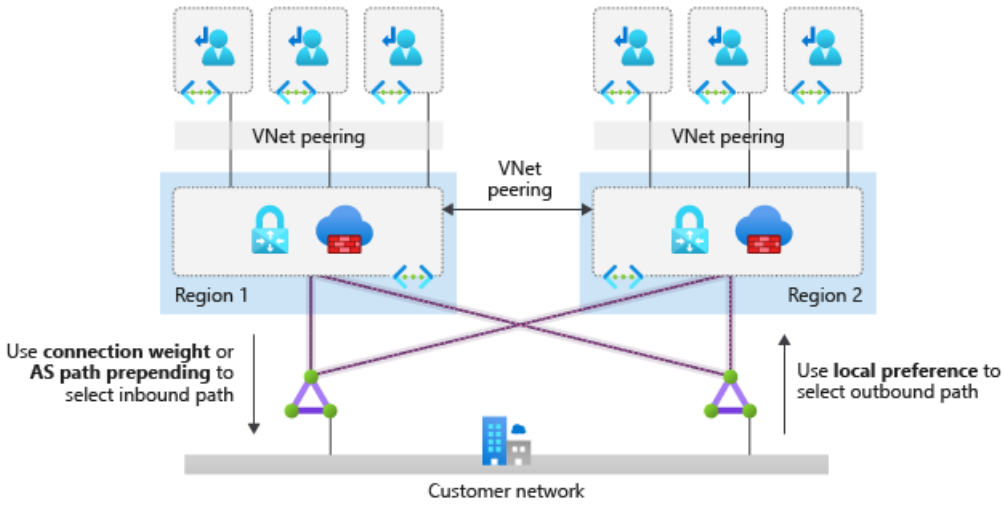


Figure 4: Options for hub-to-hub connectivity.

- When two Azure regions need to connect, evaluate and use global virtual network peering or the same ExpressRoute circuit(s) to connect both hub virtual networks.
- When more than two Azure regions need to connect, then the recommendation is for hub virtual networks in each region to connect to the same ExpressRoute circuit(s). Global virtual network peering would require managing a large number of peering relationships and a complex set of user-defined routes (UDRs) across multiple virtual networks. The following diagram shows how to connect hub-and-poke networks in three regions:

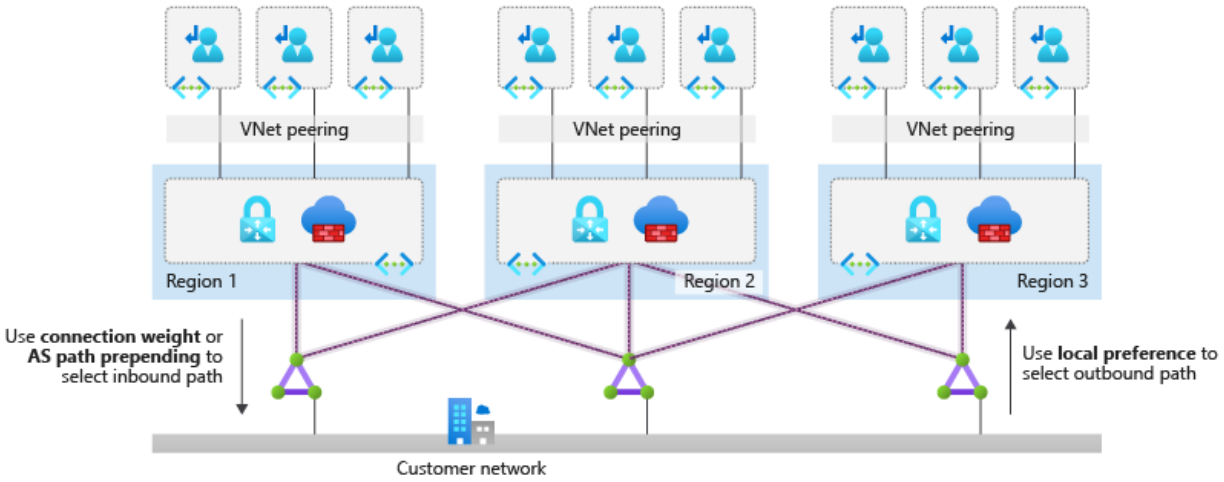


Figure 5: ExpressRoute providing hub-to-hub connectivity between multiple regions.

- When using ExpressRoute circuits for cross-region connectivity, spokes in different regions will communicate directly and bypass the firewall, as they will learn via BGP routes to the spokes of the remote hub. If you require traffic across spokes to be inspected by the firewall NVAs in the hub virtual networks, you must implement one of these options:
  - Create more specific route entries in the spoke UDRs for the firewall in the local hub virtual network to redirect traffic across hubs.
  - To simplify route configuration, disable BGP propagation on the spoke route tables.
- When your organization requires hub-and-spoke network architectures across more than two Azure regions, global transit connectivity between landing zones virtual networks across Azure regions is required, and you want to minimize network management overhead, it's recommended to evaluate a managed global transit network architecture that's based on Virtual WAN.
- Deploy each region's hub network resources into separate resource groups, and sort them into each deployed region.
- Use Azure Monitor for Networks (preview) to monitor the end-to-end state of your networks on Azure.
- Two limits must be considered when you connect spoke virtual networks to the central hub virtual network:
  - The maximum number of virtual network peering connections per virtual network
  - The maximum number of prefixes that ExpressRoute with private peering advertises from Azure to on-premises

  Ensure that the number of spoke virtual networks connected to the hub virtual network don't exceed these limits.

Connectivity to Azure

3/6/2021 • 2 minutes to read • Edit Online

This section expands on the network topology to consider recommended models for connecting on-premises locations to Azure.

Design considerations:

- Azure ExpressRoute provides dedicated private connectivity to Azure infrastructure as a service (IaaS) and platform as a service (PaaS) functionality from on-premises locations.
- You can use Private Link to establish connectivity to PaaS services over ExpressRoute with private peering.
- When multiple virtual networks are connected to the same ExpressRoute circuit, they'll become part of the same routing domain, and all virtual networks will share the bandwidth.
- You can use ExpressRoute Global Reach, where available, to connect on-premises locations together through ExpressRoute circuits to transit traffic over the Microsoft backbone network.
- ExpressRoute Global Reach is available in many ExpressRoute peering locations.
- ExpressRoute Direct allows creation of multiple ExpressRoute circuits at no additional cost, up to the ExpressRoute Direct port capacity (10 Gbps or 100 Gbps). It also allows you to connect directly to Microsoft's ExpressRoute routers. For the 100-Gbps SKU, the minimum circuit bandwidth is 5 Gbps. For the 10-Gbps SKU, the minimum circuit bandwidth is 1 Gbps.

Design recommendations:

- Use ExpressRoute as the primary connectivity channel for connecting an on-premises network to Azure. You can use VPNs as a source of backup connectivity to enhance connectivity resiliency.
- Use dual ExpressRoute circuits from different peering locations when you're connecting an on-premises location to virtual networks in Azure. This setup will ensure redundant paths to Azure by removing single points of failure between on-premises and Azure.
- When you use multiple ExpressRoute circuits, optimize ExpressRoute routing via BGP local preference and AS PATH prepending.
- Ensure that you're using the right SKU for the ExpressRoute/VPN gateways based on bandwidth and performance requirements.
- Deploy a zone-redundant ExpressRoute gateway in the supported Azure regions.
- For scenarios that require bandwidth higher than 10 Gbps or dedicated 10/100-Gbps ports, use ExpressRoute Direct.
- When low latency is required, or throughput from on-premises to Azure must be greater than 10 Gbps, enable FastPath to bypass the ExpressRoute gateway from the data path.
- Use VPN gateways to connect branches or remote locations to Azure. For higher resilience, deploy zone-redundant gateways (where available).
- Use ExpressRoute Global Reach to connect large offices, regional headquarters, or datacenters connected to Azure via ExpressRoute.
- When traffic isolation or dedicated bandwidth is required, such as for separating production and

nonproduction environments, use different ExpressRoute circuits. It will help you ensure isolated routing domains and alleviate noisy-neighbor risks.

- Proactively monitor ExpressRoute circuits by using Network Performance Monitor.
- Don't explicitly use ExpressRoute circuits from a single peering location. This creates a single point of failure and makes your organization susceptible to peering location outages.

Connectivity to Azure PaaS services

7/9/2021 • 2 minutes to read • Edit Online

Building on the previous connectivity sections, this section explores recommended connectivity approaches for using Azure PaaS services.

Design considerations:

- Azure PaaS services are typically accessed over public endpoints. However, the Azure platform provides capabilities to secure such endpoints or even make them entirely private:
  - Virtual network injection provides dedicated private deployments for supported services. Management plane traffic still flows through public IP addresses.
  - Azure Private Link provides dedicated access by using private IP addresses to Azure PaaS instances or custom services behind Azure Load Balancer Standard tier. For more information, see key benefits of Private Link.
  - Virtual network service endpoints provide service-level access from selected subnets to selected PaaS services. Review the list of supported services on the link provided.
  - What is the difference between Service Endpoints and Private Endpoints? provides an explanation of the differences between Private Link Endpoints and VNet Service Endpoints.
- Enterprises often have concerns about public endpoints for PaaS services that must be appropriately mitigated.
- For supported services, Private Link addresses data exfiltration concerns associated with VNet service endpoints. As an alternative, you can use outbound filtering via network virtual appliances (NVAs) to provide steps to mitigate data exfiltration.

Design recommendations:

- Use virtual network injection for supported Azure services to make them available from within your virtual network.
- Azure PaaS services that have been injected into a virtual network still perform management plane operations by using public IP addresses. Ensure that this communication is locked down within the virtual network by using UDRs and NSGs.
- Use Private Link, where available, for shared Azure PaaS services. Private Link is generally available for several services and is in public preview for other services.
- Access Azure PaaS services from on-premises via ExpressRoute with Private peering. Use either virtual network injection for dedicated Azure services or Azure Private Link for available shared Azure services. To access Azure PaaS services from on-premises when virtual network injection or Private Link isn't available, use ExpressRoute with Microsoft peering. This method avoids transiting over the public internet.
  - Accessing Azure PaaS Services from on-premises via ExpressRoute with Microsoft Peering does not prevent access to the public endpoints of the PaaS service. This must be configured and restricted as required separately.
- Use virtual network service endpoints to secure access to Azure PaaS services from within your virtual network, but only when Private Link isn't available and there are no data exfiltration concerns. To address

data exfiltration concerns with service endpoints, use NVA filtering or use virtual network service endpoint policies for Azure Storage.

- Don't enable virtual network service endpoints by default on all subnets.
- Don't use virtual network service endpoints when there are data exfiltration concerns, unless you use NVA filtering.
- We don't recommend that you implement forced tunneling to enable communication from Azure to Azure resources.

Plan for inbound and outbound internet connectivity

3/6/2021 • 2 minutes to read • Edit Online

This section describes recommended connectivity models for inbound and outbound connectivity to and from the public internet.

Design considerations:

- Azure-native network security services such as Azure Firewall, Azure Web Application Firewall (WAF) on Azure Application Gateway, and Azure Front Door are fully managed services. So you don't incur the operational and management costs associated with infrastructure deployments, which can become complex at scale.
- The enterprise-scale architecture is fully compatible with partner NVAs, if your organization prefers to use NVAs or for situations where native services don't satisfy your organization's specific requirements.

Design recommendations:

- Use Azure Firewall to govern:
  - Azure outbound traffic to the internet.
  - Non-HTTP/S inbound connections.
  - East/west traffic filtering (if your organization requires it).
- Use Firewall Manager with Virtual WAN to deploy and manage Azure firewalls across Virtual WAN hubs or in hub virtual networks. Firewall Manager is now in general availability for both Virtual WAN and regular virtual networks.
- Create a global Azure Firewall policy to govern security posture across the global network environment and assign it to all Azure Firewall instances. Allow for granular policies to meet requirements of specific regions by delegating incremental firewall policies to local security teams via Azure role-based access control.
- Configure supported partner SaaS security providers within Firewall Manager if your organization wants to use such solutions to help protect outbound connections.
- Use WAF within a landing-zone virtual network for protecting inbound HTTP/S traffic from the internet.
- Use Azure Front Door and WAF policies to provide global protection across Azure regions for inbound HTTP/S connections to a landing zone.
- When you're using Azure Front Door and Azure Application Gateway to help protect HTTP/S applications, use WAF policies in Azure Front Door. Lock down Azure Application Gateway to receive traffic only from Azure Front Door.
- If partner NVAs are required for east/west or south/north traffic protection and filtering:
  - For Virtual WAN network topologies, deploy the NVAs to a separate virtual network (for example, NVA virtual network). Then connect it to the regional Virtual WAN hub and to the landing zones that require access to NVAs. This article describes the process.
  - For non-Virtual WAN network topologies, deploy the partner NVAs in the central-hub virtual network.

- If partner NVAs are required for inbound HTTP/S connections, deploy them within a landing-zone virtual network and together with the applications that they're protecting and exposing to the internet.
- Use Azure DDoS Protection Standard protection plans to help protect all public endpoints hosted within your virtual networks.
- Don't replicate on-premises perimeter network concepts and architectures into Azure. Similar security capabilities are available in Azure, but the implementation and architecture must be adapted to the cloud.

Plan for application delivery

3/6/2021 • 2 minutes to read • Edit Online

This section explores key recommendations to deliver internal-facing and external-facing applications in a secure, highly scalable, and highly available way.

Design considerations:

- Azure Load Balancer (internal and public) provides high availability for application delivery at a regional level.
- Azure Application Gateway allows the secure delivery of HTTP/S applications at a regional level.
- Azure Front Door allows the secure delivery of highly available HTTP/S applications across Azure regions.
- Azure Traffic Manager allows the delivery of global applications.

Design recommendations:

- Perform application delivery within landing zones for both internal-facing and external-facing applications.
- For secure delivery of HTTP/S applications, use Application Gateway v2 and ensure that WAF protection and policies are enabled.
- Use a partner NVA if you can't use Application Gateway v2 for the security of HTTP/S applications.
- Deploy Azure Application Gateway v2 or partner NVAs used for inbound HTTP/S connections within the landing-zone virtual network and with the applications that they're securing.
- Use a DDoS standard protection plan for all public IP addresses in a landing zone.
- Use Azure Front Door with WAF policies to deliver and help protect global HTTP/S applications that span Azure regions.
- When you're using Front Door and Application Gateway to help protect HTTP/S applications, use WAF policies in Front Door. Lock down Application Gateway to receive traffic only from Front Door.
- Use Traffic Manager to deliver global applications that span protocols other than HTTP/S.

Plan for landing zone network segmentation

5/18/2021 • 2 minutes to read • Edit Online

This section explores key recommendations to deliver highly secure internal network segmentation within a landing zone to drive a network zero-trust implementation.

Design considerations:

- The zero-trust model assumes a breached state and verifies each request as though it originates from an uncontrolled network.
- An advanced zero-trust network implementation employs fully distributed ingress/egress cloud micro-perimeters and deeper micro-segmentation.
- Network security groups can use Azure service tags to facilitate connectivity to Azure PaaS services.
- Application security groups don't span or provide protection across virtual networks.
- NSG flow logs are now supported through Azure Resource Manager templates.

Design recommendations:

- Delegate subnet creation to the landing zone owner. This will enable them to define how to segment workloads across subnets (for example, a single large subnet, multitier application, or network-injected application). The platform team can use Azure Policy to ensure that an NSG with specific rules (such as deny inbound SSH or RDP from internet, or allow/block traffic across landing zones) is always associated with subnets that have deny-only policies.
- Use NSGs to help protect traffic across subnets, as well as east/west traffic across the platform (traffic between landing zones).
- The application team should use application security groups at the subnet-level NSGs to help protect multitier VMs within the landing zone.
- Use NSGs and application security groups to micro-segment traffic within the landing zone and avoid using a central NVA to filter traffic flows.
- Enable NSG flow logs and feed them into Traffic Analytics to gain insights into internal and external traffic flows.
- Use NSGs to selectively allow connectivity between landing zones.
- For Virtual WAN topologies, route traffic across landing zones via Azure Firewall if your organization requires filtering and logging capabilities for traffic flowing across landing zones.
- If your organization decides to implement forced tunneling (advertise default route) to on-premises, we recommend incorporating the following **outbound** NSG rules to deny egress traffic from VNets directly to the internet should the BGP session drop.

> **NOTE**
> Rule priorities will need to be adjusted based on your existing NSG rule set.

PRIORITY	NAME	SOURCE	DESTINATION	SERVICE	ACTION	REMARK
100	`AllowLocal`	`Any`	`VirtualNetwork`	`Any`	`Allow`	Allow traffic during normal operations. With Forced Tunneling enabled, `0.0.0.0/0` is considered part of the `VirtualNetwork` tag as long as BGP is advertising it to the ExpressRoute or VPN gateway.
110	`DenyInternet`	`Any`	`Internet`	`Any`	`Deny`	Deny traffic directly to the internet if the `0.0.0.0/0` route is withdrawn from the routes advertised (for example, due to an outage or misconfiguration).

Define network encryption requirements

3/6/2021 • 2 minutes to read • Edit Online

This section explores key recommendations to achieve network encryption between on-premises and Azure as well as across Azure regions.

Design considerations:

- Cost and available bandwidth are inversely proportional to the length of the encryption tunnel between endpoints.
- When you're using a VPN to connect to Azure, traffic is encrypted over the internet via IPsec tunnels.
- When you're using ExpressRoute with private peering, traffic isn't currently encrypted.
- Configuring a Site-to-Site VPN connection over ExpressRoute private peering is now in preview.
- You can apply media access control security (MACsec) encryption to ExpressRoute Direct to achieve network encryption.
- When Azure traffic moves between datacenters (outside physical boundaries not controlled by Microsoft or on behalf of Microsoft), MACsec data-link layer encryption is used on the underlying network hardware. This is applicable to VNet peering traffic.

Design recommendations:

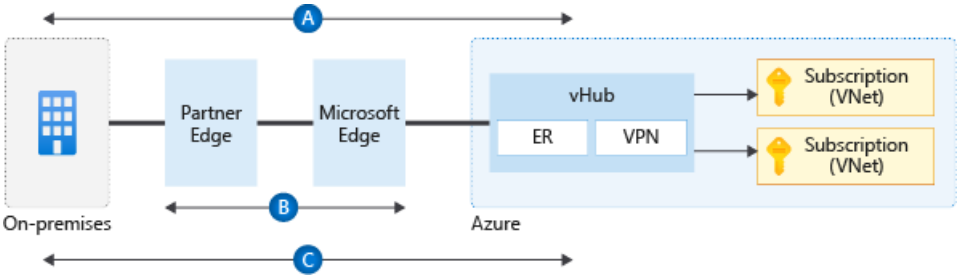


Figure 1: Encryption flows.

- When you're establishing VPN connections from on-premises to Azure by using VPN gateways, traffic is encrypted at a protocol level through IPsec tunnels. The preceding diagram shows this encryption in flow `A`.
- When you're using ExpressRoute Direct, configure MACsec in order to encrypt traffic at the layer-two level between your organization's routers and MSEE. The diagram shows this encryption in flow `B`.
- For Virtual WAN scenarios where MACsec isn't an option (for example, not using ExpressRoute Direct), use a Virtual WAN VPN gateway to establish IPsec tunnels over ExpressRoute private peering. The diagram shows this encryption in flow `C`.
- For non-Virtual WAN scenarios, and where MACsec isn't an option (for example, not using ExpressRoute Direct), the only options are:
  - Use partner NVAs to establish IPsec tunnels over ExpressRoute private peering.
  - Establish a VPN tunnel over ExpressRoute with Microsoft peering.
  - Evaluate the capability to configure a Site-to-Site VPN connection over ExpressRoute private peering (in preview.

- If traffic between Azure regions must be encrypted, use global VNet peering to connect virtual networks across regions.
- If native Azure solutions (as shown in flows `B` and `C` in the diagram) don't meet your requirements, use partner NVAs in Azure to encrypt traffic over ExpressRoute private peering.

Plan for traffic inspection

5/12/2021 • 2 minutes to read • Edit Online

In many industries, organizations require that traffic in Azure is mirrored to a network packet collector for deep inspection and analysis. This requirement typically focuses on inbound and outbound internet traffic. This section explores key considerations and recommended approaches for mirroring or tapping traffic within Azure Virtual Network.

Design considerations:

- The Azure Virtual Network terminal access point (TAP) preview is currently on hold in all Azure regions.
- Packet capture in Azure Network Watcher is generally available, but captures are limited to a maximum period of five hours.

Design recommendations:

As an alternative to Azure Virtual Network TAP, evaluate the following options:

- Use Network Watcher packets to capture despite the limited capture window.
- Evaluate whether the latest version of NSG flow logs provides the level of detail that you need.
- Use partner solutions for scenarios that require deep packet inspection.
- Don't develop a custom solution to mirror traffic. Although this approach might be acceptable for small-scale scenarios, we don't encourage it at scale because of complexity and the supportability issues that might arise.

Connectivity to other cloud providers

6/7/2021 • 2 minutes to read • Edit Online

This guidance discusses ways to connect an Azure enterprise-scale landing zone architecture to other cloud providers, such as Amazon Web Services (AWS) and Google Cloud Platform (GCP).

The various options differ in speed, latency, reliability, service level agreements (SLAs), complexity, and costs. This article considers options and makes recommendations.

> **NOTE**
> Microsoft and Oracle partnered to provide high-throughput, low-latency cross-connections between Azure and Oracle Cloud Infrastructure (OCI). For more information, see Connectivity to Oracle Cloud Infrastructure.

Design considerations

- We consider the following options to connect Azure to another cloud:
  - **Option 1** - Connect Azure ExpressRoute and the other cloud provider's equivalent private connection. The customer manages routing.
  - **Option 2** - Connect ExpressRoute and the other cloud provider's equivalent private connection. A cloud exchange provider handles routing.
  - **Option 3** - Use site-to-site VPN over the Internet. For more information, see Connect on-premises networks to Azure by using site-to-site VPN gateways (Learn).

  You can use the following cross-cloud connectivity flow chart as an aid to choosing an option:

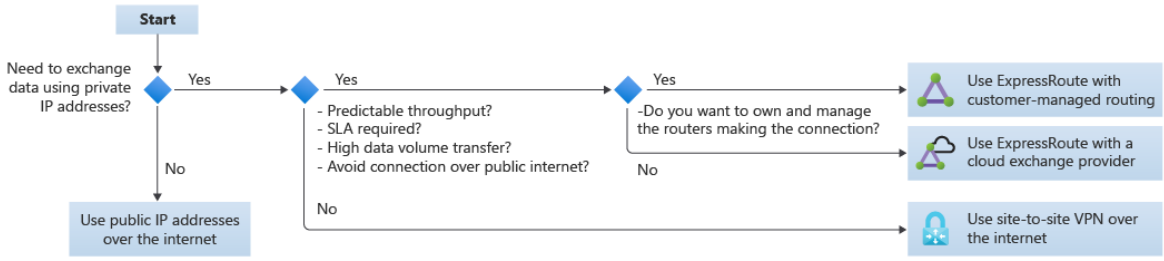


Figure 1: Cross-cloud connectivity flow chart

- You can only connect an Azure virtual network to another cloud provider's virtual private cloud (VPC) if the private IP address spaces don't overlap.
- Site-to-site VPN might have lower throughput and higher latency than the ExpressRoute options.
- Site-to-site VPN is the fastest deployment option if Azure ExpressRoute and the other cloud provider equivalent aren't already in use.
- Routing complexity of Azure ExpressRoute and other cloud provider equivalent with customer-managed routing can be high if not done through a cloud exchange provider.
- You might need to provide DNS resolution between Azure and the other cloud provider. This configuration might incur extra costs.
- The FastPath feature of ExpressRoute improves data path performance between Azure and on-premises networks, and between Azure and other cloud providers. When enabled, FastPath sends network traffic

directly to virtual machines in the virtual network, bypassing the ExpressRoute gateway. For more information, see About ExpressRoute FastPath.

- FastPath is available on all ExpressRoute circuits.
- FastPath still requires a virtual network gateway to be created for route exchange purposes. The virtual network gateway must use either the Ultra Performance SKU or the ErGw3AZ SKU for the ExpressRoute gateway to enable route management.
- There are configurations that FastPath doesn't support, such as virtual network peering. For more information, see Limitations in About ExpressRoute FastPath.

Design recommendations

- Use option 1 or option 2 to avoid use of the public internet, if you require an SLA, if you want predictable throughput, or need to handle heavy data traffic. Consider whether to use a customer-managed routing or a cloud exchange provider if you haven't implemented ExpressRoute already.
- Create the ExpressRoute circuits for option 1 and option 2 in the connectivity subscription.
- Use the ExpressRoute circuit of option 1 or option 2 to connect to the hub of a hub-and-spoke architecture, or to the hub virtual network or virtual WAN hub of an Azure virtual WAN-based network. For more information, see Figure 2 and Figure 3.

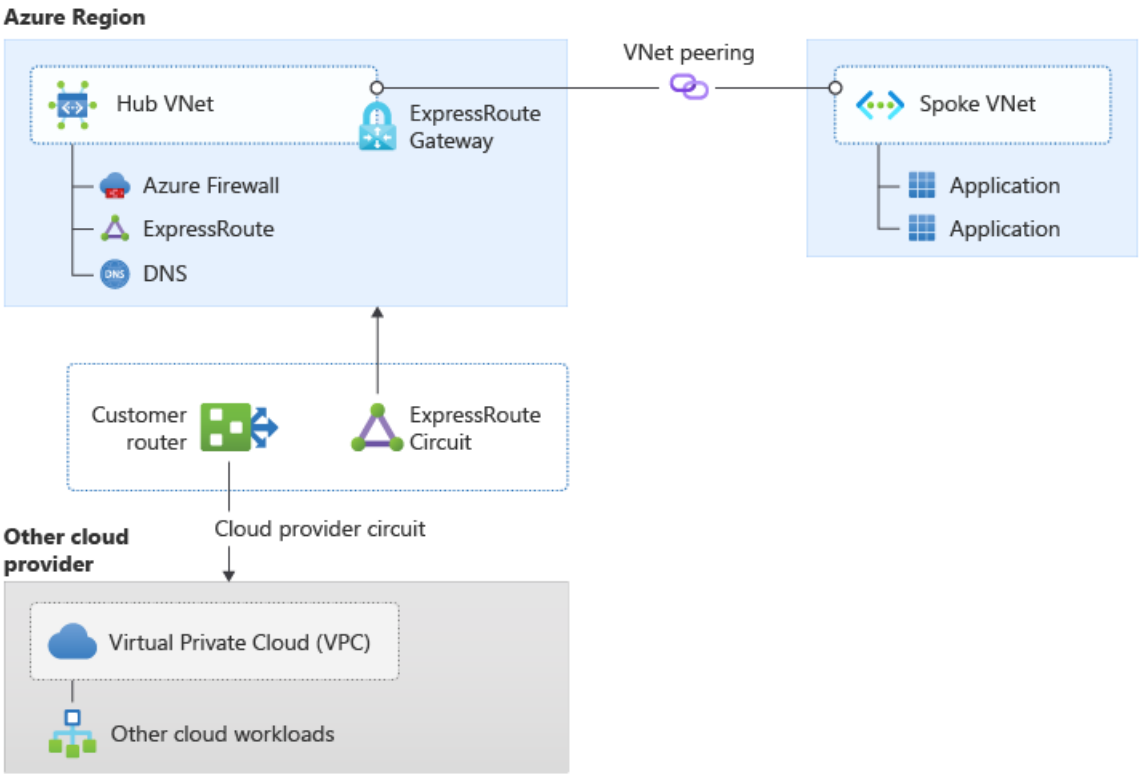


Figure 2: Cross-cloud connectivity with customer-managed routing (Option 1)

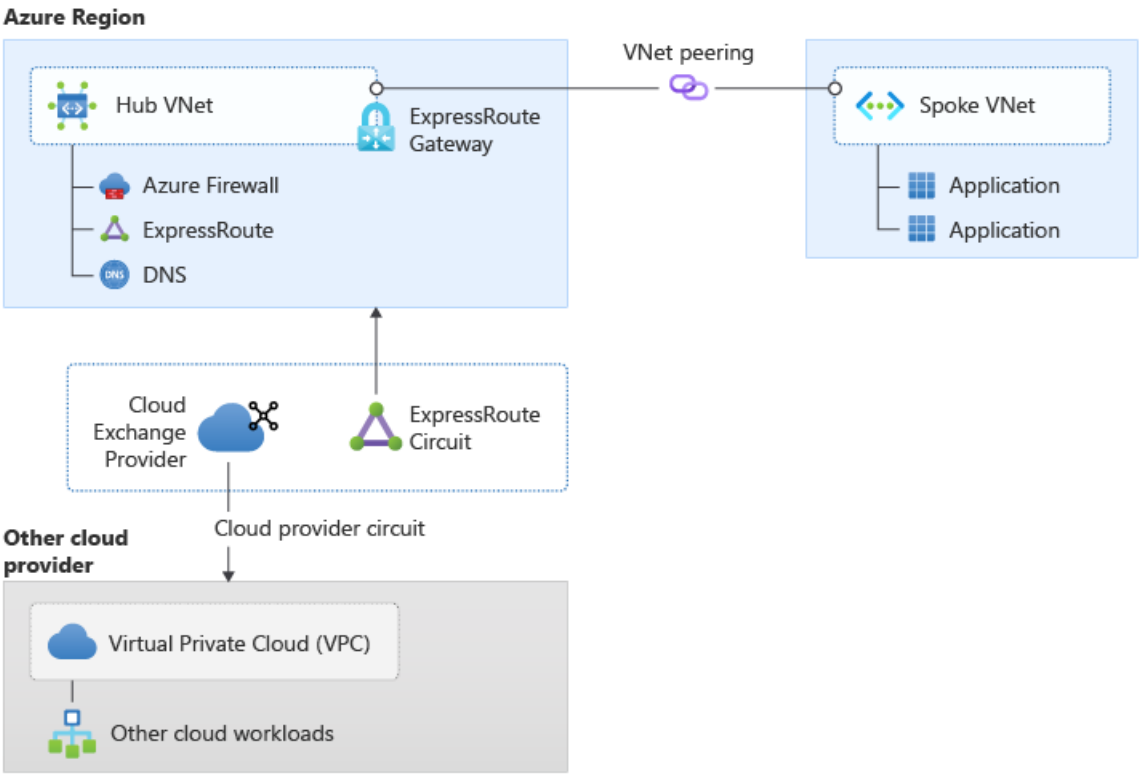
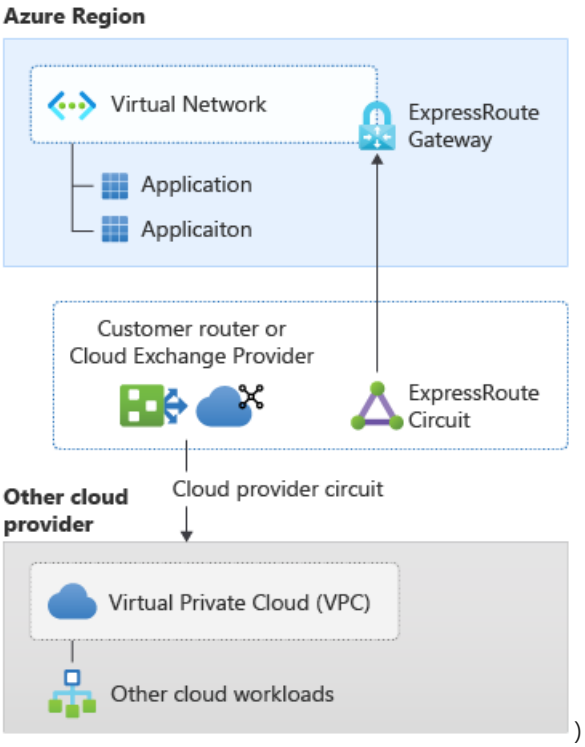


Figure 3: Cross-cloud connectivity with a cloud exchange provider (Option 2)

- If you need to minimize latency between Azure and another cloud provider, consider deploying your application in a single virtual network with an ExpressRoute gateway, and enable FastPath.



)

Figure 4: Cross-cloud connectivity with FastPath enabled

- If ExpressRoute isn't required or not available, you can use site-to-site VPN over the internet to connect between Azure and another cloud provider.

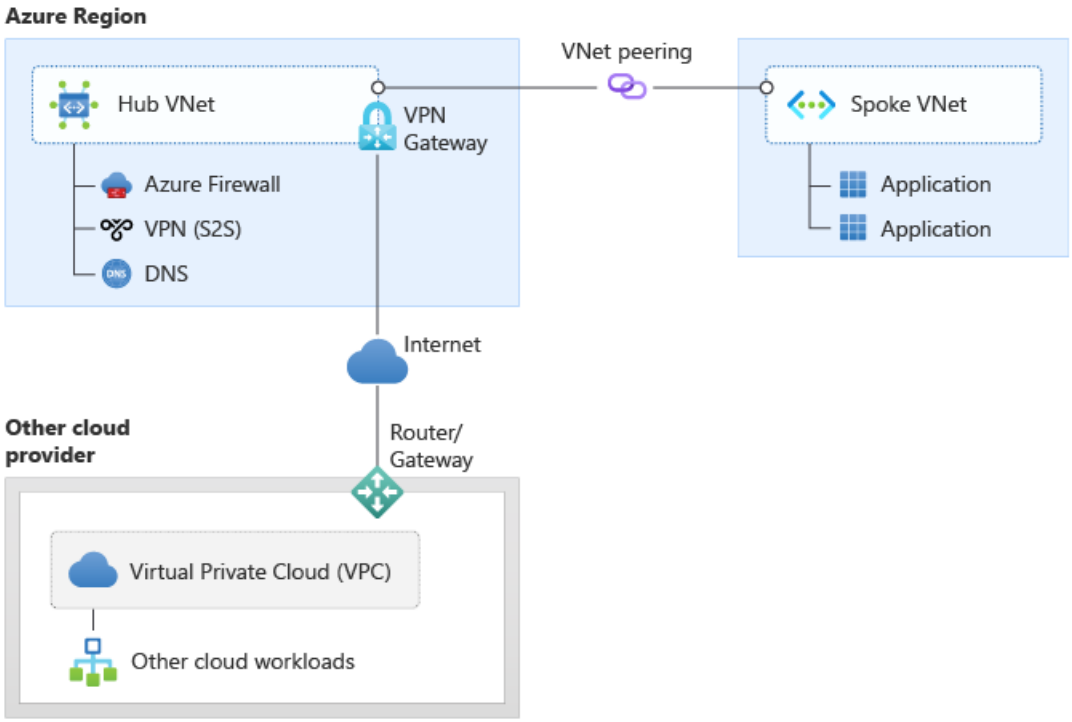


Figure 5: Cross-cloud connectivity using site-to-site VPN over the Internet

Next steps

To learn more about connectivity to Oracle Cloud Infrastructure (OCI), see Connectivity to Oracle Cloud Infrastructure.

Connectivity to Oracle Cloud Infrastructure

6/1/2021 • 2 minutes to read • Edit Online

This section provides different connectivity approaches to integrate an Azure enterprise-scale landing zone architecture to Oracle Cloud Infrastructure (OCI).

Design considerations:

- Using ExpressRoute and FastConnect, customers can connect a virtual network in Azure with a virtual cloud network in OCI, if the private IP address space doesn't overlap. Once you establish connectivity, resources in the Azure virtual network can communicate with resources in the OCI virtual cloud network as if they were both in the same network.
- Azure ExpressRoute FastPath is designed to improve the data path performance between two networks, both on-premises and Azure, and for this scenario, between OCI and Azure. When enabled, FastPath sends network traffic directly to virtual machines in the virtual network, bypassing the ExpressRoute gateway.
  - FastPath is available on all ExpressRoute circuits.
  - FastPath still requires a virtual network gateway to be created for route exchange purposes. The virtual network gateway must use either the Ultra Performance SKU or the ErGw3AZ SKU for the ExpressRoute gateway to enable route management.
- There are features that are currently not supported in ExpressRoute FastPath, such as Azure Virtual WAN hubs or VNet peering.
- While you can use ExpressRoute Global Reach to enable communication from on-premises to OCI via ExpressRoute circuits, it might incur more bandwidth costs that you can calculate by using the Azure pricing calculator. It's important to consider any extra costs when you migrate large amounts of data from on-premises to Oracle by using ExpressRoute circuits.
- In Azure regions that support Availability Zones, placing your Azure workloads in one zone or the other can have a small effect on latency. Design your application to balance availability and performances requirements.
- Interconnectivity between Azure and OCI is only available for specific regions.
- For more in-depth documentation about interconnectivity between Azure and OCI, see Oracle application solutions to integrate Microsoft Azure and Oracle Cloud Infrastructure or see Oracle documentation.

Design recommendations:

- Create the ExpressRoute circuit that will be used to interconnect Azure with OCI in the **connectivity** subscription.
- You can interconnect an Azure network architecture based on the traditional hub and spoke architecture or Azure Virtual WAN-based network topologies. It can be done by connecting the ExpressRoute circuit that will be used to interconnect Azure to OCI to the hub VNet or Virtual WAN hub as shown in the following diagram.

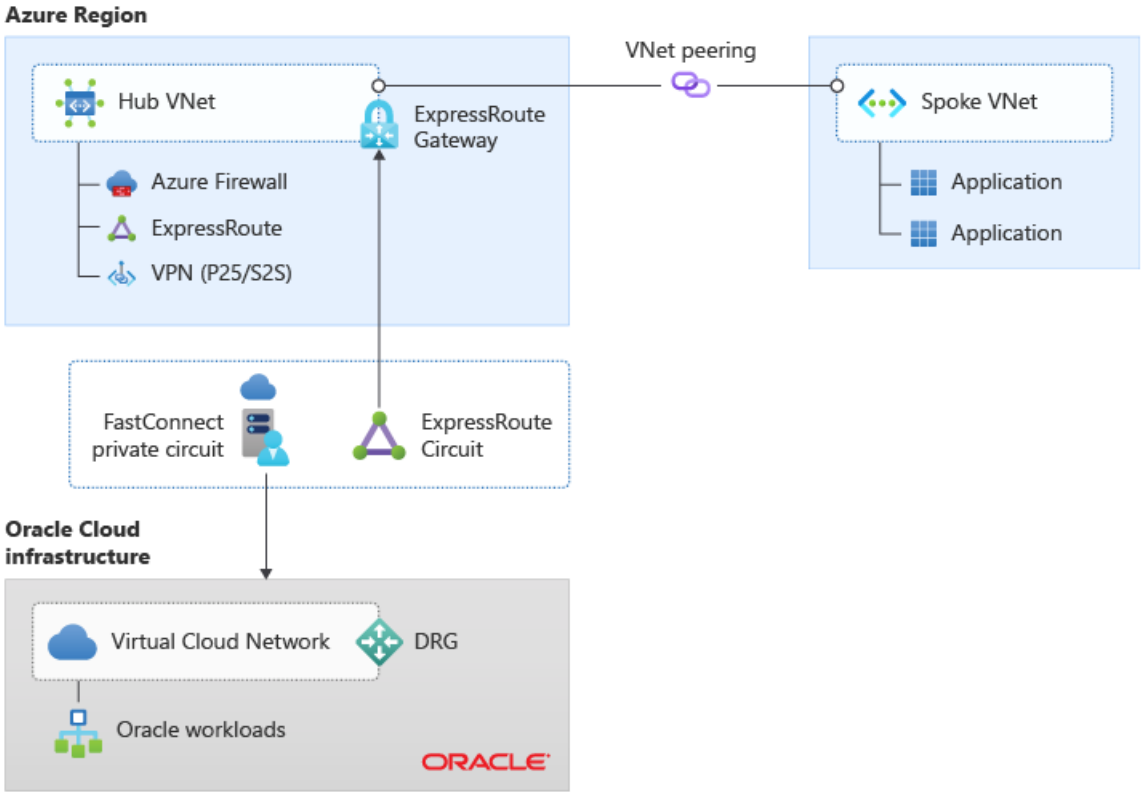


Figure 1: Interconnectivity between Azure and OCI via ExpressRoute.

- If your application requires the lowest possible latency between Azure and OCI, consider deploying your application in a single VNet with an ExpressRoute gateway and FastPath enabled.

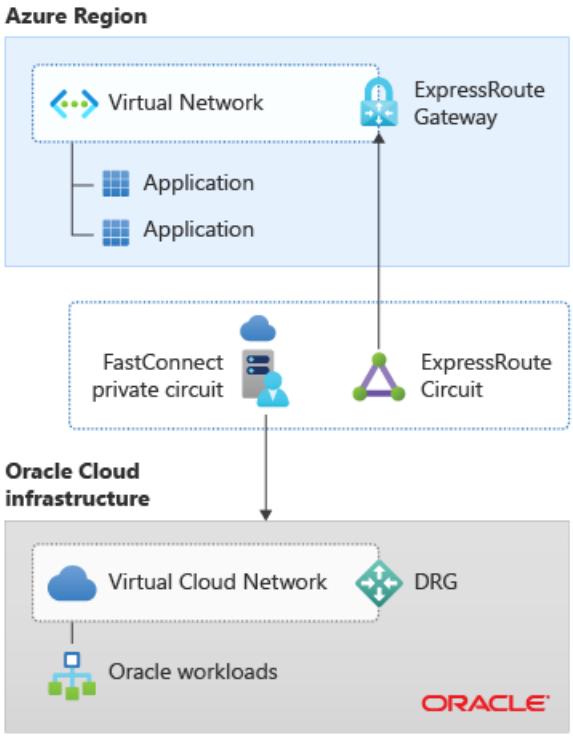


Figure 2: Interconnectivity between Azure and OCI with a single VNet.

- When you deploy Azure resources across Availability Zones, perform latency tests from Azure VMs located in different Availability Zones to OCI resources to understand which of the three Availability Zones provides the lowest latency to the OCI resources.
- To operate Oracle resources hosted in OCI by using Azure resources and technologies, you could:
  - **From Azure:** Deploy a jumpbox in a spoke VNet. The jumpbox provides access to the virtual

cloud network in OCI as shown in the following picture:

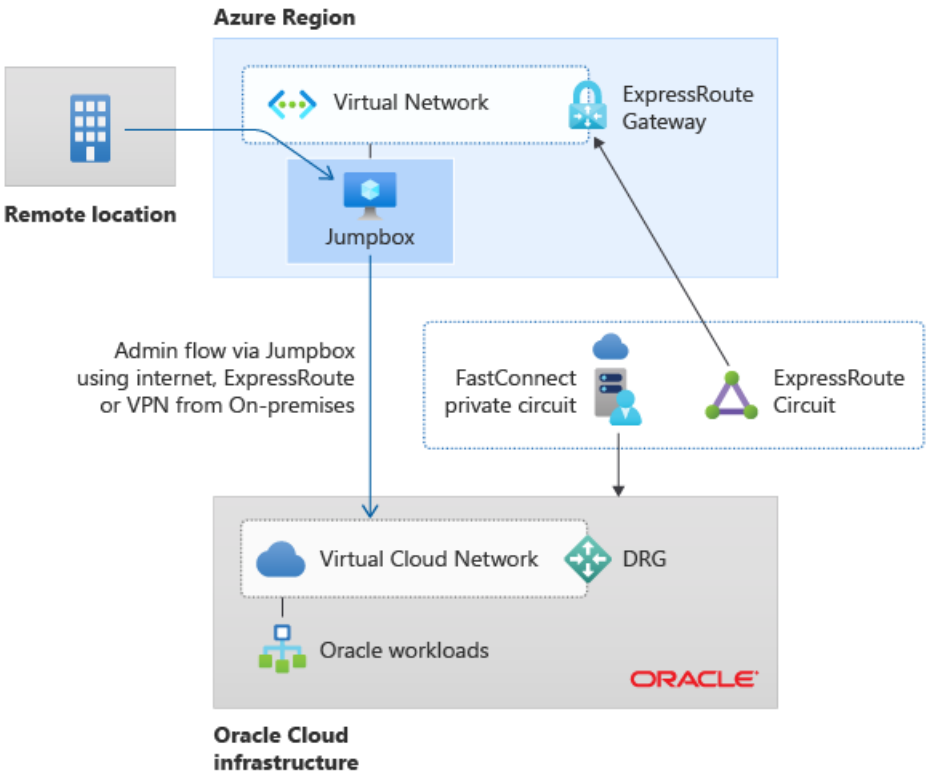


Figure 3: Managing OCI resources from Azure via a jumpbox.

- **From on-premises:** Use ExpressRoute Global Reach to bind an existing ExpressRoute circuit that connects on-premises to Azure, to an OCI ExpressRoute circuit that interconnects Azure to OCI. In this way, the Microsoft Enterprise Edge (MSEE) router becomes the central routing point between both ExpressRoute circuits.

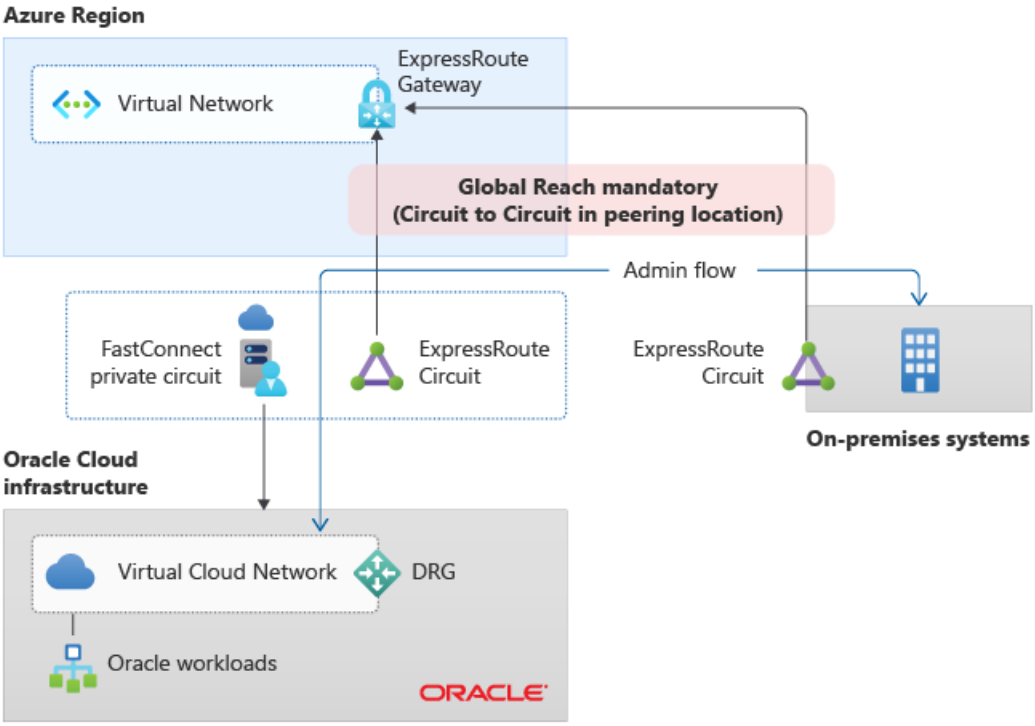


Figure 4: Managing OCI resources from on-premises via ExpressRoute Global Reach.

Next steps

For information on connectivity to other cloud providers, see Connectivity to other cloud providers.

Management and monitoring

7/20/2021 • 5 minutes to read • Edit Online

Plan platform management and monitoring

This section explores how to operationally maintain an Azure enterprise estate with centralized management and monitoring at a platform level. More specifically, it presents key recommendations for central teams to maintain operational visibility within a large-scale Azure platform.

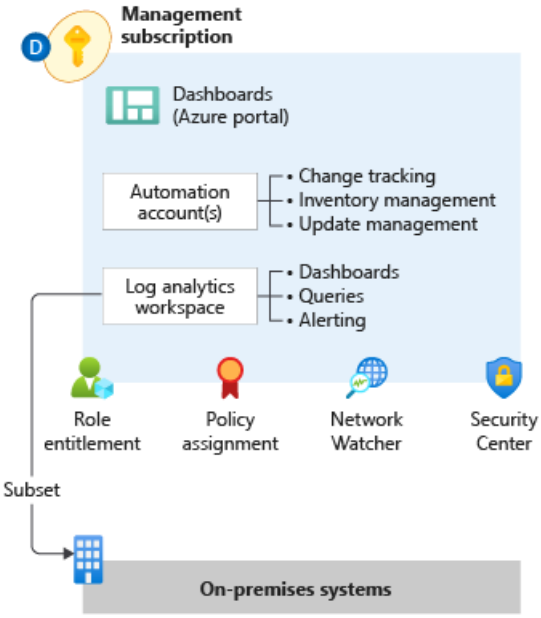


Figure 1: Platform management and monitoring.

Design considerations:

- Use an Azure Monitor Log Analytics workspace as an administrative boundary.
- Application-centric platform monitoring, encompassing both hot and cold telemetry paths for metrics and logs, respectively:
  - Operating system metrics; for example, performance counters and custom metrics
  - Operating system logs; for example, Internet Information Services, Event Tracing for Windows, and syslogs
  - Resource health events
- Security audit logging and achieving a horizontal security lens across your organization's entire Azure estate:
  - Potential integration with on-premises security information and event management (SIEM) systems such as ArcSight or the Onapsis security platform
  - Potential integration with software as a service (SaaS) offerings such as ServiceNow
  - Azure activity logs
  - Azure Active Directory (Azure AD) audit reports
  - Azure diagnostic services, logs, and metrics; Azure Key Vault audit events; network security group (NSG) flow logs; and event logs
  - Azure Monitor, Azure Network Watcher, Azure Security Center, and Azure Sentinel
- Azure data retention thresholds and archiving requirements:

    - The default retention period for Azure Monitor Logs is 30 days, with a maximum of two years.
    - The default retention period for Azure AD reports (premium) is 30 days.
    - The default retention period for the Azure diagnostic service is 90 days.
- Operational requirements:
    - Operational dashboards with native tools such as Azure Monitor Logs or third-party tooling
    - Controlling privileged activities with centralized roles
    - Managed identities for Azure resources for access to Azure services
    - Resource locks to protect editing and deleting resources

Design recommendations:

- Use a single monitor logs workspace to manage platforms centrally except where Azure role-based access control (Azure RBAC), data sovereignty requirements and data retention policies mandate separate workspaces. Centralized logging is critical to the visibility required by operations management teams. Logging centralization drives reports about change management, service health, configuration, and most other aspects of IT operations. Converging on a centralized workspace model reduces administrative effort and the chances for gaps in observability.

  In the context of the enterprise-scale architecture, centralized logging is primarily concerned with platform operations. This emphasis doesn't prevent the use of the same workspace for VM-based application logging. With a workspace configured in resource-centric access control mode, granular Azure RBAC is enforced to ensure application teams will only have access to the logs from their resources. In this model, application teams benefit from the use of existing platform infrastructure by reducing their management overhead. For any non-compute resources such as web apps or Azure Cosmos DB databases, application teams can use their own Log Analytics workspaces and configure diagnostics and metrics to be routed here.

- Export logs to Azure Storage if log retention requirements exceed two years. Use immutable storage with a write-once, read-many policy to make data non-erasable and non-modifiable for a user-specified interval.

- Use Azure Policy for access control and compliance reporting. Azure Policy provides the ability to enforce organization-wide settings to ensure consistent policy adherence and fast violation detection. For more information, see Understand Azure Policy effects.

- Monitor in-guest virtual machine (VM) configuration drift using Azure Policy. Enabling guest configuration audit capabilities through policy helps application team workloads to immediately consume feature capabilities with little effort.

- Use Update Management in Azure Automation as a long-term patching mechanism for both Windows and Linux VMs. Enforcing Update Management configurations via Azure Policy ensures that all VMs are included in the patch management regimen and provides application teams with the ability to manage patch deployment for their VMs. It also provides visibility and enforcement capabilities to the central IT team across all VMs.

- Use Network Watcher to proactively monitor traffic flows via Network Watcher NSG flow logs v2. Traffic Analytics analyzes NSG flow logs to gather deep insights about IP traffic within a virtual network and provides critical information for effective management and monitoring. Traffic Analytics provide information such as most communicating hosts and application protocols, most conversing host pairs, allowed or blocked traffic, inbound and outbound traffic, open internet ports, most blocking rules, traffic distribution per an Azure datacenter, virtual network, subnets, or rogue networks.

- Use resource locks to prevent accidental deletion of critical shared services.

- Use deny policies to supplement Azure role assignments. Deny policies are used to prevent deploying

and configuring resources that don't match defined standards by preventing the request from being sent to the resource provider. The combination of deny policies and Azure role assignments ensures the appropriate guardrails are in place to enforce *who* can deploy and configure resources and *what* resources they can deploy and configure.

- Include service and resource health events as part of the overall platform monitoring solution. Tracking service and resource health from the platform perspective is an important component of resource management in Azure.
- Don't send raw log entries back to on-premises monitoring systems. Instead, adopt a principle that *data born in Azure stays in Azure*. If on-premises SIEM integration is required, then send critical alerts instead of logs.

Plan for application management and monitoring

To expand on the previous section, this section will consider a federated model and explain how application teams can operationally maintain these workloads.

Design considerations:

- Application monitoring can use dedicated Log Analytics workspaces.
- For applications that are deployed to virtual machines, logs should be stored centrally to the dedicated Log Analytics workspace from a platform perspective. Application teams can access the logs subject to the Azure RBAC they have on their applications or virtual machines.
- Application performance and health monitoring for both infrastructure as a service (IaaS) and platform as a service (PaaS) resources.
- Data aggregation across all application components.
- Health modeling and operationalization:
  - How to measure the health of the workload and its subsystems
  - A traffic-light model to represent health
  - How to respond to failures across application components

Design recommendations:

- Use a centralized Azure Monitor Log Analytics workspace to collect logs and metrics from IaaS and PaaS application resources and control log access with Azure RBAC.
- Use Azure Monitor metrics for time-sensitive analysis. Metrics in Azure Monitor are stored in a time-series database optimized to analyze time-stamped data. These metrics are well suited for alerts and detecting issues quickly. They can also tell you how your system is performing. They typically need to be combined with logs to identify the root cause of issues.
- Use Azure Monitor Logs for insights and reporting. Logs contain different types of data that's organized into records with different sets of properties. They're useful for analyzing complex data from a range of sources, such as performance data, events, and traces.
- When necessary, use shared storage accounts within the landing zone for Azure diagnostic extension log storage.
- Use Azure Monitor alerts for the generation of operational alerts. Azure Monitor alerts unify alerts for metrics and logs and use features such as action and smart groups for advanced management and remediation purposes.

Enterprise-scale business continuity and disaster recovery

8/25/2021 • 2 minutes to read • Edit Online

Your organization or enterprise needs to design suitable, platform-level capabilities that application workloads can consume to meet their specific requirements. Specifically, these application workloads have requirements pertaining to recover time objective (RTO) and recovery point objective (RPO). Be sure that you capture disaster recovery (DR) requirements in order to design capabilities appropriately for these workloads.

Design considerations:

Consider the following factors:

- Application and data availability requirements, and the use of active-active and active-passive availability patterns (such as workload RTO and RPO requirements).
- Business continuity and DR for platform as a service (PaaS) services, and the availability of native DR and high-availability features.
- Support for multiregion deployments for failover purposes, with component proximity for performance reasons.
- Application operations with reduced functionality or degraded performance in the presence of an outage.
- Workload suitability for Availability Zones or availability sets.
    - Data sharing and dependencies between zones.
    - The impact of Availability Zones on update domains compared to availability sets and the percentage of workloads that can be under maintenance simultaneously.
    - Support for specific virtual machine (VM) stock-keeping units with Availability Zones.
    - Using Availability Zones is required if Microsoft Azure ultra disk storage is used.
- Consistent backups for applications and data.
    - VM snapshots and using Azure Backup and Recovery Services vaults.
    - Subscription limits restricting the number of Recovery Services vaults and the size of each vault.
    - Geo-replication and DR capabilities for PaaS services.
- Network connectivity if a failover occurs.
    - Bandwidth capacity planning for Azure ExpressRoute.
    - Traffic routing if a regional, zonal, or network outage occurs.
- Planned and unplanned failovers.
    - IP address consistency requirements and the potential need to maintain IP addresses after failover and failback.
    - Maintained engineering DevOps capabilities.
- Azure Key Vault DR for application keys, certificates, and secrets.

Design recommendations:

The following are best practices for your design:

- Employ Azure Site Recovery for Azure-to-Azure Virtual Machines disaster recovery scenarios. This enables you to replicate workloads across regions.

  Site Recovery provides built-in platform capabilities for VM workloads to meet low RPO/RTO requirements through real-time replication and recovery automation. Additionally, the service supports you to run recovery drills without affecting the workloads in production. You can use Azure Policy to enable replication and also audit the protection of your VMs.

- Use native PaaS service disaster recovery capabilities.

  The built-in features provide an easy solution to the complex task of building replication and failover into a workload architecture, simplifying both design and deployment automation. An organization that has defined a standard for the services they use can also audit and enforce the service configuration through Azure Policy.

- Use Azure-native backup capabilities.

  Azure Backup and PaaS-native backup features remove the need for managing third-party backup software and infrastructure. As with other native features, you can set, audit, and enforce backup configurations with Azure Policy. This ensures that services remain compliant with the organization's requirements.

- Use multiple regions and peering locations for ExpressRoute connectivity.

  A redundant hybrid network architecture can help ensure uninterrupted cross-premises connectivity if an outage affects an Azure region or peering provider location.

- Avoid using overlapping IP address ranges for production and DR sites.

  When possible, plan for a business continuity and DR network architecture that provides concurrent connectivity to all sites. DR networks that use the same classless inter-domain routing blocks like production networks require a network failover process that can complicate and delay application failover if an outage occurs.

Enterprise-scale security governance and compliance

8/25/2021 • 10 minutes to read • Edit Online

This article defines encryption and key management, helps plan for governance, defines security monitoring and an audit policy, and helps plan for platform security. At the end of this guidance, you can refer to a table that describes a framework to assess enterprise security readiness of Azure services.

Define encryption and key management

Encryption is a vital step toward ensuring data privacy, compliance, and data residency in Microsoft Azure. It's also one of the most important security concerns of many enterprises. This section covers design considerations and recommendations as they pertain to encryption and key management.

Design considerations:

- Subscription and scale limits as they apply to Azure Key Vault: Key Vault has transaction limits for keys and secrets. To throttle transactions per vault in a certain period, see Azure limits.
- Key Vault serves a security boundary because access permissions for keys, secrets, and certificates are at the vault level. Key Vault access policy assignments grant permissions separately to keys, secrets, or certificates. They don't support granular, object-level permissions like a specific key, secret, or certificate key management.
- You can isolate application-specific and workload-specific secrets and shared secrets, as appropriate control access.
- You can optimize Premium SKUs where hardware-security-module-protected keys are required. Underlying hardware security modules (HSMs) are FIPS 140-2 Level 2 compliant. Manage Azure dedicated HSM for FIPS 140-2 Level 3 compliance by considering the supported scenarios.
- Key rotation and secret expiration:
  - Certificate procurement and signing by using Key Vault about certificates.
  - Alerting/notifications and automated certificate renewals.
- Disaster recovery requirements for keys, certificates, and secrets.

  Key Vault service replication and failover capabilities: availability and redundancy.
- Monitoring key, certificate, and secret usage:

  Detecting unauthorized access by using a key vault or Azure Monitor Log Analytics workspace: monitoring and alerting.
- Delegated Key Vault instantiation and privileged access: secure access.
- Requirements for using customer-managed keys for native encryption mechanisms such as Azure Storage encryption:
  - Customer-managed keys.
  - Whole-disk encryption for virtual machines (VMs).
  - Data-in-transit encryption.
  - Data-at-rest encryption.

Design recommendations:

- Use a federated Azure Key Vault model to avoid transaction scale limits.
- Provision Azure Key Vault with the soft delete and purge policies enabled to allow retention protection for deleted objects.
- Follow a least privilege model by limiting authorization to permanently delete keys, secrets, and certificates to specialized custom Azure Active Directory (Azure AD) roles.
- Automate the certificate management and renewal process with public certificate authorities to ease administration.
- Establish an automated process for key and certificate rotation.
- Enable firewall and virtual network service endpoint on the vault to control access to the key vault.
- Use the platform-central Azure Monitor Log Analytics workspace to audit key, certificate, and secret usage within each instance of Key Vault.
- Delegate Key Vault instantiation and privileged access and use Azure Policy to enforce a consistent compliant configuration.
- Default to Microsoft-managed keys for principal encryption functionality and use customer-managed keys when required.
- Don't use centralized instances of Key Vault for application keys or secrets.
- Don't share Key Vault instances between applications to avoid secret sharing across environments.

Plan for governance

Governance provides mechanisms and processes to maintain control over your applications and resources in Azure. Azure Policy is essential to ensuring security and compliance within enterprise technical estates. It can enforce vital management and security conventions across Azure platform services. It can also supplement Azure role-based access control that controls what actions authorized users can do.

Design considerations:

- Determine what Azure policies are needed.
- Enforce management and security conventions, such as the use of private endpoints.
- Manage and create policy assignments by using policy definitions can be reused at multiple inherited assignment scopes. You can have centralized, baseline policy assignments at management group, subscription, and resource group scopes.
- Ensure continuous compliance with compliance reporting and auditing.
- Understand that Azure Policy has limits, such as the restriction of definitions at any particular scope: policy limits.
- Understand regulatory compliance policies. The policies might include HIPAA, PCI DSS, or SOC 2 Trust Services Criteria.

Design recommendations:

- Identify required Azure tags and use the append policy mode to enforce usage.
- Map regulatory and compliance requirements to Azure Policy definitions and Azure role assignments.
- Establish Azure Policy definitions at the top-level root management group so that they can be assigned at

inherited scopes.

- Manage policy assignments at the highest appropriate level with exclusions at bottom levels, if necessary.
- Use Azure Policy to control resource provider registrations at the subscription or management group levels.
- Use built-in policies where possible to minimize operational overhead.
- Assign the built-in Resource Policy Contributor role at a particular scope to enable application-level governance.
- Limit the number of Azure Policy assignments made at the root management group scope to avoid managing through exclusions at inherited scopes.

Define security monitoring and an audit policy

An enterprise must have visibility into what's happening within their technical cloud estate. Security monitoring and audit logging of Azure platform services is a key component of a scalable framework.

Design considerations:

- Data retention periods for audit data. Azure AD Premium reports have a 30-day retention period.
- Long-term archiving of logs such as Azure activity logs, VM logs, and platform as a service (PaaS) logs.
- Baseline security configuration via Azure in-guest VM policy.
- Emergency patching for critical vulnerabilities.
- Patching for VMs that are offline for extended periods of time.
- Requirements for real-time monitoring and alerting.
- Security information and event management integration with Azure Security Center and Azure Sentinel.
- Vulnerability assessment of VMs.

Design recommendations:

- Use Azure AD reporting capabilities to generate access control audit reports.
- Export Azure activity logs to Azure Monitor Logs for long-term data retention. Export to Azure Storage for long-term storage beyond two years, if necessary.
- Enable Security Center Standard for all subscriptions, and use Azure Policy to ensure compliance.
- Monitor base operating system patching drift via Azure Monitor Logs and Azure Security Center.
- Use Azure policies to automatically deploy software configurations through VM extensions and enforce a compliant baseline VM configuration.
- Monitor VM security configuration drift via Azure Policy.
- Connect default resource configurations to a centralized Azure Monitor Log Analytics workspace.
- Use an Azure Event Grid-based solution for log-oriented, real-time alerting.

Plan for platform security

Maintain a healthy security posture as you adopt Azure. Besides visibility, you can control the initial settings and changes as the Azure services evolve. Therefore, planning for platform security is key.

Design considerations:

- Shared responsibility.
- High availability and disaster recovery.
- Consistent security across Azure services for data management and analytics and control plane operations.
- Multitenancy for key platform components. The multitenancy includes Hyper-V, the HSMs underpinning Key Vault, and database engines.

Design recommendations:

- In the context of your underlying requirements, conduct a joint examination of each required service. If you want to bring your own keys, it might not be supported across all considered services. Implement relevant mitigation so that inconsistencies don't hinder desired outcomes. Choose appropriate region pairs and disaster recovery regions that minimize latency.
- Develop a security allowlist plan to assess services security configuration, monitoring, alerts, and how to integrate them with existing systems.
- Determine the incident response plan for Azure services before allowing it into production.
- Use Azure AD reporting capabilities to generate access control audit reports.
- Align your security requirements with Azure platform roadmaps to stay current with newly released security controls.
- Implement a zero-trust approach for access to the Azure platform, where appropriate.

Azure Security Benchmark

The Azure Security Benchmark includes a collection of high-impact security recommendations to help you secure most of the services you use in Azure. You can think of these recommendations as "general" or "organizational" as they are applicable to most Azure services. The Azure Security Benchmark recommendations are then customized for each Azure service, and this customized guidance is contained in service recommendations articles.

The Azure Security Benchmark documentation specifies security controls and service recommendations.

- Security controls: The Azure Security Benchmark recommendations are categorized by security controls. Security controls represent high-level vendor-agnostic security requirements, such as network security and data protection. Each security control has a set of security recommendations and instructions that help you implement those recommendations.
- Service recommendations: When available, benchmark recommendations for Azure services will include Azure Security Benchmark recommendations that are tailored specifically for that service.

Service enablement framework

As business units request to deploy workloads to Azure, you need more visibility into a workload to determine how to achieve appropriate levels of governance, security, and compliance. When a new service is required, you need to allow it. The following table provides a framework to assess enterprise security readiness of Azure services:

ASSESSMENT	CATEGORY	CRITERIA

ASSESSMENT	CATEGORY	CRITERIA
Security	Network endpoint	Does the service have a public endpoint that is accessible outside of a virtual network?
		Does it support virtual network service endpoints?
		Can Azure services interact directly with the service endpoint?
		Does it support Azure Private Link endpoints?
		Can it be deployed within a virtual network?
	Data exfiltration prevention	Does the PaaS service have a separate Border Gateway Protocol (BGP) community in Azure ExpressRoute Microsoft peering? Does ExpressRoute expose a route filter for the service?
		Does the service support Private Link endpoints?
	Enforce network traffic flow for management and data plane operations	Is it possible to inspect traffic entering/exiting the service? Can traffic be force-tunnelled with user-defined routing?
		Do management operations use Azure shared public IP ranges?
		Is management traffic directed via a link-local endpoint exposed on the host?
	Data encryption at-rest	Is encryption applied by default?
		Can encryption be disabled?
		Is encryption done with Microsoft-managed keys or customer-managed keys?
	Data encryption in-transit	Is traffic to the service encrypted at a protocol level (SSL/TLS)?
		Are there any HTTP endpoints, and can they be disabled?
		Is underlying service communication also encrypted?

ASSESSMENT	CATEGORY	CRITERIA
		Is encryption done with Microsoft-managed keys or customer-managed keys? (Is brought your own encryption supported?)
	Software deployment	Can application software or third-party products be deployed to the service?
		How is software deployment done and managed?
		Can policies be enforced to control source or code integrity?
		If software is deployable, can antimalware capability, vulnerability management, and security monitoring tools be used?
		Does the service provide such capabilities natively, such as with Azure Kubernetes Service?
Identity and access management	Authentication and access control	Are all control plane operations governed by Azure AD? Is there a nested control plane, such as with Azure Kubernetes Service?
		What methods exist to provide access to the data plane?
		Does the data plane integrate with Azure AD?
		Does authentication between Azure services use managed identities or service principals?
		Is Azure-to-IaaS (service-to-virtual-network) authentication via Azure AD?
		How are any applicable keys or shared access signatures managed?
		How can access be revoked?
	Segregation of duties	Does the service separate control plane and data plane operations within Azure AD?
	Multifactor authentication and conditional access	Is multifactor authentication enforced for user to service interactions?
Governance	Data export and import	Does service allow you to import and export data securely and encrypted?

ASSESSMENT	CATEGORY	CRITERIA
	Data privacy and usage	Can Microsoft engineers access the data?
		Is any Microsoft Support interaction with the service audited?
	Data residency	Is data contained to the service deployment region?
Operations	Monitoring	Does the service integrate with Azure Monitor?
	Backup management	Which workload data need to be backed up?
		How are backups captured?
		How frequently can backups be taken?
		How long can backups be kept for?
		Are backups encrypted?
		Is backup encryption performed with Microsoft-managed keys or customer-managed keys?
	Disaster recovery	How can the service be used in a regional redundant fashion?
		What is the attainable recovery time objective and recovery point objective?
	SKU	What SKUs are available? And how do they differ?
		Are there any features related to security for Premium SKU?
	Capacity management	How is capacity monitored?
		What is the unit of horizontal scale?
	Patch and update management	Does the service require active updating or do updates happen automatically?
		How frequently are updates applied? Can they be automated?

ASSESSMENT	CATEGORY	CRITERIA
	Audit	Are nested control plane operations captured (for example, Azure Kubernetes Service or Azure Databricks)?
		Are key data plane activities recorded?
	Configuration management	Does it support tags and provide a `put` schema for all resources?
Azure service compliance	Service attestation, certification, and external audits	Is the service PCI/ISO/SOC compliant?
	Service availability	Is the service generally available?
		In what regions is the service available?
		What is the deployment scope of the service? Is it a regional or global service?
	Service-level agreements (SLAs)	What is the SLA for service availability?
		If applicable, what is the SLA for performance?

Platform automation and DevOps

5/18/2021 • 4 minutes to read • Edit Online

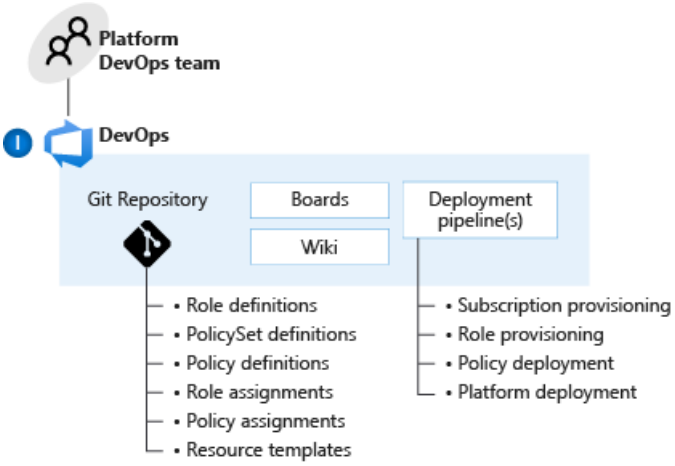


Figure 1: Platform automation and DevOps.

Planning for a DevOps approach

Many traditional IT operating models aren't compatible with the cloud, and organizations must undergo operational and organizational transformation to deliver against enterprise migration targets. You should use a DevOps approach for both application and central teams.

Design considerations:

- Where central teams are concerned, you should use pipelines for continuous integration and continuous deployment. Use the pipelines to manage policy definitions, role definitions, policy assignments, management group hierarchies, and subscriptions. These pipelines help ensure that you can operationally manage multiple subscriptions while still conforming to a desired state.
- The blanket application of a DevOps model won't instantly establish capable DevOps teams.
- Investing in engineering capabilities and resources is critical.
- You can arrange internal and external DevOps roles and functions from a variety of sources that align with your organization's strategy.
- For some legacy applications, the associated application team might not have engineering resources required to align with a DevOps strategy.

Design recommendations:

Establish a cross-functional DevOps platform team to build, manage, and maintain your enterprise-scale architecture. This team should include members from your central IT team, security, compliance, and business units to ensure that a wide spectrum of your enterprise is represented. The following list presents a recommended set of DevOps roles for a central platform team:

- **PlatformOps** (platform operations) for:
  - Subscription provisioning and delegation of required network, identity and access management, and policies.
  - Platform management and monitoring (holistic).

- Cost management (holistic).
- Platform-as-code (managing templates, scripts, and other assets).
- Responsible for overall operations on Microsoft Azure within the Azure Active Directory tenant (managing service principals, registering the Microsoft Graph API, and defining roles).

- **SecOps** (security operations)
  - Azure role-based access control (Azure RBAC) (holistic).
  - Key management (for central services, simple mail-transfer protocol, and domain controller).
  - Policy management and enforcement (holistic).
  - Security monitoring and audit (holistic).
- **NetOps** (network operations)
  - Network management (holistic).
- **AppDevOps.** Allow application owners to create and manage application resources through a DevOps model. The following list presents a recommended DevOps role for application teams:
  - Application migration or transformation.
  - Application management and monitoring.
  - Azure RBAC (application resources).
  - Security monitoring and audit (application resources).
  - Cost management (application resources).
  - Network management (application resources).
  - In some instances, you might want to break AppDevOps into more granular roles such as AppDataOps for database management or AppSecOps for more security-sensitive applications.
  - Provide a central application DevOps function to support applications that don't have existing DevOps capabilities or a business case to establish one (for example, legacy applications with minimal development capabilities).
  - Use a policy-driven approach with clear Azure RBAC boundaries to centrally enforce consistency and security across application teams. This ensures a least privilege approach is taken by using a combination of Azure RBAC and Azure Policy, and that workloads are compliant with Azure Policy assignments at all times.
  - To accelerate Azure adoption, the central platform team should establish a common set of templates and libraries for application teams to draw upon. For example, horizontal (cross-function) guidance can help to support migrations through subject matter expertise and to ensure alignment with the overall target enterprise-scale architecture.
  - Don't restrict application teams to use central artifacts or approaches because it hinders their agility. You can enforce consistent baseline configurations through a policy-driven infrastructure approach and Azure RBAC. This ensures that application (business unit) teams are flexible enough to innovate while still able to draw from a predefined set of templates.
  - Don't force application teams to use a central process or provisioning pipeline for the instantiation or management of application resources. Existing teams that already rely on a DevOps pipeline for application delivery should still be able to use the same tools they have been using. Remember that you can still use Azure Policy to maintain guardrails, independent of how resources are

deployed in Azure.

Define central and federated responsibilities

The distribution of roles, responsibilities, and trust between central IT teams and applications teams is paramount to the operational transformation your organization must undergo when adopting the cloud at scale.

Design considerations:

Central teams strive to maintain full control while application owners seek to maximize agility. The balance between these goals can greatly influence the success of the migration.

Design recommendations:

The following list presents a recommended distribution of responsibilities between the central IT team and application teams. You're striving to empower migration and transformation activities with minimal central dependencies. At the same time, you want to support the centralized governance of security and operability across the entire estate.

- **Application functions**
  - Application migration and transformation.
  - Application management and monitoring (application resources).
  - Key management (application keys).
  - Azure RBAC (application resources).
  - Security monitoring and audit (application resources).
  - Cost management (application resources).
  - Network management (application resources).
- **Central functions**
  - Architecture governance.
  - Subscription management.
  - Platform as code (management of templates, scripts, and other assets).
  - Policy management and enforcement (holistic).
  - Platform management and monitoring (holistic).
  - Azure RBAC (holistic).
  - Key management (central services).
  - Network management (including networks and network virtual appliances).
  - Security monitoring and audit (holistic).
  - Cost management (holistic).

An Azure DevOps model that's based on these recommendations provides the desired control for central teams and the migration agility required by application teams.

Implement Cloud Adoption Framework enterprise-scale landing zones in Azure

8/25/2021 • 2 minutes to read • Edit Online

When business requirements necessitate a rich initial implementation of landing zones with fully integrated governance, security, and operations control plane from the start, use the enterprise-scale example options listed here. With this approach, you can use the Azure portal or infrastructure as code to set up and configure your Azure environment. It's also possible to transition between the portal and infrastructure as code (recommended) when your organization is ready.

Reference implementation

The following table lists example reference implementations based on the recommended enterprise-scale architecture.

EXAMPLE DEPLOYMENT	DESCRIPTION	GITHUB REPO	DEPLOY TO AZURE
Enterprise-scale foundation	This is the suggested foundation for enterprise-scale adoption.	Example in GitHub	Deploy to Azure
Enterprise-scale hub and spoke	Add a hub and spoke network module to the enterprise-scale foundation.	Example in GitHub	Deploy to Azure
Enterprise-scale Virtual WAN	Add a Virtual WAN network module to the enterprise-scale foundation.	Example in GitHub	Deploy to Azure
Enterprise-scale for small enterprises	Add a hub and spoke network architecture for small organizations.	Example in GitHub	Deploy to Azure

Each reference implementation deploys platform resources to the selected target environment. Deployment details and an overview of the deployed resources can be found using the GitHub link in the table above.

Next steps

These examples provide an easy deployment option to support continued learning for the enterprise-scale approach. Before you use these examples in a production version of enterprise-scale, review the enterprise-scale architecture.

Review the enterprise-scale architecture

Testing approach for enterprise-scale

7/14/2021 • 7 minutes to read • Edit Online

> **NOTE**
> This article only applies to Microsoft Azure and not to any other Microsoft Cloud offerings such as Microsoft 365 or Microsoft Dynamics 365.

Some organizations might want to test their enterprise-scale platform deployment for Azure Policy definitions and assignments, role-based access control (RBAC) custom roles and assignments, and so on. The tests can be completed via automation by using Azure Resource Manager templates (ARM templates), AzOps, Terraform, or manually via the Azure portal. This guidance provides an approach that can be used to test changes and their impact in an enterprise-scale platform deployment.

This article can also be used with the Platform automation and DevOps critical design area guidance as it relates to the PlatformOps and Central functions teams and tasks.

This guidance is most suited to organizations with robust change management processes governing changes to the production environment management group hierarchy. The *canary* management group hierarchy can be independently used to author and test deployments before you deploy them into the production environment.

> **NOTE**
> The term "canary" is used to avoid confusion with development environments or test environments. This name is used for illustration purposes only. You might define any name you deem as appropriate for your canary enterprise-scale environment.
>
> Similarly, the term “production environment” is used throughout this guidance to refer to the management group hierarchy your organization might have in place that contains the Azure subscriptions and resources for your workloads.

Platform definition

> **IMPORTANT**
> This guidance is not for development environments or test environments that will be used by application or service owners known as landing zones, workloads, applications, or services. These are placed and handled within the production environment management group hierarchy and associated governance (RBAC and Azure Policy).
>
> This guidance is only for platform level testing and changes in the context of enterprise-scale.

Enterprise-scale helps you design and deploy the required Azure platform components to enable you to construct and operationalize landing zones at scale.

The platform resources in scope for this article and this testing approach are:

PRODUCT OR SERVICE	RESOURCE PROVIDER AND TYPE
Management groups	Microsoft.Management/managementGroups
Management groups subscription association	Microsoft.Management/managementGroups/subscriptions

PRODUCT OR SERVICE	RESOURCE PROVIDER AND TYPE
Policy definitions	Microsoft.Authorization/policyDefinitions
Policy initiative definitions or policy set definitions	Microsoft.Authorization/policySetDefinitions
Policy assignments	Microsoft.Authorization/policyAssignments
RBAC role definitions	Microsoft.Authorization/roleDefinitions
RBAC role assignments	Microsoft.Authorization/roleAssignments
Subscriptions	Microsoft.Subscription/aliases

Example scenarios and outcomes

An example of this scenario is an organization that wants to test the impact and result of a new Azure Policy to govern resources and settings in all landing zones, as per the Policy-driven governance design principle. They don't want to make this change directly to the production environment as they're concerned about the impact it might have.

Using the canary environment to test this platform change will allow the organization to implement and review the impact and result of the Azure Policy change. This process will ensure it satisfies the organization's requirements before they implement the Azure Policy to their production environment.

A similar scenario might be a change to the Azure RBAC role assignments and Azure AD group memberships. It might require a form of testing before the changes are made in the production environment.

> **IMPORTANT**
> This is not a common deployment approach or pattern for most customers. It isn't mandatory for enterprise-scale deployments.

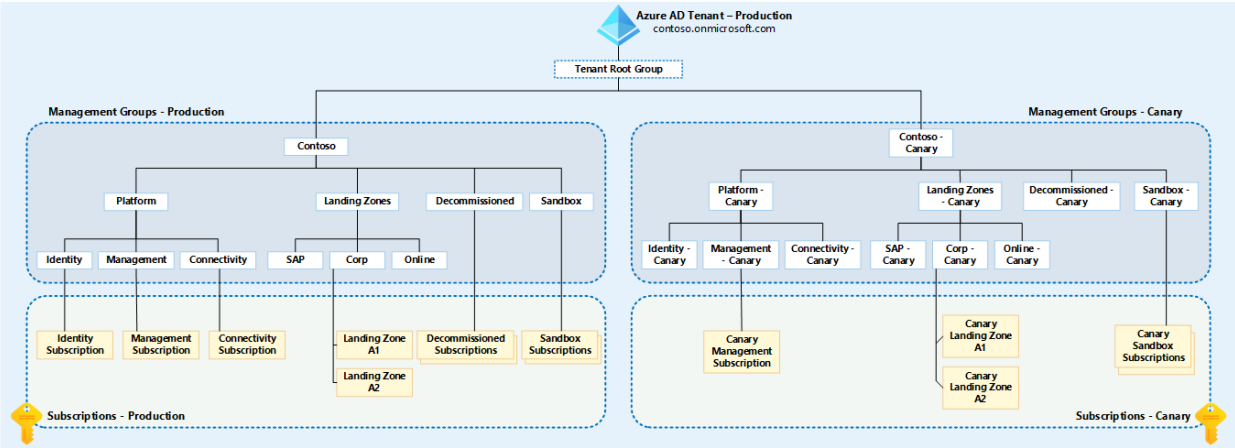


Figure 1: Canary management group hierarchy.

As the diagram shows, the entire enterprise-scale production environment management group hierarchy is duplicated under the `Tenant Root Group`. The *canary* name is appended to the management group display names and IDs. The IDs must be unique within a single Azure AD tenant.

> **NOTE**
>
> The canary environment management group display names can be the same as the production environment management group display names. This might cause confusion for users. Because of this, we recommend to append the name "canary" to the display names, as well as to their IDs.

The canary environment management group hierarchy is then used to simplify testing of the following resource types:

- Management groups
  - Subscription placement
- RBAC
  - Roles (built-in and custom)
  - Assignments
- Azure Policy
  - Definitions (built-in and custom)
  - Initiatives, also known as set definitions
  - Assignments

What if you don't want to deploy the entire canary environment management group hierarchy?

If you don't want to deploy the entire canary environment management group hierarchy, you can test platform resources within the production environment hierarchy by using sandbox subscriptions as shown in the diagram.

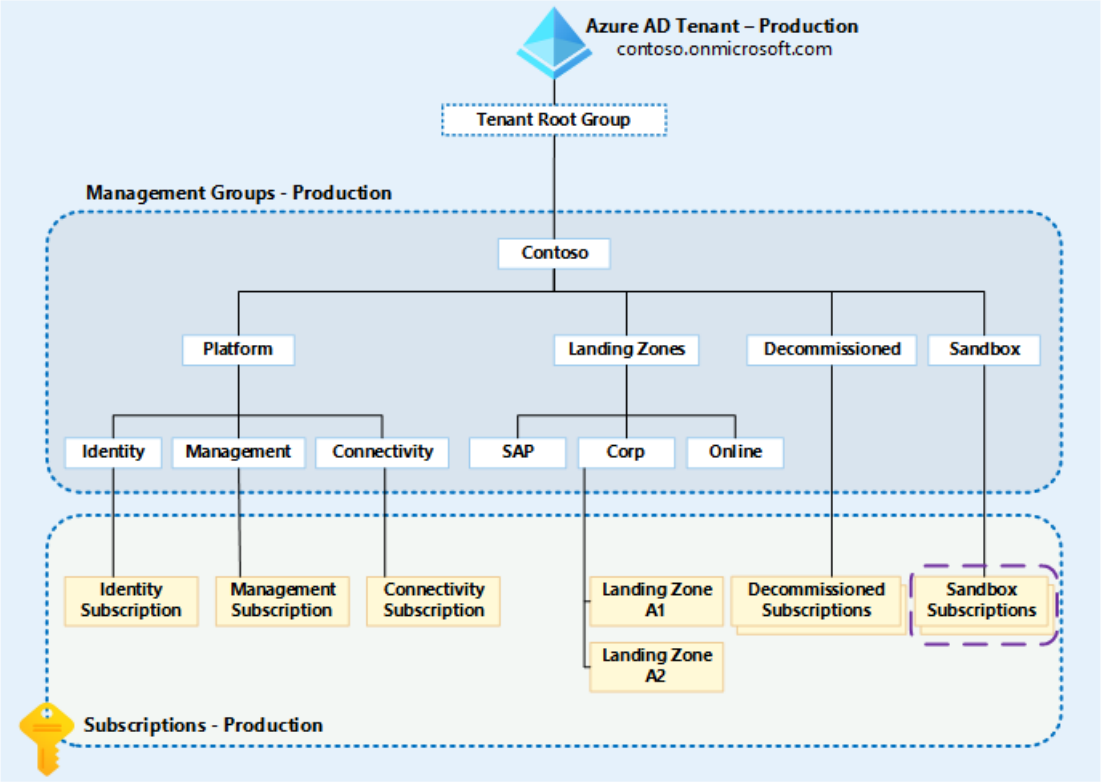


Figure 2: Enterprise-scale management group hierarchy highlighting sandboxes.

To test Azure Policy and RBAC in this scenario, you need a single Azure subscription with the Owner RBAC role assigned to the identity you wish to complete the testing as, for example, User Account, Service Principal, or Managed Service Identity. This configuration will allow you to author, assign, and remediate Azure Policy

definitions and assignments within the scope of the sandbox subscription only.

This sandbox approach can also be used for RBAC testing within the subscription, for example, if you're developing a new custom RBAC role to grant permissions for a particular use case. This testing can all be done in the sandbox subscription and tested before you create and assign roles higher up in the hierarchy.

A benefit of this approach is that the sandbox subscriptions can be used for the time that they're required, and then deleted from the environment.

However, this approach doesn't allow you to test with the inheritance of RBAC and Azure policies from the management group hierarchy.

Using a single Azure AD tenant

Considerations to take into account when you use a single Azure AD tenant are:

- Follows Enterprise-scale design recommendations for Azure AD Tenants.
- As per the Cloud Adoption Framework Azure best practices, standardize on a single directory and identity guidance, single Azure AD tenants are best practice for most.
  - In a single Azure AD tenant, you can use the different Azure AD groups for both production environments and canary enterprise-scale environments, with the same users, assigned to their relevant management group hierarchy within the same Azure AD tenant.
- Increased or duplicated Azure AD licensing costs because of multiple identities across different Azure AD tenants.
  - This point is especially relevant to customers who use Azure AD Premium features.
- RBAC changes will be more complex in both canary environments and production environments, as it's likely that the users and groups aren't identical across both Azure AD tenants.
  - Furthermore, the users and groups IDs won't be the same across Azure AD tenants because of them being globally unique.
- Reduces complexity and management overhead caused by managing multiple Azure AD Tenants.
  - Privileged users that must maintain access and sign in to separate tenants to perform testing might make changes to the production environment accidentally, instead of making changes to the canary environment and vice versa.
- Reduces the likelihood of configuration drift and deployment failures.
- Doesn’t require extra security and break-glass or emergency access processes to be created.
- Reduces friction and the time required to implement changes to the enterprise-scale deployment.

Implementation guidance

Below is guidance on how to implement and use the canary management group hierarchy for enterprise-scale alongside a production environment management group hierarchy.

1. Use separate Azure AD Service Principals (SPNs) or Managed Service Identities (MSIs) that are granted permissions over the relevant production environment or canary environment management group hierarchy.
   - This guidance follows the principle of least privilege (PoLP)
2. Use separate folders within a git repository, branches, or repositories to hold the Infrastructure-as-Code for the production environment and canary environment enterprise-scale deployments.
   - Using the relevant Azure AD Service Principals (SPNs) or Managed Service Identities (MSIs) as part of the CI/CD pipelines depending on which hierarchy is being deployed.
3. Implement git branch policies or security for the canary environment as you have in place for the production environment.
   - Consider reducing the number of approvers and checks for the canary environment to fail-fast.

4. Use the same Azure Pipelines or GitHub actions that use environment variables to change which hierarchy is being deployed. Another option is to clone the pipelines and amend the hard-coded settings to define which hierarchy is being deployed.
    - Using Azure Pipelines DevOps templates or GitHub Actions Workflow Templates will help prevent the "Don't Repeat Yourself" (DRY) principle.
5. Have a set of canary subscriptions under a separate EA department and account that can be moved around the canary management group hierarchy as needed.
    - It might be beneficial to have a set of resources always deployed into the canary environment subscriptions.
    - It might be helpful to have Infrastructure-as-Code templates such as ARM templates, Bicep, or Terraform, that create a set of resources that enable validation of changes in the canary environment.
6. Send all Azure activity logs for all Azure subscriptions, including any canary environment subscriptions, to the production environment Azure Log Analytics workspace as per the enterprise-scale design recommendations.

Terraform module for Cloud Adoption Framework enterprise-scale

5/24/2021 • 4 minutes to read • Edit Online

Azure provides native services for deploying your landing zones. Other third-party tools can also help with this effort. One such tool that customers and partners often use to deploy landing zones is Terraform by HashiCorp.

This section shows how to use the official Terraform module for Cloud Adoption Framework enterprise-scale to accelerate your journey to managing Azure landing zones at scale using Terraform.

Purpose of enterprise-scale landing zones

The Terraform Module for Cloud Adoption Framework enterprise-scale provides a simplified approach for deploying the Management Group and Subscription organization hierarchy from the enterprise scale reference architecture using Terraform, enabling:

1. Managed resource hierarchy for *subscription* organization using *management groups*.
2. A scalable security and governance model using *Azure Policy* and *Access Control (IAM)*, with an extensive library of custom definitions ready to assign.
3. Enforced security and governance settings across *subscriptions* through *management group* inheritance.

By packaging these capabilities into a single Terraform module, it becomes easier to build and enforce consistency across the Azure platform when operating at scale.

Use standard modules

Reuse of components is a fundamental principle of infrastructure as code. Modules are instrumental in defining standards and consistency across resource deployment within and across environments. This module is published to the official Terraform registry and is verified by HashiCorp.

Architecture diagram

The Terraform module for Cloud Adoption Framework enterprise-scale deploys the following components from the enterprise scale reference architecture:

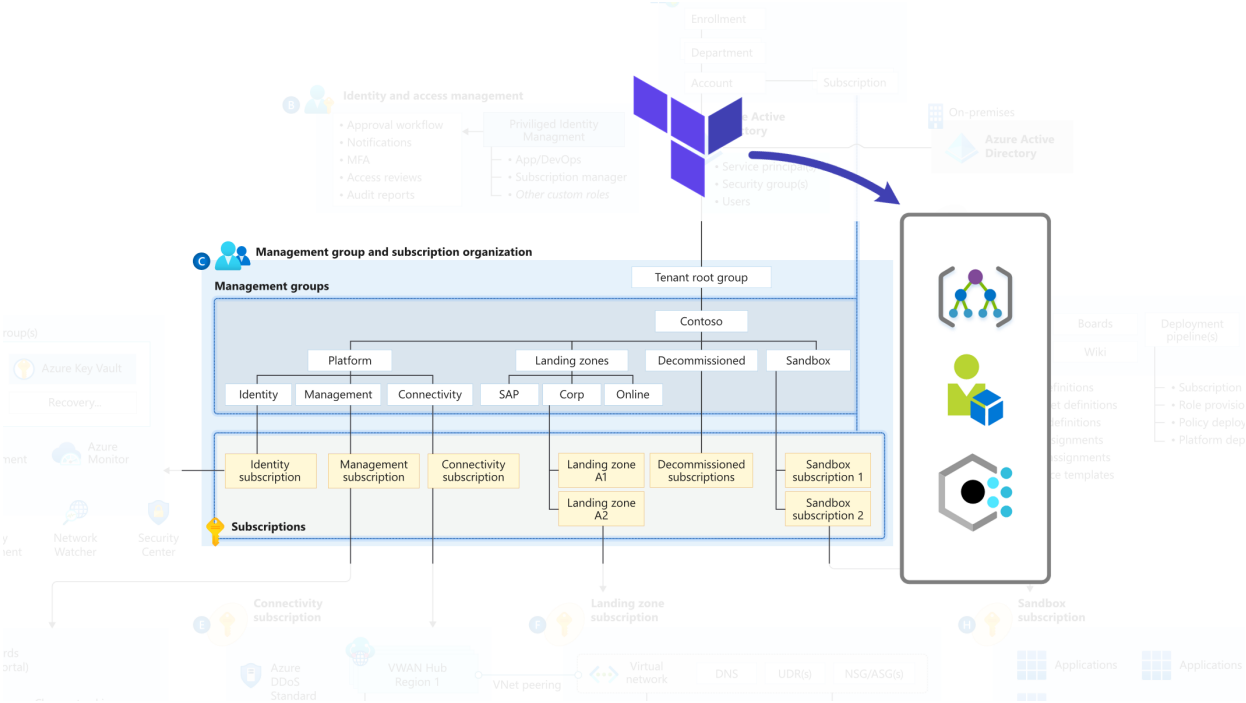


Figure 1: Overview of resources deployed by the Terraform module for Cloud Adoption Framework enterprise-scale.

Capabilities

The components deployed and their purpose include the following:

COMPONENT	RESPONSIBILITY
Management groups	Core management groups provide the resource hierarchy foundations of the enterprise-scale reference architecture: • Optional management groups for demo landing zones (SAP, corp and online) • Optional management groups for custom landing zones (define your own)
Azure Policy	Azure Policy to provide security and governance across the platform: • Custom policy definitions and policy set definitions to cover common governance patterns not covered by the built-in policies • Create and assign policies at any scope within the management group hierarchy to ensure compliance is enforced through inheritance • Expand with your own custom definitions to meet your specific security and governance requirements
Access control (IAM) settings	Create and assign roles in the resource hierarchy to ensure compliance with RBAC policies: • Create custom role definitions to ensure principle of least privilege • Create role assignments at the right scope to ensure platform and application teams have the right permissions
Subscriptions	Subscriptions are assigned to management groups: • Enables policy and RBAC inheritance • Ensures compliance with platform security and governance requirements

Customize and deploy

To simplify getting started with this module, it has been published to the Terraform registry. This allows you to consume it directly from the registry as a re-usable module and can benefit from updates as they get released.

The only dependencies for this module are as follows:

- Terraform (*recommended version 0.13.2 and later*)
- AzureRM provider (*recommended version 2.34.0 and later*)

> **IMPORTANT**
> There are known issues with some Terraform and AzureRM provider version combinations. Some of these are due to new bugs being introduced which have since been remediated, whilst others are transient errors which can typically be resolved by re-running your deployment. We generally recommend pinning to specific versions, and testing thoroughly before upgrading. As each new version of the module is released, the project team plan to rebase the module to ensure compatibility with the latest Terraform and AzureRM provider versions.

Simple example

The following is a simple starting configuration for your `main.tf` root module:

> **TIP**
> Although the module has only one mandatory variable `root_parent_id`, we also recommend setting `root_id`. Changing the `root_id` value will initiate a full redeployment of all resources managed by the module, including downstream dependencies.

```
# We strongly recommend using the required_providers block to set the
# Azure Provider source and version being used.

terraform {
  required_providers {
    azurerm = {
      source  = "hashicorp/azurerm"
      version = ">= 2.46.1"
    }
  }
}

provider "azurerm" {
  features {}
}

# You can use the azurerm_client_config data resource to dynamically
# extract the current Tenant ID from your connection settings.

data "azurerm_client_config" "current" {
}

# Call the caf-enterprise-scale module directly from the Terraform Registry
# pinning to the latest version

module "enterprise_scale" {
  source  = "Azure/caf-enterprise-scale/azurerm"
  version = "0.1.0"

  root_parent_id = data.azurerm_client_config.current.tenant_id
  root_id        = "demo"

}
```

This example code will deploy the minimum recommended management group and subscription organization from the enterprise-scale reference architecture. You can then start to customize your deployment once you've got this up and running.

> **TIP**
> If you are new to Terraform, refer to this tutorial on HashiCorp Learn, covering installation and use of Terraform, and the AzureRM provider guides for information on how to configure the provider and authenticate with Azure.

Next steps

The Terraform module for Cloud Adoption Framework enterprise-scale provides an accelerated path to building out your enterprise-scale landing zones. It also provides a great deal of flexibility to expand and customize your deployment whilst retaining a simplified approach to managing the configuration of each landing zone.

To find out more, review the module on Terraform registry, and explore the GitHub wiki where we will post more examples and tutorials covering how to customize your deployment.

Review the module on Terraform registry

Transition existing Azure environments to enterprise-scale

5/5/2021 • 5 minutes to read • Edit Online

We recognize that most organizations may have an existing footprint in Azure, one or more subscriptions, and potentially an existing structure of their management groups. Depending on their initial business requirements and scenarios, Azure resources such as hybrid connectivity (for example with Site-to-Site VPN and/or ExpressRoute) may have been deployed.

This article helps organizations to navigate the right path based on an existing Azure environment transitioning into enterprise-scale. This article also describes considerations for moving resources in Azure (for example, moving a subscription from one existing management group to another management group), which will help you evaluate and plan for transitioning your existing Azure environment to enterprise-scale landing zones.

Moving resources in Azure

Some resources in Azure can be moved post creation, and there are different approaches organizations can take subject to users Azure RBAC permissions at - and across scopes. The following table outlines which resources can be moved, at which scope, and the pros/cons associated with each.

SCOPE	DESTINATION	PROS	CONS
Resources in resource groups	Can be moved to new resource group in same or different subscription	Allows you to modify resource composition in a resource group after deployment	- Not supported by all resourceTypes - Some resourceTypes have specific limitations or requirements - resourceIds are updated and impacts existing monitoring, alerts, and control plane operations - Resource groups are locked during the move period - Requires assessment of policies and RBAC pre and post-move operation
Subscriptions in a tenant	Can be moved to different management groups	No impact to existing resources within the subscription, as no resourceId values will be changed	Requires assessment of policies and RBAC pre and post-move operation

To understand which move strategy you should use, we will go through examples of both:

Subscription move

The common use cases for moving subscriptions are to organize subscriptions into management groups or when transferring subscriptions to a new Azure Active Directory tenant. Subscription moves for enterprise-scale focuses on moving subscriptions to management groups. Moving a subscription to a new tenant is mainly for transferring billing ownership. For more guidance about how to move subscriptions across management

groups in the same tenant, see Moving management groups and subscriptions.

Azure RBAC requirements

To assess a subscription prior to a move, it is important that the user has the appropriate Azure RBAC such as being an owner on the subscription (direct role assignment), and has write permission on the target management group (built-in roles that support this are the Owner role, the Contributor role, and the Management Group Contributor role).

If the user has an inherited Owner role permission on the subscription from an existing management group, the subscription can only be moved to the management group where the user has been assigned the Owner role.

Policy

Existing subscriptions may be subject to Azure policies assigned either directly, or at the management group where they are currently located. It is important to assess current policies, and the policies that may exist in the new management group/management group hierarchy.

Azure Resource Graph can be used to perform an inventory of existing resources and compare their configuration with the policies existing at the destination.

Once subscriptions are moved to a management group with existing Azure RBAC and policies in place, consider the following options:

- Any Azure RBAC that is inherited to the moved subscriptions can take up to 30 minutes before the user tokens in the management group cache are refreshed. To expedite this process, you can refresh the token by signing out and in or request a new token.
- Any policy where the assignment scope includes the moved subscriptions, will perform audit operations only on the existing resources. More specifically:
  - Any existing resource in the subscription subject to a `DeployIfNotExists` policy effect will appear as noncompliant and will not be remediated automatically but requires user interaction to perform the remediation manually.
  - Any existing resource in the subscription subject to `Deny` policy effect will appear as noncompliant and will not be rejected. User must manually mitigate this result as appropriate.
  - Any existing resource in the subscription subject to `Append` and `Modify` policy effect will appear as noncompliant and requires user interaction to mitigate.
  - Any existing resource in the subscription subject to `Audit` and `AuditIfNotExist` policy effect will appear as noncompliant and requires user interaction to mitigate.
- All new writes to resources in the moved subscription is subject to the assigned policies at real-time as normal.

Resource move

The primary use cases to perform a resource move is when you want to consolidate resources into the same resource group if they share the same lifecycle, or move resources to a different subscription due to cost, ownership, or Azure RBAC requirements.

When performing a resource move, both the source resource group and the target resource group are locked (this lock will not affect any of the resources in the resource group) during the move operation, meaning you cannot add, update, or delete resources in the resource groups. A resource move operation will not change the location of the resources.

For more guidance about how to move resources across resource groups and subscriptions in the same tenant, see Move resources to a new resource group or subscription.

Before you move resources

Prior to a move operation, you must verify that the resources in scope are supported as well as assessing their

requirements and dependencies. For instance, moving a peered virtual network requires you to disable virtual network peering first, and re-enable the peering once the move operation has completed. This disable/re-enable dependency requires planning upfront to understand the impact to any existing workload that may be connected to your virtual networks.

Post-move operation

When the resources are moved into a new resource group in the same subscription, any inherited Azure RBAC and policies from management group or/and subscription scope will still apply. If you move to a resource group in a new subscription - where the subscription may be subject to other Azure RBAC and policy assignment, same guidance applies as to the move subscription scenario to validate the resource compliance and access controls.

Scenario-specific enterprise-scale landing zones in Azure

8/9/2021 • 2 minutes to read • Edit Online

Scenarios for enterprise-scale landing zones enable effective adoption and operationalization of key technologies on Azure. These specific scenarios can be used to accelerate adoption when your organization identifies business or technical requirements that require additional tier-1 technology platforms to support mission-critical workloads or business processes.

Scenarios for enterprise-scale landing zones provide specific, refined deployments which can be run on top of existing enterprise-scale landing zones.

Use the following information to prepare your landing zones for mission-critical technology platforms and any supported workloads.

Scenarios

These landing zones have been developed in the broader context of common scenarios, such as:

SCENARIO	DESCRIPTION	LANDING ZONE
Hybrid and multicloud	Guidance on shaping an organizations approach to implementing a hybrid cloud strategy.	Enterprise-scale for hybrid with Azure Arc
SAP	Guidance for migrating or adopting SAP workloads as part of your cloud strategy.	Enterprise-scale for SAP on Azure
Virtual desktop	Guidance for migrating virtual desktops, or creating new as part of a cloud-focused productivity strategy.	Enterprise-scale for Azure Virtual Desktop
Modern application platform	Guidance for how application services and containers can be integrated into your cloud adoption strategy.	Enterprise-scale for AKS

Next steps

Scenario-specific enterprise-scale landing zones provide an accelerated path to building out your scalable, secure landing zones for various workloads. Select one of the links above to get started.

Deploy a migration landing zone in Azure

8/25/2021 • 5 minutes to read • Edit Online

A migration landing zone is an environment that has been provisioned and prepared to host workloads that are being migrated from an on-premises environment into Azure.

Deploy the blueprint

Before you use the CAF Migration landing zone blueprint in the Cloud Adoption Framework, review the following design principles, assumptions, decisions, and implementation guidance. If this guidance aligns with the desired cloud adoption plan, the CAF Migration landing zone blueprint can be deployed using the deployment steps.

Deploy the blueprint sample

Design principles

This implementation option provides an opinionated approach to the common design areas shared by all Azure landing zones. See the assumptions and decisions below for addition technical detail.

Deployment options

This implementation option deploys a minimum viable product (MVP) to start a migration. As the migration progresses, the customer will follow a modular refactoring-based approach to mature the operating model in parallel guidance, using the Govern methodology and the Manage methodology to address those complex topics in parallel to the initial migration effort.

The specific resources deployed by this MVP approach are outlined in the decisions section below.

Enterprise enrollment

This implementation option doesn't take an inherent position on enterprise enrollment. This approach is designed to be applicable to customers regardless of contractual agreements with Microsoft or Microsoft partners. Prior to deployment of this implementation option, it is assumed that the customer has created a target subscription.

Identity

This implementation option assumes that the target subscription is already associated with an Azure Active Directory instance in accordance with identity management best practices

Network topology and connectivity

This implementation option creates a virtual network with subnets for gateway, firewall, jump box, and landing zone. As a next step iteration, the team would follow the networking decisions guide to implement the appropriate form of connectivity between the gateway subnet and other networks in alignment with network security best practices.

Resource organization

This implementation option creates a single landing zone, in which resources will be organized into workloads defined by specific resource groups. Choosing this minimalist approach to resource organization defers the technical decision of resource organization until the team's cloud operating model is more clearly defined.

This approach is based on an assumption that the cloud adoption effort will not exceed subscription limits. This option also assumes limited architectural complexity and security requirements within this landing zone.

If this changes through the course of the cloud adoption plan, the resource organization may need to be refactored using the guidance in the Govern methodology.

Governance disciplines

This implementation option doesn't implement any governance tooling. In the absence of defined policy automation, this landing zone should not be used for any mission critical workloads or sensitive data. It is assumed that this landing zone is being used for limited production deployment to initiate learning, iteration, and development of the overall operating model in parallel to these early stage migration efforts.

To accelerate parallel development of governance disciplines, review the Govern methodology and consider deploying the CAF Foundation blueprint in addition to the CAF Migration landing zone blueprint.

> **WARNING**
> As the governance disciplines mature, refactoring may be required. Specifically, resources may later need to be moved to a new subscription or resource group.

Operations baseline

This implementation option doesn't implement any operations. In the absence of a defined operations baseline, this landing zone should not be used for any mission critical workloads or sensitive data. It is assumed that this landing zone is being used for limited production deployment to initiate learning, iteration, and development of the overall operating model in parallel to these early stage migration efforts.

To accelerate parallel development of an operations baseline, review the Manage methodology and consider deploying the Azure server management guide.

> **WARNING**
> As the operations baseline is developed, refactoring may be required. Specifically, resources may later need to be moved to a new subscription or resource group.

Business continuity and disaster recovery (BCDR)

This implementation option doesn't implement any BCDR solution. It is assumed that the solution for protection and recover will be addressed by the development of the operations baseline.

Assumptions

This initial landing zone includes the following assumptions or constraints. If these assumptions align with your constraints, you can use the blueprint to create your first landing zone. The blueprint also can be extended to create a landing zone blueprint that meets your unique constraints.

- **Subscription limits:** This adoption effort isn't expected to exceed subscription limits.
- **Compliance:** No third-party compliance requirements are needed in this landing zone.
- **Architectural complexity:** Architectural complexity doesn't require additional production subscriptions.
- **Shared services:** No existing shared services in Azure require this subscription to be treated like a spoke in a hub and spoke architecture.
- **Limited production scope:** This landing zone could potentially host production workloads. It is not a suitable environment for sensitive data or mission-critical workloads.

If these assumptions align with your current adoption needs, then this blueprint might be a starting point for building your landing zone.

Decisions

The following decisions are represented in the landing zone blueprint.

COMPONENT	DECISIONS	ALTERNATIVE APPROACHES
Migration tools	Azure Site Recovery will be deployed and an Azure Migrate project will be created.	Migration tools decision guide
Logging and monitoring	Operational insights workspace and diagnostic storage account will be provisioned.	
Network	A virtual network will be created with subnets for gateway, firewall, jump box, and landing zone.	Networking decisions
Identity	It's assumed that the subscription is already associated with an Azure Active Directory instance.	Identity management best practices
Policy	This blueprint currently assumes that no Azure policies are to be applied.	
Subscription design	N/A - designed for a single production subscription.	Create initial subscriptions
Resource groups	N/A - designed for a single production subscription.	Scale subscriptions
Management groups	N/A - designed for a single production subscription.	Organize and manage subscriptions
Data	N/A	Choose the correct SQL Server option in Azure and Azure data store guidance
Storage	N/A	Azure Storage guidance
Naming and tagging standards	N/A	Naming and tagging best practices
Cost management	N/A	Tracking costs
Compute	N/A	Compute options

Customize or deploy a landing zone

Learn more and download a reference sample of the CAF Migration landing zone blueprint for deployment or customization from the Azure blueprint samples.

Deploy the blueprint sample

For guidance on customizations that should be made to this blueprint or the resulting landing zone, see the landing zone considerations.

Next steps

After deploying your first landing zone, you're ready to expand your landing zone.

Expand your landing zone

Deploy a CAF Foundation blueprint in Azure

8/9/2021 • 4 minutes to read • Edit Online

The CAF Foundation blueprint does not deploy a landing zone. Instead, it deploys the tools required to establish a governance MVP (minimum viable product) to begin developing your governance disciplines. This blueprint is designed to be additive to an existing landing zone and can be applied to the CAF Migration landing zone blueprint with a single action.

Deploy the blueprint

Before you use the CAF Foundation blueprint in the Cloud Adoption Framework, review the following design principles, assumptions, decisions, and implementation guidance. If this guidance aligns with the desired cloud adoption plan, the CAF Foundation blueprint can be deployed using the deployment steps.

Deploy the blueprint sample

Design principles

This implementation option provides an opinionated approach to the common design areas shared by all Azure landing zones. See the assumptions and decisions below for addition technical detail.

Deployment options

This implementation option deploys an MVP to serve as the foundation for your governance disciplines. The team will follow a modular refactoring-based approach to mature the governance disciplines using the Govern methodology.

Enterprise enrollment

This implementation option does not take an inherent position on enterprise enrollment. This approach is designed to be applicable to customers regardless of contractual agreements with Microsoft or Microsoft partners. Prior to deployment of this implementation option, it's assumed that the customer has already created a target subscription.

Identity

This implementation option assumes that the target subscription is already associated with an Azure Active Directory instance in accordance with identity management best practices.

Network topology and connectivity

This implementation option assumes the landing zone already has a defined network topology in accordance with network security best practices.

Resource organization

This implementation option demonstrates how Azure Policy can add some elements of resource organization through the application of tags. Specifically, a `CostCenter` tag will be appended to resources using Azure Policy.

The governance team should compare and contrast the elements of resource organization to be addressed by tagging versus those that should be addressed through subscription design. These fundamental decisions will inform resource organization as your cloud adoption plans progress.

To aid in this comparison early in adoption cycles, the following articles should be considered:

- Initial Azure subscriptions: At this stage of adoption scale, does your operating model require two, three, or four subscriptions?

- Scale subscriptions: As adoption scales, what criteria will be used to drive subscription scaling?
- Organize subscriptions: How will you organize subscriptions as you scale?
- Tagging standards: What other criteria need to be consistently captured in tags to augment your subscription design?

To aid in this comparison when teams are further along with cloud adoption, see the governance patterns section of the governance guide: prescriptive guidance article. This section of the prescriptive guidance demonstrates a set of patterns based on a specific narrative and operating model. That guidance also includes links to other patterns that should be considered.

Governance disciplines

This implementation demonstrates one approach to maturity in the Cost Management discipline of the Govern methodology. Specifically, it demonstrates how Azure Policy can be used to create an allow list of specific SKUs. Limiting the types and sizes of resources that can be deployed into a landing zone reduces the risk of overspending.

To accelerate parallel development of the other governance disciplines, review the Govern methodology. To continue maturing the Cost Management discipline of governance, see the Cost Management discipline guidance.

> **WARNING**
> As the governance disciplines mature, refactoring may be required. Refactoring may be required. Specifically, resources may later need to be moved to a new subscription or resource group.

Operations baseline

This implementation option does not implement any aspects of the operations baseline. In the absence of a defined operations baseline, this landing zone should not be used for any mission critical workloads or sensitive data. It is assumed that this landing zone is being used for limited production deployment to initiate learning, iteration, and development of the overall operating model in parallel to these early stage migration efforts.

To accelerate parallel development of an operations baseline, review the Manage methodology and consider deploying the Azure server management guide.

> **WARNING**
> As the operations baseline is developed, refactoring may be required. Specifically, resources may later need to be moved to a new subscription or resource group.

Business continuity and disaster recovery (BCDR)

This implementation option does not implement any BCDR solution. It is assumed that the solution for protection and recover will be addressed by the development of the operations baseline.

Assumptions

This initial blueprint assumes that the team is committed to maturing governance capabilities in parallel to the initial cloud migration efforts. If these assumptions align with your constraints, you can use the blueprint to begin the process of developing governance maturity.

- **Compliance:** No third-party compliance requirements are needed in this landing zone.
- **Limited production scope:** This landing zone could potentially host production workloads. It is not a suitable environment for sensitive data or mission-critical workloads.

If these assumptions align with your current adoption needs, then this blueprint might be a starting point for

building your landing zone.

Customize or deploy this blueprint

Learn more and download a reference sample of the CAF Foundation blueprint for deployment or customization from the Azure blueprint samples.

Deploy the blueprint sample

Next steps

After deploying your first landing zone, you're ready to expand your landing zone.

Expand your landing zone

Use Terraform to build your landing zones

5/18/2021 • 6 minutes to read • Edit Online

Azure provides native services for deploying your landing zones. Other third-party tools can also help with this effort. One such tool that customers and partners often use to deploy landing zones is Terraform by HashiCorp. This section shows how to use a sample landing zone to deploy foundational governance, accounting, and security capabilities for an Azure subscription.

Purpose of the landing zone

The Cloud Adoption Framework foundations landing zone for Terraform provides features to enforce logging, accounting, and security. This landing zone uses standard components known as Terraform modules to enforce consistency across resources deployed in the environment.

Use standard modules

Reuse of components is a fundamental principle of infrastructure as code. Modules are instrumental in defining standards and consistency across resource deployment within and across environments. The modules used to deploy this first landing zone are available in the official Terraform registry.

Architecture diagram

The first landing zone deploys the following components in your subscription:

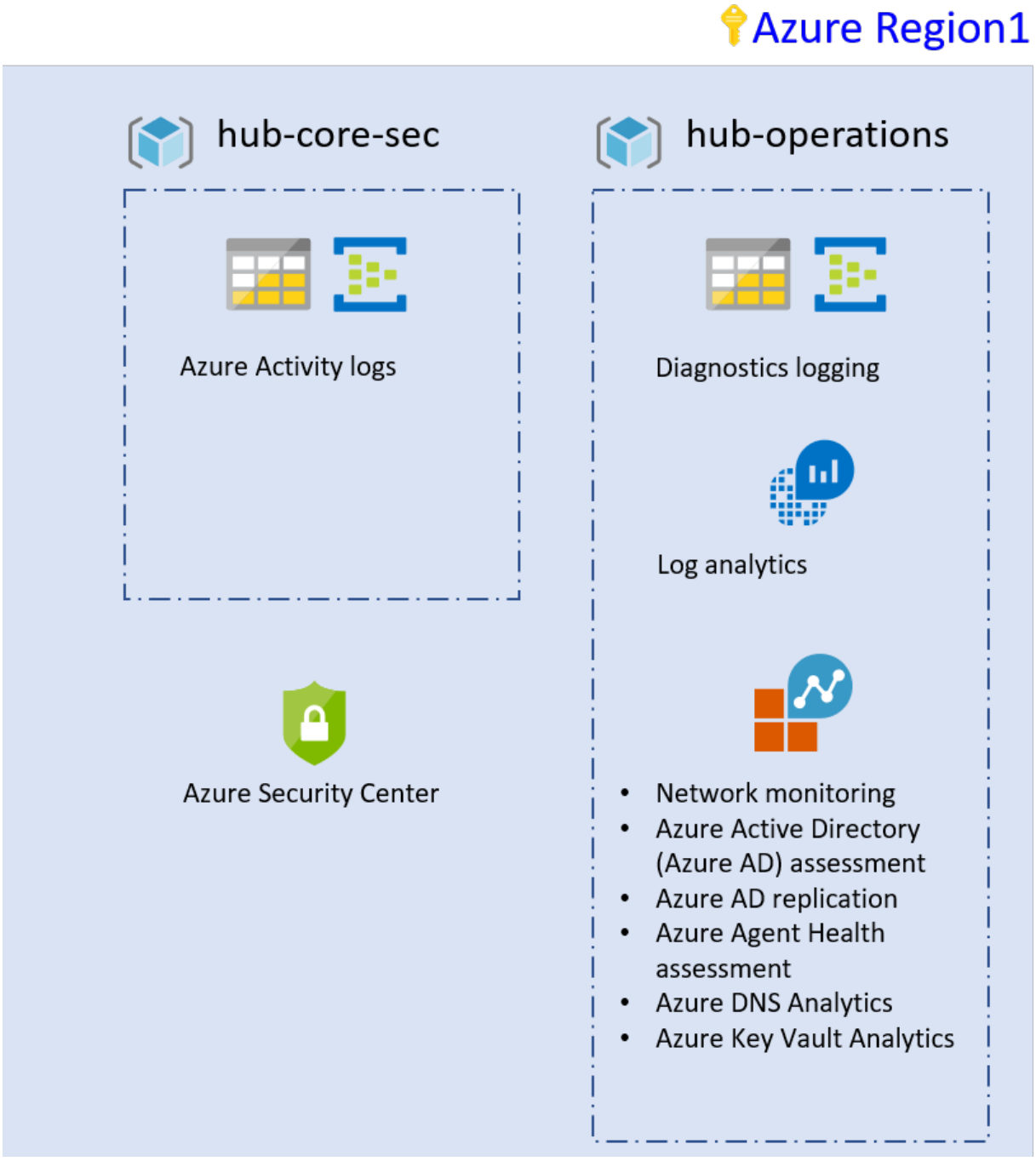


Figure 1: A foundation landing zone using Terraform.

Capabilities

The components deployed and their purpose include the following:

COMPONENT	RESPONSIBILITY
Resource groups	Core resource groups needed for the foundation
Activity logging	Auditing all subscription activities and archiving: • Storage account • Azure Event Hubs
Diagnostics logging	All operation logs kept for a specific number of days: • Storage account • Event Hubs

COMPONENT	RESPONSIBILITY
Log Analytics	Stores the operation logs. Deploy common solutions for deep application best practices review: • NetworkMonitoring • AdAssessment • AdReplication • AgentHealthAssessment • DnsAnalytics • KeyVaultAnalytics
Azure Security Center	Security hygiene metrics and alerts sent to email and phone number

Use this blueprint

Before you use the Cloud Adoption Framework foundation landing zone, review the following assumptions, decisions, and implementation guidance.

Assumptions

The following assumptions or constraints were considered when this initial landing zone was defined. If these assumptions align with your constraints, you can use the blueprint to create your first landing zone. The blueprint also can be extended to create a landing zone blueprint that meets your unique constraints.

- **Subscription limits:** This adoption effort is unlikely to exceed subscription limits. Two common indicators are an excess of 25,000 VMs or 10,000 vCPUs.
- **Compliance:** No third-party compliance requirements are needed for this landing zone.
- **Architectural complexity:** Architectural complexity doesn't require additional production subscriptions.
- **Shared services:** No existing shared services in Azure require this subscription to be treated like a spoke in a hub and spoke architecture.

If these assumptions match your current environment, this blueprint might be a good way to start building your landing zone.

Design decisions

The following decisions are represented in the CAF Terraform modules:

COMPONENT	DECISIONS	ALTERNATIVE APPROACHES
Logging and monitoring	Azure Monitor Log Analytics workspace is used. A diagnostics storage account as well as event hub is provisioned.	
Network	N/A - network is implemented in another landing zone.	Networking decisions
Identity	It's assumed that the subscription is already associated with an Azure Active Directory instance.	Identity management best practices

COMPONENT	DECISIONS	ALTERNATIVE APPROACHES
Policy	This landing zone currently assumes that no Azure policies are to be applied.	
Subscription design	N/A - designed for a single production subscription.	Create initial subscriptions
Resource groups	N/A - designed for a single production subscription.	Scale subscriptions
Management groups	N/A - designed for a single production subscription.	Organize subscriptions
Data	N/A	Choose the correct SQL Server option in Azure and Azure data store guidance
Storage	N/A	Azure Storage guidance
Naming standards	When the environment is created, a unique prefix is also created. Resources that require a globally unique name (such as storage accounts) use this prefix. The custom name is appended with a random suffix. Tag usage is mandated as described in the following table.	Naming and tagging best practices
Cost management	N/A	Tracking costs
Compute	N/A	Compute options

Tagging standards

The minimum set of tags shown below must be present on all resources and resource groups:

TAG NAME	DESCRIPTION	KEY	EXAMPLE VALUES
Business unit	Top-level division of your company that owns the subscription or workload the resource belongs to.	`BusinessUnit`	`finance`, `marketing`, `<product-name>`, `corp`, `shared`
Cost center	Accounting cost center associated with this resource.	`CostCenter`	`<cost-center-number>`
Disaster recovery	Business criticality of the application, workload, or service.	`DR`	`dr-enabled`, `non-dr-enabled`
Environment	Deployment environment of the application, workload, or service.	`Env`	`prod`, `dev`, `qa`, `staging`, `test`, `training`

TAG NAME	DESCRIPTION	KEY	EXAMPLE VALUES
Owner name	Owner of the application, workload, or service.	`Owner`	`email`
Deployment type	Defines how the resources are being maintained.	`DeploymentType`	`manual`, `terraform`
Version	Version of the blueprint deployed.	`Version`	`v0.1`
Application name	Name of the associated application, service, or workload associated with the resource.	`ApplicationName`	`<app-name>`

Customize and deploy your first landing zone

You can clone your Terraform foundation landing zone. Get started easily with the landing zone by modifying the Terraform variables. In our example, we use **blueprint_foundations.sandbox.auto.tfvars**, so Terraform automatically sets the values in this file for you.

Let's look at the different variable sections.

In this first object, we create two resource groups in the `southeastasia` region named `-hub-core-sec` and `-hub-operations` along with a prefix added at runtime.

```
resource_groups_hub = {
    HUB-CORE-SEC    = {
        name = "-hub-core-sec"
        location = "southeastasia"
    }
    HUB-OPERATIONS  = {
        name = "-hub-operations"
        location = "southeastasia"
    }
}
```

Next, we specify the regions where we can set the foundations. Here, `southeastasia` is used to deploy all the resources.

```
location_map = {
    region1   = "southeastasia"
    region2   = "eastasia"
}
```

Then, we specify the retention period for the operations logs and the Azure subscription logs. This data is stored in separate storage accounts and an event hub, whose names are randomly generated because they must be unique.

```
azure_activity_logs_retention = 365
azure_diagnostics_logs_retention = 60
```

Into the tags_hub, we specify the minimum set of tags that are applied to all resources created.

```
tags_hub = {
    environment     = "DEV"
    owner           = "Arnaud"
    deploymentType  = "Terraform"
    costCenter      = "65182"
    BusinessUnit    = "SHARED"
    DR              = "NON-DR-ENABLED"
}
```

Then, we specify the Log Analytics name and a set of solutions that analyze the deployment. Here, we retained network monitoring, Active Directory assessment and replication, DNS Analytics, and Key Vault analytics.

```
analytics_workspace_name = "lalogs"

solution_plan_map = {
    NetworkMonitoring = {
        "publisher" = "Microsoft"
        "product"   = "OMSGallery/NetworkMonitoring"
    },
    ADAssessment = {
        "publisher" = "Microsoft"
        "product"   = "OMSGallery/ADAssessment"
    },
    ADReplication = {
        "publisher" = "Microsoft"
        "product"   = "OMSGallery/ADReplication"
    },
    AgentHealthAssessment = {
        "publisher" = "Microsoft"
        "product"   = "OMSGallery/AgentHealthAssessment"
    },
    DnsAnalytics = {
        "publisher" = "Microsoft"
        "product"   = "OMSGallery/DnsAnalytics"
    },
    KeyVaultAnalytics = {
        "publisher" = "Microsoft"
        "product"   = "OMSGallery/KeyVaultAnalytics"
    }
}
```

Next, we configured the alert parameters for Azure Security Center.

```
# Azure Security Center Configuration
security_center = {
    contact_email   = "joe@contoso.com"
    contact_phone   = "+6500000000"
}
```

Take action

After you've reviewed the configuration, you can deploy the configuration as you would deploy a Terraform environment. We recommend that you use the rover, which is a Docker container that allows deployment from Windows, Linux, or macOS. You can get started with the landing zones.

Next steps

The foundation landing zone lays the groundwork for a complex environment in a decomposed manner. This

edition provides a set of simple capabilities that can be extended by adding other modules to the blueprint or layering additional landing zones on top of it.

Layering your landing zones is a good practice for decoupling systems, versioning each component that you're using, and allowing fast innovation and stability for your infrastructure as code deployment.

Future reference architectures will demonstrate this concept for a hub and spoke topology.

Review the sample foundation Terraform landing zone

Expand your landing zone

5/18/2021 • 2 minutes to read • Edit Online

This section of the Ready methodology builds on the principles of landing zone refactoring. As outlined in that article, a refactoring approach to infrastructure as code removes blockers to business success while minimizing risk. This series of articles assumes that you've deployed your first landing zone and would now like to expand that landing zone to meet enterprise requirements.

Shared architecture principles

Expanding your landing zone provides a code-first approach to embedding the following principles into the landing zone and more broadly into your overall cloud environment.



Figure 1: Shared architecture principles.

These same architecture principles are shared by Azure Advisor, the Microsoft Azure Well-Architected Framework, and the solutions in the Azure Architecture Center.

Applying these principles to your landing zone improvements

To better align with the methodologies of the Cloud Adoption Framework, the principles above are grouped into actionable landing zone improvements:

- Basic considerations: refactor a landing zone to refine hosting, fundamentals, and other foundational

elements.

- Operations expansions: add operations management configurations to improve **performance, reliability, and operational excellence**.
- Governance expansions: add governance configurations to improve **cost, reliability, security**, and consistency.
- Security expansions: add **security** configurations to improve protection of sensitive data and critical systems.

> **WARNING**
> Adoption teams who have a midterm objective (within 24 months) to **host more than 1,000 assets (applications, infrastructure, or data assets) in the cloud** should consider each of these expansions early in their cloud adoption journey. For all other adoption patterns, landing zone expansions could be a parallel iteration, allowing for early business success.

Next steps

Before refactoring your first landing zone, it is important to understand test-driven development of landing zones.

Test-driven development of landing zones

Landing zone considerations

5/24/2021 • 2 minutes to read • Edit Online

A landing zone is the basic building block of any cloud adoption environment. The term *landing zone* refers to an environment that's been provisioned and prepared to host workloads in a cloud environment like Azure. A fully functioning landing zone is the final deliverable of any iteration of the Cloud Adoption Framework's Ready methodology.

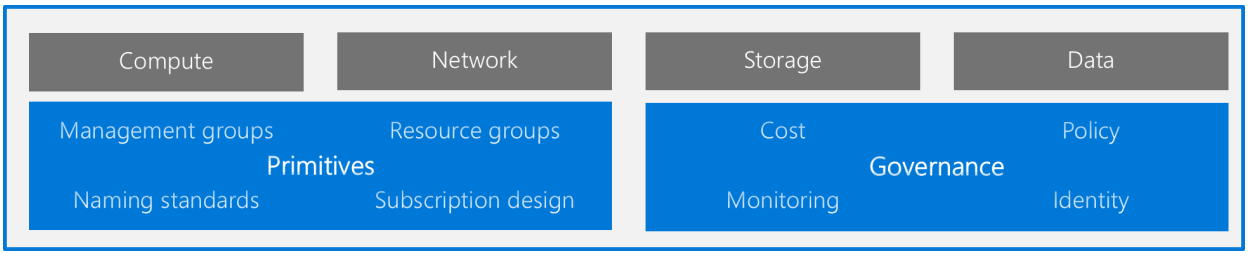


Figure 1: Landing zone considerations.

This image shows the major considerations for implementing any landing zone deployment. The considerations can be broken into three categories or types of considerations: hosting, Azure fundamentals, and governance.

Hosting considerations

All landing zones provide structure for hosting options. The structure is created explicitly through governance controls or organically through the adoption of services within the landing zone. The following articles can help you make decisions that will be reflected in the blueprint or other automation scripts that create your landing zone:

- Compute decisions: To minimize operational complexity, align compute options with the purpose of the landing zone. This decision can be enforced by using automation toolchains like Azure Policy initiatives and landing zones.
- Storage decisions: Choose the right Azure Storage solution to support your workload requirements.
- Networking decisions: Choose networking services, tools, and architectures to support your organization's workload, governance, and connectivity requirements.
- Database decisions: Determine which database technology is best suited for your workload requirements.

Azure fundamentals

Each landing zone is part of a broader solution for organizing resources across a cloud environment. Azure fundamentals are the foundational building blocks for an organization.

- Azure fundamental concepts: Learn fundamental concepts and terms that are used to organize resources in Azure and how the concepts relate to one another.
- Resource consistency decision guide: When you understand each of the fundamentals, the resource organization decision guide can help you make decisions that shape the landing zone.

Governance considerations

The Cloud Adoption Framework's Govern methodologies establish a process for governing the environment as a whole. Many use cases might require you to make governance decisions on a per-landing-zone basis. In many scenarios, governance baselines are enforced on a per-landing-zone basis even though the baselines are established holistically. It's true for the first few landing zones that an organization deploys.

The following articles can help you make governance-related decisions about your landing zone. You can factor each decision into your governance baselines.

- **Cost requirements.** Based on an organization's motivation for cloud adoption and operational commitments made about its environment, various cost management configurations might need to be changed for the landing zone.
- **Monitoring decisions.** Depending on the operational requirements for a landing zone, various monitoring tools can be deployed. The monitoring decisions article can help you determine the most appropriate tools to deploy.
- **Access control.** Azure role-based access control (RBAC) offers fine-grained, group-based access management for resources that are organized around user roles.
- **Policy decisions.** Azure Blueprints samples provide premade compliance blueprints, each with predefined policy initiatives. Policy decisions help inform a selection of the best blueprint or policy initiative based on your requirements and constraints.
- **Create hybrid cloud consistency.** Create consistent hybrid cloud solutions that give your organization the benefits of cloud innovation while maintaining many of the conveniences of on-premises management.

Review your compute options

7/15/2021 • 7 minutes to read • Edit Online

Determining the compute requirements for hosting your workloads is a key consideration as you prepare for your cloud adoption. Azure compute products and services support a wide variety of workload computing scenarios and capabilities. How you configure your landing zone environment to support your compute requirements depends on your workload's governance, technical, and business requirements.

Identify compute services requirements

As part of your landing zone evaluation and preparation, you need to identify all compute resources that your landing zone will need to support. This process involves assessing each of the applications and services that make up your workloads to determine your compute and hosting requirements. After you identify and document your requirements, you can create policies for your landing zone to control what resource types are allowed based on your workload needs.

For each application or service you'll deploy to your landing zone environment, use the following decision tree as a starting point to help you determine your compute services requirements:

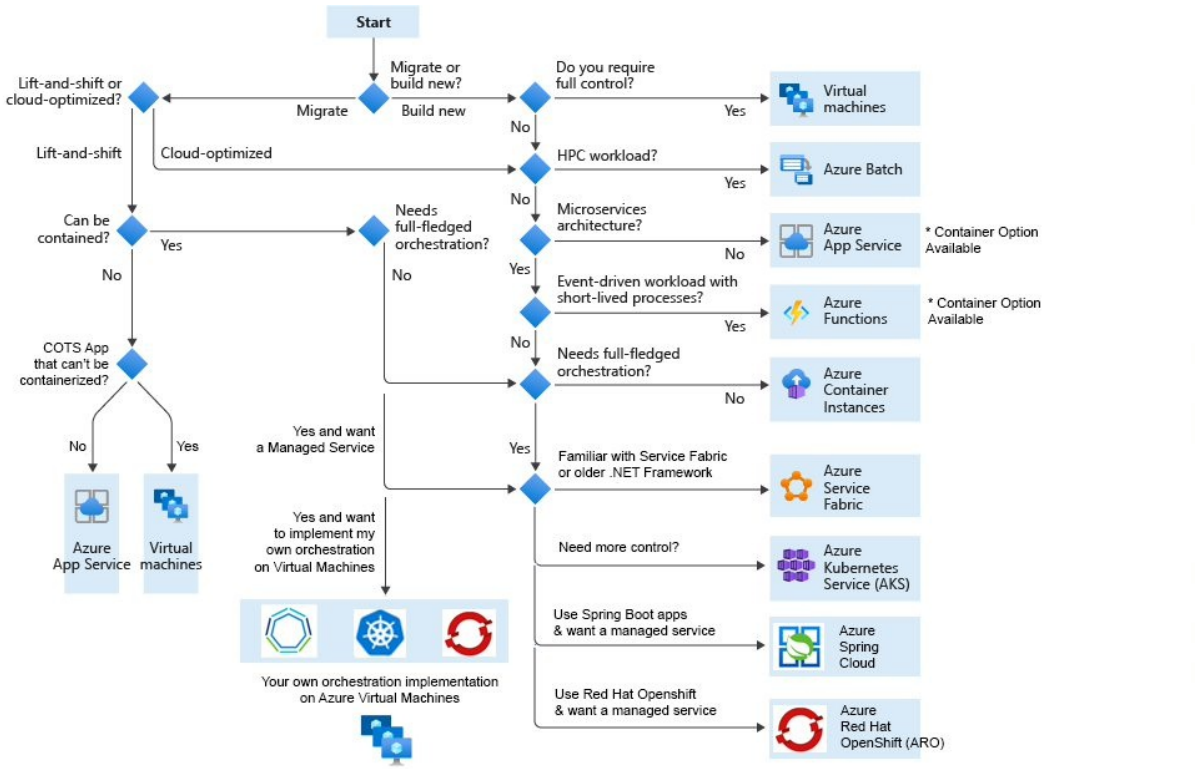


Figure 1: An Azure compute services decision tree.

Definitions:

- *Lift and shift* is a strategy for migrating a workload to the cloud without redesigning the application or making code changes. Also called rehosting. For more information, visit the Azure migration and modernization center.
- *Cloud-optimized* is a strategy for migrating to the cloud by refactoring an application to take advantage of cloud-native features and capabilities.

The output from this flowchart is a starting point for consideration. Next, perform a more detailed evaluation of

the service to see if it meets your needs.

> **NOTE**
> Learn more about how to assess compute options for each of your applications or services in the Azure application architecture guide.

Key questions

Answer the following questions about your workloads to help you make decisions based on the Azure compute services decision tree:

- **Are you building net-new applications and services or migrating from existing on-premises workloads?** Developing new applications as part of your cloud adoption efforts allows you to take full advantage of modern cloud-based hosting technologies from the design phase moving forward.
- **If you're migrating existing workloads, can they take advantage of modern cloud technologies?** Migrating on-premises workloads requires analysis. Can you easily optimize existing applications and services to take advantage of modern cloud technologies, or will a lift-and-shift approach work better for your workloads?
- **Can your applications or services take advantage of containers?** If your applications are good candidates for containerized hosting, you can take advantage of the resource efficiency, scalability, and orchestration capabilities provided by container services in Azure. Both Azure managed disks and Azure Files can be used for persistent storage in containerized applications.
- **Are your applications web- or API-based, and do they use PHP, ASP.NET, Node.js, or similar technologies?** Web apps can be deployed to managed Azure App Service instances, so you don't have to maintain virtual machines for hosting purposes.
- **Will your applications use Spring Boot?** Applications that use Spring Boot can take advantage of differentiated application level monitoring, hosted Spring middleware and application lifecycle management on Azure Spring Cloud.
- **Will you require full control over the OS and hosting environment of your workload?** If you need to control the hosting environment, including OS, disks, locally running software, and other configurations, you can use Azure Virtual Machines to host your applications and services. In addition to choosing your virtual machine sizes and performance tiers, your decisions regarding virtual disk storage will affect performance and SLAs related to your infrastructure as a service workloads. For more information, see the Azure disk storage documentation.
- **Will your workload involve high-performance computing (HPC) capabilities?** Azure Batch provides job scheduling and autoscaling of compute resources as a platform service, so it's easy to run large-scale parallel and HPC applications in the cloud.
- **Will your applications use a microservices architecture?** Applications that use a microservices-based architecture can take advantage of several optimized compute technologies. Self-contained, event-driven workloads can use Azure Functions to build scalable, serverless applications that don't need an infrastructure. For applications that require more control over the environment where microservices run, you can use container services like Azure Container Instances, Azure Kubernetes Service, and Azure Service Fabric. For applications that are built with the Spring framework, consider Azure Spring Cloud which offers hosted Spring middleware to facilitate communication between Spring microservices.

> **NOTE**
> Most Azure compute services are used in combination with Azure Storage. Consult the storage decisions guidance for related storage decisions.

Common compute scenarios

The following table illustrates a few common use scenarios and the recommended compute services for handling them:

SCENARIO	COMPUTE SERVICE
I need to provision Linux and Windows virtual machines in seconds with the configurations of my choice.	Azure Virtual Machines
I need to achieve high availability by autoscaling to create thousands of VMs in minutes.	Virtual machine scale sets
I want to simplify the deployment, management, and operations of Kubernetes.	Azure Kubernetes Service (AKS)
I need to accelerate application development by using an event-driven serverless architecture.	Azure Functions
I need to develop microservices and orchestrate containers on Windows and Linux.	Azure Service Fabric
I want to quickly create cloud applications for web and mobile by using a fully managed platform.	Azure App Service
I want to containerize applications and easily run containers by using a single command.	Azure Container Instances
I need cloud-scale job scheduling and compute management with the ability to scale to tens, hundreds, or thousands of virtual machines.	Azure Batch
I need to create highly available, scalable cloud applications and APIs that can help me focus on applications instead of hardware.	Azure Cloud Services
I need to run Spring Boot applications at cloud scale without worrying about infrastructure management.	Azure Spring Cloud

Regional availability

Azure lets you deliver services at the scale you need to reach your customers and partners **wherever they are**. A key factor in planning your cloud deployment is to determine which Azure region will host your workload resources.

Some compute options such as Azure App Service are generally available in most Azure regions while other compute services are supported only in certain regions. Some virtual machine types and their associated storage types have limited regional availability. Before you decide the regions to which you will deploy your compute resources, we recommend that you refer to the regions page to check the latest status of regional availability.

To learn more about the Azure global infrastructure, see the Azure regions page. You can also view products available by region for specific details about the overall services that are available in each Azure region.

Data residency and compliance requirements

Legal and contractual requirements related to data storage often will apply to your workloads. These requirements might vary based on the location of your organization, the jurisdiction where files and data are stored and processed, and your applicable business sector. Components of data obligations to consider include data classification, data location, and the respective responsibilities for data protection under the shared responsibility model. Many compute solutions depend on linked storage resources. This requirement also might influence your compute decisions. For help with understanding these requirements, see the white paper achieving compliant data residency and security with Azure.

Part of your compliance efforts might include controlling where your compute resources are physically located. Azure regions are organized into groups called geographies. An Azure geography ensures that data residency, sovereignty, compliance, and resiliency requirements are honored within geographical and political boundaries. If your workloads are subject to data sovereignty or other compliance requirements, you must deploy your storage resources to regions in a compliant Azure geography.

Establish controls for compute services

When you prepare your landing zone environment, you can establish controls that limit what resources each user can deploy. The controls can help you manage costs and limit security risks while still allowing developers and IT teams to deploy and configure resources that are needed to support your workloads.

After you identify and document your landing zone's requirements, you can use Azure Policy to control the compute resources that you allow users to create. Controls can take the form of allowing or denying the creation of compute resource types. For example, you might restrict users to creating only Azure App Service or Azure Functions resources. You also can use policy to control the allowable options when a resource is created, like restricting what virtual machine SKUs can be provisioned or allowing only specific VM images.

Policies can be scoped to resources, resource groups, subscriptions, and management groups. You can include your policies in Azure blueprint definitions and apply them repeatedly throughout your cloud estate.

Review your network options

8/25/2021 • 7 minutes to read • Edit Online

Designing and implementing Azure networking capabilities is a critical part of your cloud adoption efforts. You'll need to make networking design decisions to properly support the workloads and services that will be hosted in the cloud. Azure networking products and services support a wide variety of networking capabilities. How you structure these services and the networking architectures you choose depends on your organization's workload, governance, and connectivity requirements.

Identify workload networking requirements

As part of your landing zone evaluation and preparation, you need to identify the networking capabilities that your landing zone should support. This process involves assessing each of the applications and services that make up your workloads to determine their connectivity network control requirements. After you identify and document the requirements, you can create policies for your landing zone to control the allowed networking resources and configuration based on your workload needs.

For each application or service you'll deploy to your landing zone environment, use the following decision tree as a starting point to help you determine the networking tools or services to use:

CAF Ready
Networking decision tree

This decision tree serves to guide high-level initial decisions. A deep understanding of business motivations, technology strategy, timeline, and landing zone design is required to validate these decision.

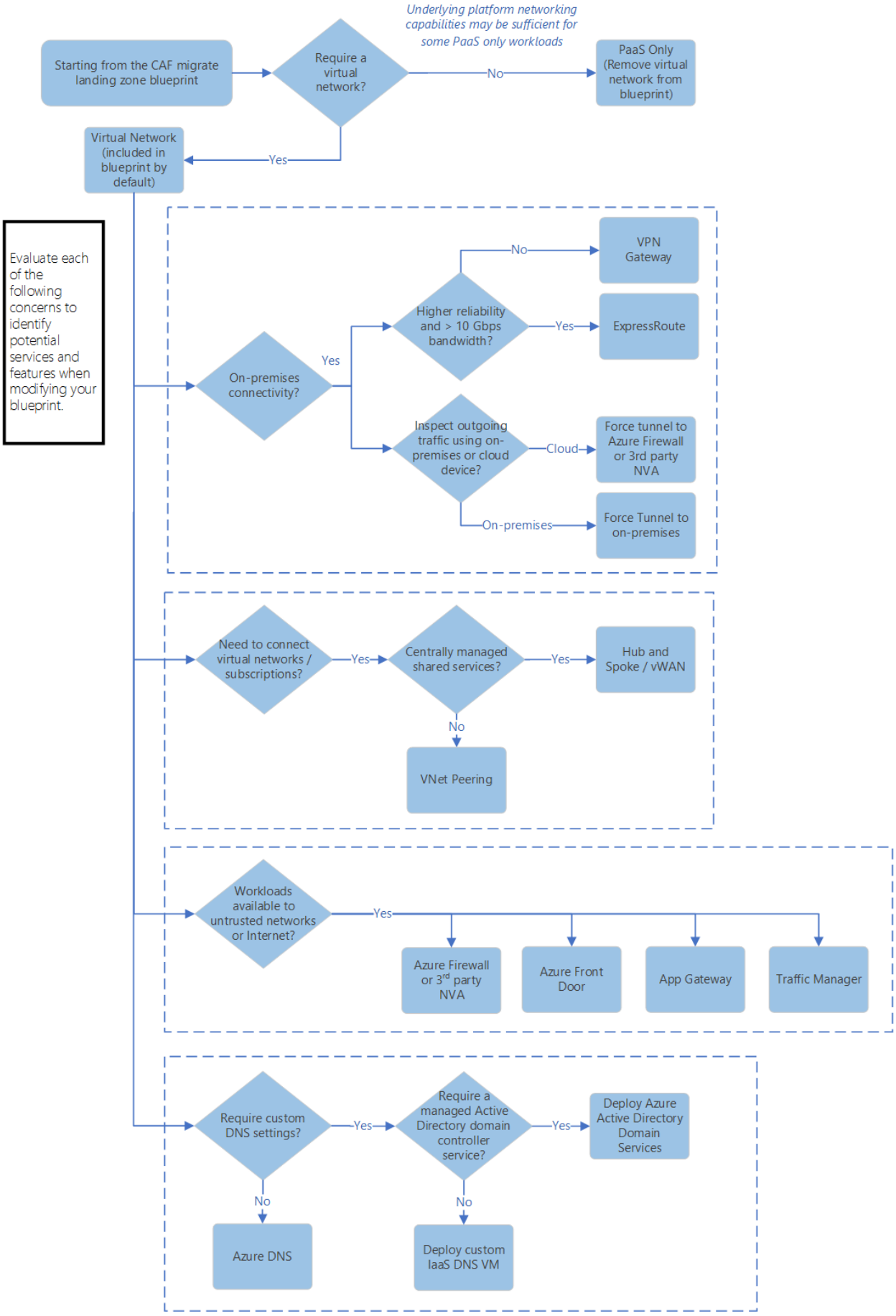


Figure 1: The Azure networking service decision tree.

Key questions

Answer the following questions about your workloads to help you make decisions based on the Azure networking services decision tree:

- **Will your workloads require a virtual network?** Managed platform as a service (PaaS) resource types use underlying platform network capabilities that don't always require a virtual network. If your workloads don't require advanced networking features and you don't need to deploy infrastructure as a service (IaaS) resources, the default native networking capabilities provided by PaaS resources might meet your workload connectivity and traffic management requirements.
- **Will your workloads require connectivity between virtual networks and your on-premises datacenter?** Azure provides two solutions for establishing hybrid networking capabilities: Azure VPN Gateway and Azure ExpressRoute. Azure VPN Gateway connects your on-premises networks to Azure through Site-to-Site VPNs similar to how you might set up and connect to a remote branch office. VPN Gateway has a maximum bandwidth of 10 Gbps. Azure ExpressRoute offers higher reliability and lower latency by using a private connection between Azure and your on-premises infrastructure. Bandwidth options for ExpressRoute range from 50 Mbps to 100 Gbps.
- **Will you need to inspect and audit outgoing traffic by using on-premises network devices?** For cloud-native workloads, you can use Azure Firewall or cloud-hosted, third-party network virtual appliances (NVAs) to inspect and audit traffic moving to or from the public internet. But many enterprise IT security policies require internet-bound outgoing traffic to pass through centrally managed devices in the organization's on-premises environment. Forced tunneling supports these scenarios. Not all managed services support forced tunneling. Services and features like App Service Environment in Azure App Service, Azure API Management, Azure Kubernetes Service (AKS), Azure SQL Managed Instance, Azure Databricks, and Azure HDInsight support this configuration when the service or feature is deployed inside a virtual network.
- **Do you need to connect multiple virtual networks?** You can use virtual network peering to connect multiple instances of Azure Virtual Network. Peering can support connections across subscriptions and regions. For scenarios where you provide services that are shared across multiple subscriptions or need to manage a large number of network peerings, consider adopting a hub and spoke networking architecture or using Azure Virtual WAN. Virtual network peering provides connectivity only between two peered networks. By default, it doesn't provide transitive connectivity across multiple peerings.
- **Will your workloads be accessible over the internet?** Azure provides services that are designed to help you manage and secure external access to your applications and services:
  - Azure Firewall
  - Network appliances
  - Azure Front Door
  - Azure Application Gateway
  - Azure Traffic Manager
- **Will you need to support custom DNS management?** Azure DNS is a hosting service for DNS domains. Azure DNS provides name resolution by using the Azure infrastructure. If your workloads require name resolution that goes beyond the features that are provided by Azure DNS, you might need to deploy other solutions. If your workloads also require Active Directory services, consider using Azure Active Directory Domain Services to augment Azure DNS capabilities. For more capabilities, you can also deploy custom IaaS virtual machines to support your requirements.

Common networking scenarios

Azure networking is composed of multiple products and services that provide different networking capabilities. As part of your networking design process, you can compare your workload requirements to the networking scenarios in the following table to identify the Azure tools or services you can use to provide these networking capabilities:

SCENARIO	NETWORKING PRODUCT OR SERVICE
I need the networking infrastructure to connect everything, from virtual machines to incoming VPN connections.	Azure Virtual Network
I need to balance inbound and outbound connections and requests to my applications or services.	Azure Load Balancer
I want to optimize delivery from application server farms while increasing application security with a web application firewall.	Azure Application Gateway Azure Front Door
I need to securely use the internet to access Azure Virtual Network through high-performance VPN gateways.	Azure VPN Gateway
I want to ensure ultra-fast DNS responses and ultra-high availability for all my domain needs.	Azure DNS
I need to accelerate the delivery of high-bandwidth content to customers worldwide, from applications and stored content to streaming video.	Azure Content Delivery Network (CDN)
I need to protect my Azure applications from DDoS attacks.	Azure DDoS protection
I need to distribute traffic optimally to services across global Azure regions, while providing high availability and responsiveness.	Azure Traffic Manager Azure Front Door
I need to add private network connectivity to access Microsoft cloud services from my corporate networks, as if they were on-premises and residing in my own datacenter.	Azure ExpressRoute
I want to monitor and diagnose conditions at a network-scenario level.	Azure Network Watcher
I need native firewall capabilities, with built-in high availability, unrestricted cloud scalability, and zero maintenance.	Azure Firewall
I need to connect business offices, retail locations, and sites securely.	Azure Virtual WAN
I need a scalable, security-enhanced delivery point for global microservices-based web applications.	Azure Front Door

Choose a networking architecture

After you identify the Azure networking services that you need to support your workloads, you also need to design the architecture that will combine these services to provide your landing zone's cloud networking infrastructure. The Cloud Adoption Framework Software Defined Networking decision guide provides details about some of the most common networking architecture patterns used on Azure.

The following table summarizes the primary scenarios that these patterns support:

SCENARIO	SUGGESTED NETWORK ARCHITECTURE
All of the Azure-hosted workloads deployed to your landing zone will be entirely PaaS-based, won't require a virtual network, and aren't part of a wider cloud adoption effort that includes IaaS resources.	PaaS-only
Your Azure-hosted workloads will deploy IaaS-based resources like virtual machines or otherwise require a virtual network, but don't require connectivity to your on-premises environment.	Cloud-native
Your Azure-hosted workloads require limited access to on-premises resources, but you're required to treat cloud connections as untrusted.	Cloud DMZ
Your Azure-hosted workloads require limited access to on-premises resources, and you plan to implement mature security policies and secure connectivity between the cloud and your on-premises environment.	Hybrid
You need to deploy and manage a large number of VMs and workloads, potentially exceeding Azure subscription limits, you need to share services across subscriptions, or you need a more segmented structure for role, application, or permission segregation.	Hub and spoke
You have many branch offices that need to connect to each other and to Azure.	Azure Virtual WAN

Azure Virtual Datacenter

In addition using one of these architecture patterns, if your enterprise IT group manages large cloud environments, consider consulting the CAF enterprise-scale landing zone. When you design your Azure-based cloud infrastructure, the CAF enterprise-scale landing zone provides a combined approach to networking, security, management, and infrastructure if you have a mid-term objective (within 24 months) to **host more than 1,000 assets (applications, infrastructure, or data assets) in the cloud**.

For organizations that meet the following criteria, you may also want to start with the CAF enterprise-scale landing zone:

- Your enterprise is subject to regulatory compliance requirements that require centralized monitoring and audit capabilities.
- You need to maintain common policy and governance compliance and centralized IT control over core services.
- Your industry depends on a complex platform that requires complex controls and deep domain expertise to govern the platform. This is most common in large enterprises within finance, oil and gas, or manufacturing.
- Your existing IT governance policies require tighter parity with existing features, even during early stage adoption.

Follow Azure networking best practices

As part of your networking design process, see these articles:

- Virtual network planning. Learn how to plan for virtual networks based on your isolation, connectivity, and location requirements.
- Azure best practices for network security. Learn about Azure best practices that can help you enhance your

network security.

- Best practices for networking when you migrate workloads to Azure. Get more guidance about how to implement Azure networking to support IaaS-based and PaaS-based workloads.

Review your storage options

8/20/2021 • 16 minutes to read • Edit Online

Storage capabilities are critical for supporting workloads and services that are hosted in the cloud. As you prepare for your cloud adoption, review this information to plan for your storage needs.

Select storage tools and services to support your workloads

Azure Storage is the Azure platform's managed service for providing cloud storage. Azure Storage is composed of several core services and supporting features. Storage in Azure is highly available, secure, durable, scalable, and redundant. Use these scenarios and considerations to choose Azure services and architectures. For more information, see Azure Storage documentation.

Key questions

Answer the following questions about your workloads to help you make decisions about your storage needs:

- **Do your workloads require disk storage to support the deployment of infrastructure as a service (IaaS) virtual machines?** Azure managed disks provide virtual disk capabilities for IaaS virtual machines.
- **Will you need to provide downloadable images, documents, or other media as part of your workloads?** Azure Blob Storage hosts static files, which are then accessible for download over the internet. For more information, see Static website hosting in Azure Storage.
- **Will you need a location to store virtual machine logs, application logs, and analytics data?** You can use Blob Storage to store Azure Monitor log data. See Storage Analytics.
- **Will you need to provide a location for backup, disaster recovery, or archiving workload-related data?** Blob Storage provides backup and disaster recovery capabilities. For more information, see Backup and disaster recovery for Azure IaaS disks.

  You can also use Blob Storage to back up other resources, like on-premises or IaaS virtual machine-hosted SQL Server data. See SQL Server Backup and Restore.
- **Will you need to support big data analytics workloads?** Azure Data Lake Storage Gen2 is built on Azure Blob Storage. Data Lake Storage Gen2 supports large-enterprise data lake functionality. It also can handle storing petabytes of information while sustaining hundreds of gigabits of throughput.
- **Will you need to provide cloud-native file shares?** Azure has two services that provide cloud-hosted file shares:
  - Azure NetApp Files provides high-performance NFS shares that are well suited to common enterprise workloads like SAP.
  - Azure Files provides file shares accessible over SMB 3.0 and HTTPS.
- **Will you need to support hybrid cloud storage for on-premises high-performance computing (HPC) workloads?** Avere vFXT for Azure is a hybrid caching solution. You can expand your on-premises storage capabilities by using cloud-based storage. Avere vFXT for Azure is optimized for read-heavy HPC workloads that involve 1,000 to 40,000 CPU cores. Avere vFXT for Azure can integrate with on-premises hardware network attached storage (NAS), Blob Storage, or both.
- **Will you need to perform large-scale archiving and syncing of your on-premises data?** Azure Data Box products are designed to help you move large amounts of data from your on-premises environment to the cloud.

    - Azure Data Box Gateway is a virtual device that is on-premises. Data Box Gateway helps you manage large-scale data migration to the cloud.
    - Azure Stack Edge accelerates processing and the secure transfer of data to Azure. If you need to analyze, transform, or filter data before you move it to the cloud, use Data Stack Edge.
- **Do you want to expand an existing on-premises file share to use cloud storage?** Azure File Sync lets you use the Azure Files service as an extension of file shares that are hosted on your on-premises Windows Server computers. The syncing service transforms Windows Server into a quick cache of your Azure file share. It allows your on-premises computers that access the share to use any protocol that's available on Windows Server.

Common storage scenarios

Azure offers multiple products and services for different storage capabilities. The following table describes potential storage scenarios and the recommended Azure services.

Block storage scenarios

SCENARIO	SUGGESTED AZURE SERVICES	CONSIDERATIONS FOR SUGGESTED SERVICES
I have bare-metal servers or virtual machines (Hyper-V or VMware) with direct attached storage running line-of-business applications.	Azure Premium SSD	For production services, Premium SSD option provides consistent low-latency coupled with high input/output operations per second (IOPS) and throughput.
I have servers that will host web and mobile apps.	Azure Standard SSD	Standard SSD IOPS and throughput might be sufficient at a lower cost than Premium SSD for CPU-bound web and application servers in production.
I have an enterprise SAN or all-flash array.	Premium SSD or Azure Ultra Disk Storage or Azure NetApp Files	Ultra Disk Storage is NVMe-based and offers submillisecond latency with high IOPS and bandwidth. Ultra Disk Storage is scalable up to 64 TB. The choice of Premium SSD or Ultra Disk Storage depends on peak latency, IOPS, and scalability requirements.
I have high-availability clustered servers, such as SQL Server FCI or Windows Server failover clustering.	Azure Files or Premium SSD or Ultra Disk Storage	Clustered workloads require multiple nodes to mount the same underlying shared storage for failover or high availability. Premium file shares offer shared storage that's mountable by using SMB. Shared block storage also can be configured on Premium SSD or Ultra Disk Storage by using partner solutions. See SIOS DataKeeper Cluster Edition.
I have a relational database or data warehouse workload, such as SQL Server or Oracle.	Premium SSD or Ultra Disk Storage	The choice of Premium SSD or Ultra Disk Storage depends on peak latency, IOPS, and scalability requirements. Ultra Disk Storage also reduces complexity by removing the need for storage pool configuration for scalability. See Mission critical performance.

SCENARIO	SUGGESTED AZURE SERVICES	CONSIDERATIONS FOR SUGGESTED SERVICES
I have a NoSQL cluster such as Cassandra or MongoDB.	Premium SSD	Azure disk storage Premium SSD provides consistent low-latency coupled with high IOPS and throughput.
I have containers with persistent volumes.	Azure Files or Standard SSD, Premium SSD, or Ultra Disk Storage	File (RWX) and block (RWO) volumes driver options are available for both Azure Kubernetes Service and custom Kubernetes deployments. Persistent volumes can map to either an Azure disk storage disk or a managed Azure Files share. Choose premium versus standard options based on workload requirements for persistent volumes.
I have a data lake such as a Hadoop cluster for HDFS data.	Data Lake Storage Gen2 or Standard SSD or Premium SSD	The Data Lake Storage Gen2 feature of Blob Storage provides server-side HDFS compatibility and petabyte scale for parallel analytics. It also offers high availability and reliability. Software like Cloudera can use Premium SSD or Standard SSD on controller/worker nodes, if needed.
I have an SAP or SAP HANA deployment.	Premium SSD or Ultra Disk Storage	Ultra Disk Storage is optimized to offer submillisecond latency for tier-1 SAP workloads. Premium SSD, coupled with M-series virtual machines, offers a general-availability option.
I have a disaster recovery site with strict RPO/RTO that syncs from my primary servers.	Azure page blobs	Page blobs are used by replication software to enable low-cost replication to Azure without the need for compute virtual machines until failover occurs. For more information, see Backup and disaster recovery for Azure IaaS disks. **Note:** Page blobs support a maximum of 8 TB.

File and object storage scenarios

SCENARIO	SUGGESTED AZURE SERVICES	CONSIDERATIONS FOR SUGGESTED SERVICES
I use Windows file server.	Azure Files or Azure File Sync	With Azure File Sync, you can store rarely used data on cloud-based file shares while caching your most frequently used files on-premises. You can also keep files in sync across multiple servers. If you plan to migrate your workloads to a cloud-only deployment, Azure Files might be sufficient.

SCENARIO	SUGGESTED AZURE SERVICES	CONSIDERATIONS FOR SUGGESTED SERVICES
I have an enterprise network attached storage such as NetApp or Dell-EMC Isilon.	Azure NetApp Files or Azure Files (premium)	If you have an on-premises deployment of NetApp, consider using Azure NetApp Files to migrate your deployment to Azure. If you're using or migrating to a Windows or Linux server, consider using Azure Files. Also, if you have basic file-share needs, consider using Azure Files. For continued on-premises access, use Azure File Sync to sync file shares with on-premises file shares by using a cloud-tiering mechanism. Cloud tiering uses your server as a cache for the relevant files while scaling cold data in Azure file shares.
I have as SMB or NFS file share.	Azure Files or Azure NetApp Files	The choice of premium or standard Azure Files tiers depends on IOPS, throughput, and your need for latency consistency. If you have an on-premises deployment of NetApp, consider using Azure NetApp Files. If you need to migrate your access control lists and timestamps to the cloud, Azure File Sync can bring these settings to your Azure file shares.
I have an on-premises object storage system for petabytes of data, such as Dell-EMC ECS.	Blob Storage	Azure Blob Storage provides premium, hot, cool, and archive tiers to match your workload performance and cost needs.
I have a Distributed File System Replication deployment or another way of handling branch offices.	Azure Files or Azure File Sync	Azure File Sync offers multisite sync for multiple servers and native Azure file shares. Move to a fixed storage footprint on-premises by using cloud tiering.
I have a tape library for backup and disaster recovery or long-term data retention.	Blob Storage	A Blob Storage archive tier has the lowest possible cost. It might require hours to copy the offline data to a cool, hot, or Premium tier to allow access. Cool tiers provide instantaneous access at low cost.
I have file or object storage configured to receive my backups.	Blob Storage or Azure File Sync	To back up data for long-term retention with lowest-cost storage, move data to Blob Storage and use cool and archive tiers. To enable fast disaster recovery for file data on a server, sync shares to individual Azure file shares by using Azure File Sync. With Azure file share snapshots, you can restore earlier versions. Sync them back to connected servers or access them natively in the Azure file share.

SCENARIO	SUGGESTED AZURE SERVICES	CONSIDERATIONS FOR SUGGESTED SERVICES
I run data replication to a disaster recovery site.	Azure Files or Azure File Sync	Azure File Sync removes the need for a disaster recovery server and stores files in native Azure SMB shares. Fast disaster recovery rebuilds any data on a failed on-premises server quickly. You can even keep multiple server locations in sync or use cloud tiering to store only relevant data on-premises.
I manage data transfer in disconnected scenarios.	Azure Stack Edge or Data Box Gateway	Using Data Stack Edge or Data Box Gateway, you can copy data in disconnected scenarios. When the gateway is offline, it saves all files you copy in the cache, then uploads them when you're connected.
I manage an ongoing data pipeline to the cloud.	Azure Stack Edge or Data Box Gateway	Move data to the cloud from systems that are constantly generating data by having them copy that data to the storage gateway.
I have bursts of data that arrive at the same time.	Azure Stack Edge or Data Box Gateway	Manage large quantities of data that arrive at the same time. Some examples are when an autonomous car pulls into the garage or a gene sequencing machine that finishes its analysis. Copy all that data to Data Box Gateway at fast local speeds. Then let the gateway upload it as your network allows.

Plan based on data workloads

SCENARIO	SUGGESTED AZURE SERVICES	CONSIDERATIONS FOR SUGGESTED SERVICES
I want to develop a new cloud-native application that needs to persist unstructured data.	Blob Storage	
I need to migrate data from an on-premises NetApp instance to Azure.	Azure NetApp Files	
I need to migrate data from on-premises Windows file server instances to Azure.	Azure Files	
I need to move file data to the cloud but continue to primarily access the data from on-premises.	Azure Files or Azure File Sync	
I need to support burst compute. I have NFS/SMB read-heavy, file-based workloads with data assets that are on-premises while computation runs in the cloud.	Avere vFXT for Azure	IaaS scale-out NFS/SMB file caching.

SCENARIO	SUGGESTED AZURE SERVICES	CONSIDERATIONS FOR SUGGESTED SERVICES
I need to move an on-premises application that uses a local disk or iSCSI.	Azure disk storage	
I need to migrate a container-based application that has persistent volumes.	Azure disk storage or Azure Files	
I need to move file shares that aren't on Windows Server or NetApp to the cloud.	Azure Files or Azure NetApp Files	Protocol supports regional availability performance requirements snapshot and clone capabilities price sensitivity.
I need to transfer terabytes to petabytes of data from on-premises to Azure.	Azure Stack Edge	
I need to process data before transferring it to Azure.	Azure Stack Edge	
I need to support continuous data ingestion in an automated way by using local cache.	Data Box Gateway	

Learn more about Azure Storage services

After you identify the Azure tools that best match your requirements, use this detailed documentation to learn about yourself with these services:

SERVICE	DESCRIPTION
Blob Storage	Blob Storage is Microsoft's object storage solution for the cloud. Blob Storage is optimized for storing massive amounts of unstructured data. Unstructured data is data that doesn't adhere to a specific data model or definition, such as text or binary data. Use Blob Storage for the following needs: • Serving images or documents directly to a browser. • Storing files for distributed access. • Streaming video and audio. • Writing to log files. • Storing data for backup and restore, disaster recovery, and archiving. • Storing data for analysis by an on-premises or Azure-hosted service.
Data Lake Storage Gen2	Blob Storage supports Data Lake Storage Gen2, Microsoft's enterprise big data analytics solution for the cloud. Data Lake Storage Gen2 offers a hierarchical file system, with the advantages of Blob Storage, including low-cost tiered storage, high availability, strong consistency, and disaster recovery capabilities.

SERVICE	DESCRIPTION
Azure disk storage	Azure disk storage offers persistent, high-performance block storage to power Azure Virtual Machines. Azure disks are highly durable, secure, and offer the industry's only single-instance service-level agreement (SLA) for virtual machines that use Premium SSD or Ultra Disk Storage. Azure disks provide high availability with availability sets and Availability Zones for your Azure Virtual Machines fault domains. Azure manages disks as a top-level resource. Azure Resource Manager capabilities are provided, such as Azure role-based access control (Azure RBAC), policy, and tagging by default.
Azure Files	Azure Files provides fully managed, native SMB file shares, without the need to run a virtual machine. You can mount an Azure Files share as a network drive to any Azure virtual machine or on-premises computer.
Azure File Sync	Azure File Sync can be used to centralize your file shares in Azure Files. Azure File Sync offers the flexibility, performance, and compatibility of an on-premises file server.
Azure NetApp Files	The Azure NetApp Files service is an enterprise-class, high-performance, metered file storage service. Azure NetApp Files supports any workload type and is highly available by default. You can select service and performance levels and set up snapshots through the service.
Azure Stack Edge	Azure Stack Edge is an on-premises network device that moves data into and out of Azure. Data Stack Edge has AI-enabled edge compute to preprocess data during upload. Data Box Gateway is a virtual version of the device but with the same data transfer capabilities.
Data Box Gateway	Data Box Gateway is a storage solution that enables you to seamlessly send data to Azure. Data Box Gateway is a virtual device based on a virtual machine provisioned in your virtualized environment or hypervisor. The virtual device is on-premises and you write data to it by using the NFS and SMB protocols. The device then transfers your data to Azure block blobs or page blobs, or to Azure Files.
Avere vFXT for Azure	Avere vFXT for Azure is a filesystem caching solution for data-intensive HPC tasks. Take advantage of cloud computing's scalability to make your data accessible, even for data that's stored in your own on-premises hardware.

Data redundancy and availability

Azure Storage has various redundancy options to help ensure durability and high availability based on your needs.

- Locally redundant storage
- Zone-redundant storage
- Geo-redundant storage (GRS)
- Read-access GRS (RA-GRS)
- Geo-zone-redundant storage (GZRS)

To learn more about these capabilities and how you can decide on the best redundancy option for your use cases, see Azure Storage redundancy.

SLAs for storage services provide guarantees that are financially backed. For more information, see SLA for managed disks, SLA for virtual machines, and SLA for storage accounts.

For help with planning the right solution for Azure disks, see Backup and disaster recovery for Azure disk storage.

Security

To help you protect your data in the cloud, Azure Storage offers several best practices for data security and encryption:

- Secure the storage account by using Azure RBAC and Azure Active Directory (Azure AD).
- Secure data in transit between an application and Azure by using client-side encryption, HTTPS, or SMB 3.0.
- Set data to be encrypted when it's written to Azure Storage by using Azure Storage encryption.
- Grant delegated access to the data objects in Azure Storage by using shared access signatures.
- Use analytics to track the authentication method that someone is using when they access storage in Azure.

These security features apply to Azure Blob Storage (block and page) and to Azure Files. For more information, see Security recommendations for Blob storage.

Azure Storage provides encryption at rest and safeguards your data. Azure Storage encryption is enabled by default for managed disks, snapshots, and images in all the Azure regions. All new managed disks, snapshots, images, and new data written to existing managed disks are encrypted at rest using keys managed by Microsoft. For more information, see Azure Storage encryption and Managed Disks and Storage Service Encryption.

Azure Disk Encryption allows you to encrypt managed disks that are attached to IaaS virtual machines at rest and in transit. Azure Key Vault stores your keys. For Windows, the drives are encrypted by using industry-standard BitLocker encryption technology. For Linux, the disks are encrypted by using the dm-crypt subsystem. The encryption process is integrated with Azure Key Vault to allow you to control and manage the disk encryption keys. For more information, see Azure Disk Encryption for virtual machines and virtual machine scale sets.

Regional availability

You can use Azure to deliver services at the scale that you need to reach your customers and partners wherever they are. Checking the regional availability of a service beforehand can help you make the right decision for your workload and customer needs. To check availability, see managed disks available by region and Azure Storage available by region.

Managed disks are available in all Azure regions that have Premium SSD and Standard SSD offerings. Ultra Disk Storage is offered in several Availability Zones. Verify the regional availability when you plan mission-critical, top-tier workloads that require Ultra Disk Storage.

Hot and cool Blob Storage, Data Lake Storage Gen2, and Azure Files storage are available in all Azure regions. Archival blob storage, premium file shares, and premium lock Blob Storage are limited to certain regions. Refer to the regions page to check the current status.

To learn more about Azure global infrastructure, see Azure geographies. Consult Products available by region for storage options available in each Azure region.

Data residency and compliance requirements

Legal and contractual requirements that are related to data storage often apply to your workloads. These

requirements depend on the location of your organization, the jurisdiction of the physical assets that host your data stores, and your business sector. Consider data classification, data location, and the respective responsibilities for data protection under the shared responsibility model. For more information, see Enabling Data Residency and Data Protection in Microsoft Azure Regions.

Part of your compliance efforts might include controlling where your database resources are physically located. Azure regions are organized into groups called geographies. An Azure geography ensures that data residency, sovereignty, compliance, and resiliency requirements are honored within geographical and political boundaries. If your workloads are subject to data sovereignty or other compliance requirements, deploy your storage resources to regions that are in a compliant Azure geography. For more information, see Azure geographies.

Next steps

Review your data options

Review your data options

5/18/2021 • 6 minutes to read • Edit Online

When you prepare your landing zone environment for your cloud adoption, you need to determine the data requirements for hosting your workloads. Azure database products and services support a wide variety of data storage scenarios and capabilities. How you configure your landing zone environment to support your data requirements depends on your workload governance, technical, and business requirements.

Identify data services requirements

As part of your landing zone evaluation and preparation, you need to identify the data stores that your landing zone needs to support. The process involves assessing each of the applications and services that make up your workloads to determine their data storage and access requirements. After you identify and document these requirements, you can create policies for your landing zone to control allowed resource types based on your workload needs.

For each application or service you'll deploy to your landing zone environment, use the following decision tree as a starting point to help you determine the appropriate data store services to use:

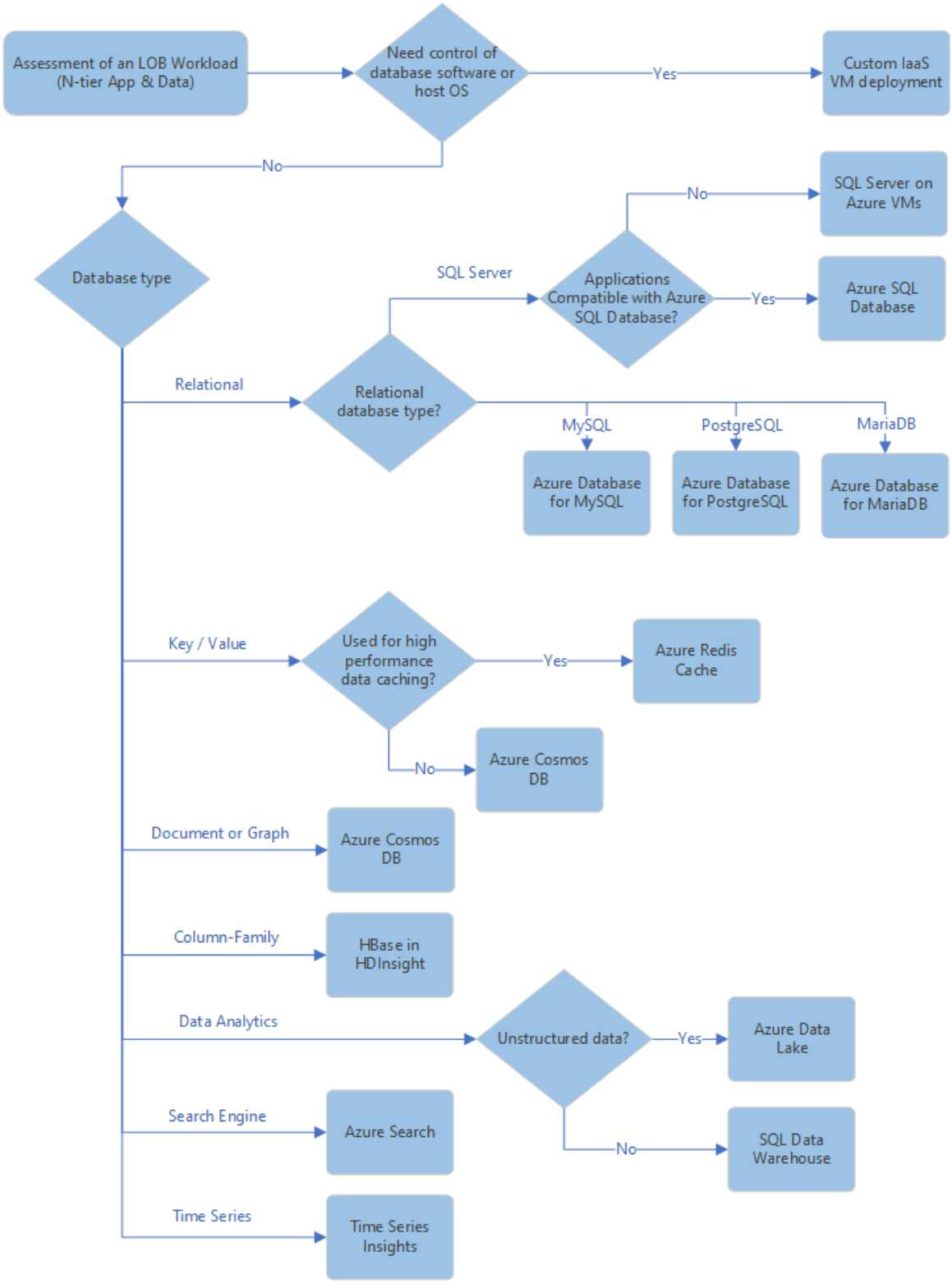


Figure 1: An Azure database services decision tree.

Key questions

Answer the following questions about your workloads to help you make decisions based on the Azure database services decision tree:

- **Do you need full control or ownership of your database software or host OS?** Some scenarios require you to have a high degree of control or ownership of the software configuration and host servers for your database workloads. In these scenarios, you can deploy custom infrastructure as a service (IaaS) virtual machines to fully control the deployment and configuration of data services. If you don't have these requirements, platform as a service (PaaS) database services might reduce your management and operations costs.
- **Will your workloads use a relational database technology?** If so, what technology do you plan to use?

Azure provides managed PaaS database capabilities for Azure SQL Database, MySQL, PostgreSQL, and MariaDB.

- **Will your workloads use SQL Server?** In Azure, you can have your workloads running in IaaS-based SQL Server on Azure Virtual Machines or on the PaaS-based Azure SQL Database hosted service. Choosing which option to use is primarily a question of whether you want to manage your database, apply patches, and take backups, or if you want to delegate these operations to Azure. In some scenarios, compatibility issues might require the use of IaaS-hosted SQL Server. For more information about how to choose the correct option for your workloads, see Choose the right SQL Server option in Azure.
- **Will your workloads use key/value database storage?** Azure Cache for Redis offers a high-performance cached key/value data storage solution that can power fast, scalable applications. Azure Cosmos DB also provides general-purpose key/value storage capabilities.
- **Will your workloads use document or graph data?** Azure Cosmos DB is a multi-model database service that supports a wide variety of data types and APIs. Azure Cosmos DB also provides document and graph database capabilities.
- **Will your workloads use column-family data?** Apache HBase in Azure HDInsight is built on Apache Hadoop. It supports large amounts of unstructured and semi-structured data in a schemaless database that's organized by column families.
- **Will your workloads require high-capacity data analytics capabilities?** You can use Azure SQL Data Warehouse to effectively store and query structured petabyte-scale data. For unstructured big data workloads, you can use Azure data lake to store and analyze petabyte-size files and trillions of objects.
- **Will your workloads require search engine capabilities?** You can use Azure Cognitive Search to build AI-enhanced cloud-based search indexes that can be integrated into your applications.
- **Will your workloads use time series data?** Azure Time Series Insights is built to store, visualize, and query large amounts of time series data, such as data generated by IoT devices.

> **NOTE**
> Learn more about how to assess database options for each of your application or services in the Azure application architecture guide.

Common database scenarios

The following table illustrates a few common use scenario requirements and the recommended database services for handling them:

SCENARIO	DATA SERVICE
I need a globally distributed, multi-model database with support for NoSQL choices.	Azure Cosmos DB
I need a fully managed relational database that provisions quickly, scales on the fly, and includes built-in intelligence and security.	Azure SQL Database
I need a fully managed, scalable MySQL relational database that has high availability and security built in at no extra cost.	Azure Database for MySQL
I need a fully managed, scalable PostgreSQL relational database that has high availability and security built in at no extra cost.	Azure Database for PostgreSQL

SCENARIO	DATA SERVICE
I plan to host enterprise SQL Server applications in the cloud and have full control over the server OS.	SQL Server on virtual machines
I need a fully managed elastic data warehouse that has security at every level of scale at no extra cost.	Azure SQL Data Warehouse
I need Data Lake Storage resources that are capable of supporting Hadoop clusters or HDFS data.	Azure data lake
I need high throughput and consistent, low-latency access for my data to support fast, scalable applications.	Azure Cache for Redis
I need a fully managed, scalable MariaDB relational database that has high availability and security built in at no extra cost.	Azure Database for MariaDB

Regional availability

Azure lets you deliver services at the scale you need to reach your customers and partners, **wherever they are**. A key factor in planning your cloud deployment is to determine what Azure region will host your workload resources.

Most database services are generally available in most Azure regions. But there are a few regions, mostly targeting governmental customers, that support only a subset of these products. Before you decide which regions you will deploy your database resources to, we recommend that you refer to the regions page to check the latest status of regional availability.

To learn more about Azure global infrastructure, see the Azure regions page. You can also view products available by region for specific details about the overall services that are available in each Azure region.

Data residency and compliance requirements

Legal and contractual requirements that are related to data storage often will apply to your workloads. These requirements might vary based on the location of your organization, the jurisdiction of the physical assets that host your data stores, and your applicable business sector. Components of data obligations to consider include data classification, data location, and the respective responsibilities for data protection under the shared responsibility model. For help with understanding these requirements, see the white paper achieving compliant data residency and security with Azure.

Part of your compliance efforts might include controlling where your database resources are physically located. Azure regions are organized into groups called geographies. An Azure geography ensures that data residency, sovereignty, compliance, and resiliency requirements are honored within geographical and political boundaries. If your workloads are subject to data sovereignty or other compliance requirements, you must deploy your storage resources to regions in a compliant Azure geography.

Establish controls for database services

When you prepare your landing zone environment, you can establish controls that limit what data stores users can deploy. Controls can help you manage costs and limit security risks while still allowing developers and IT teams to deploy and configure resources that are needed to support your workloads.

After you identify and document your landing zone's requirements, you can use Azure Policy to control the database resources that you allow users to create. Controls can take the form of allowing or denying the

creation of database resource types. For example, you might restrict users to creating only Azure SQL Database resources. You can also use policy to control the allowable options when a resource is created, like restricting what SQL Database SKUs can be provisioned or allowing only specific versions of SQL Server to be installed on an IaaS VM.

Policies can be scoped to resources, resource groups, subscriptions, and management groups. You can include your policies in Azure blueprint definitions and apply them repeatedly throughout your cloud estate.

Azure role-based access control

5/18/2021 • 5 minutes to read • Edit Online

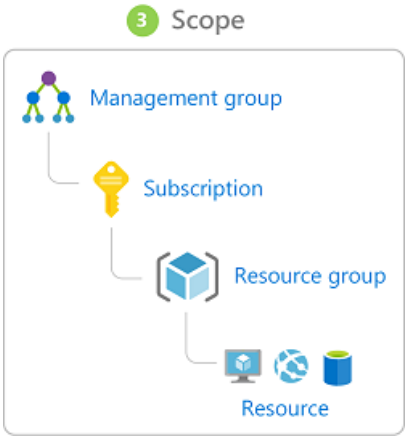
Group-based access rights and privileges are a good practice. Dealing with groups rather than individual users simplifies maintenance of access policies, provides consistent access management across teams, and reduces configuration errors. Assigning users to and removing users from appropriate groups helps keep current the privileges of a specific user. Azure role-based access control (Azure RBAC) offers fine-grained access management for resources organized around user roles.

For an overview of recommended Azure RBAC practices as part of an identity and security strategy, see Azure identity management and access control security best practices.

Overview of Azure role-based access control

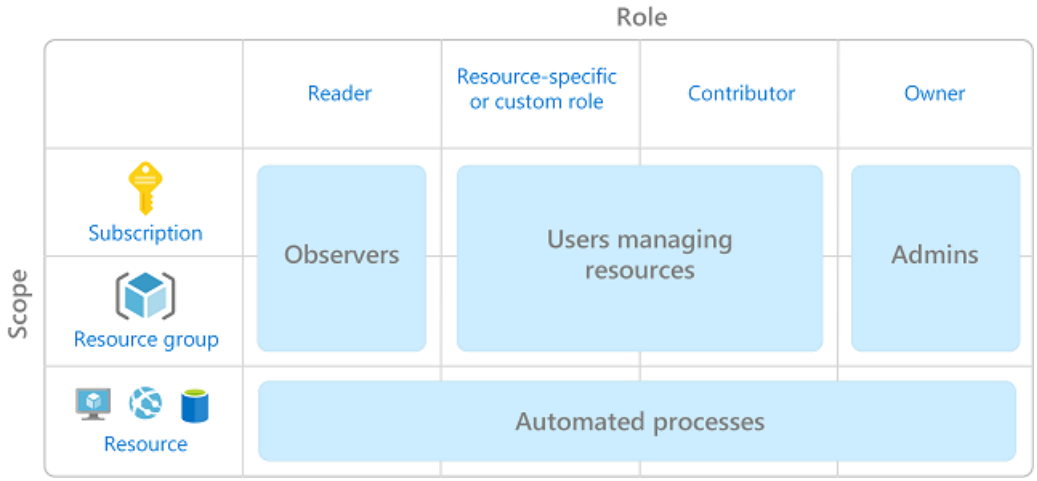
By using Azure role-based access control, you can separate duties within your team and grant only enough access for specific Azure Active Directory (Azure AD) users, groups, service principals, or managed identities to perform their jobs. Instead of giving everybody unrestricted access to your Azure subscription or resources, you can limit permissions for each set of resources.

Azure role definitions list operations that are permitted or disallowed for users or groups assigned to that role. A role's scope specifies which resources these defined permissions apply to. Scopes can be specified at multiple levels: management group, subscription, resource group, or resource. Scopes are structured in a parent/child relationship.



For detailed instructions for assigning users and groups to specific roles and assigning roles to scopes, see Add or remove Azure role assignments using the Azure portal.

When planning your access control strategy, use a least-privilege access model that grants users only the permissions required to perform their work. The following diagram shows a suggested pattern for using Azure RBAC through this approach.



NOTE

The more specific or detailed permissions are that you define, the more likely it is that your access controls will become complex and difficult to manage. This is especially true as your cloud estate grows in size. Avoid resource-specific permissions. Instead, use management groups for enterprise-wide access control and resource groups for access control within subscriptions. Also avoid user-specific permissions. Instead, assign access to groups in Azure AD.

Use Azure built-in roles

Azure provides a many built-in role definitions, with three core roles for providing access:

- The Owner role can manage everything, including access to resources.
- The Contributor role can manage everything except access to resources.
- The Reader role can view everything but not make any changes.

Beginning from these core access levels, additional built-in roles provide more detailed controls for accessing specific resource types or Azure features. For example, you can manage access to virtual machines by using the following built-in roles:

- The Virtual Machine Administrator Login role can view virtual machines in the portal and sign in as `administrator`.
- The Virtual Machine Contributor role can manage virtual machines, but it can't access them or the virtual network or storage account they're connected to.
- The Virtual Machine User Login role can view virtual machines in the portal and sign in as a regular user.

For another example of using built-in roles to manage access to particular features, see the discussion on controlling access to cost-tracking features in Track costs across business units, environments, or projects.

For a complete list of available built-in roles, see Azure built-in roles.

Use custom roles

Although the roles built in to Azure support a wide variety of access control scenarios, they might not meet all the needs of your organization or team. For example, if you have a single group of users responsible for managing virtual machines and Azure SQL Database resources, you might want to create a custom role to optimize management of the required access controls.

The Azure RBAC documentation contains instructions on creating custom roles, along with details on how role definitions work.

Separation of responsibilities and roles for large organizations

Azure RBAC allows organizations to assign different teams to various management tasks within large cloud estates. It can allow central IT teams to control core access and security features, while also giving software developers and other teams large amounts of control over specific workloads or groups of resources.

Most cloud environments can also benefit from an access-control strategy that uses multiple roles and emphasizes a separation of responsibilities between these roles. This approach requires that any significant change to resources or infrastructure involves multiple roles to complete, ensuring that more than one person must review and approve a change. This separation of responsibilities limits the ability of a single person to access sensitive data or introduce vulnerabilities without the knowledge of other team members.

The following table illustrates a common pattern for dividing IT responsibilities into separate custom roles:

GROUP	COMMON ROLE NAME	RESPONSIBILITIES
Security operations	SecOps	Provides general security oversight. Establishes and enforces security policy such as encryption at rest. Manages encryption keys. Manages firewall rules.
Network operations	NetOps	Manages network configuration and operations within virtual networks, such as routes and peerings.
Systems operations	SysOps	Specifies compute and storage infrastructure options, and maintains resources that have been deployed.
Development, test, and operations	DevOps	Builds and deploys workload features and applications. Operates features and applications to meet service-level agreements and other quality standards.

The breakdown of actions and permissions in these standard roles are often the same across your applications, subscriptions, or entire cloud estate, even if these roles are performed by different people at different levels. Accordingly, you can create a common set of Azure role definitions to apply across different scopes within your environment. Users and groups can then be assigned a common role, but only for the scope of resources, resource groups, subscriptions, or management groups that they're responsible for managing.

For example, in a hub and spoke network topology with multiple subscriptions, you might have a common set of role definitions for the hub and all workload spokes. A hub subscription's NetOps role can be assigned to members of the organization's central IT team, who are responsible for maintaining networking for shared services used by all workloads. A workload spoke subscription's NetOps role can then be assigned to members of that specific workload team, allowing them to configure networking within that subscription to best support their workload requirements. The same role definition is used for both, but scope-based assignments ensure that users have only the access that they need to perform their job.

Create hybrid cloud consistency

5/18/2021 • 6 minutes to read • Edit Online

This article guides you through the high-level approaches for creating hybrid cloud consistency.

Hybrid deployment models during migration can reduce risk and contribute to a smooth infrastructure transition. Cloud platforms offer the greatest level of flexibility when it comes to business processes. Many organizations are hesitant to make the move to the cloud. Instead, they prefer to keep full control over their most sensitive data. Unfortunately, on-premises servers don't allow for the same rate of innovation as the cloud. A hybrid cloud solution offers the speed of cloud innovation and the control of on-premises management.

Integrate hybrid cloud consistency

Using a hybrid cloud solution allows organizations to scale computing resources. It also eliminates the need to make massive capital expenditures to handle short-term spikes in demand. Changes to your business can drive the need to free up local resources for more sensitive data or applications. It's easier, faster, and less expensive to deprovision cloud resources. You pay only for those resources your organization temporarily uses, instead of having to purchase and maintain additional resources. This approach reduces the amount of equipment that might remain idle over long periods of time. Hybrid cloud computing delivers all the benefits of cloud computing flexibility, scalability, and cost efficiencies with the lowest possible risk of data exposure.

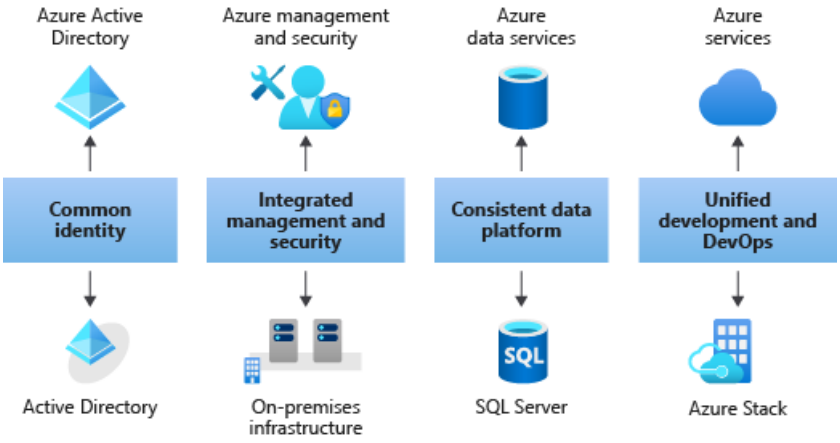


Figure 1: Creating hybrid cloud consistency across identity, management, security, data, development, and DevOps.

A true hybrid cloud solution must provide four components, each of which brings significant benefits:

- **Common identity for on-premises and cloud applications:** This component improves user productivity by giving users single sign-on (SSO) to all their applications. It also ensures consistency as applications and users cross network or cloud boundaries.
- **Integrated management and security across your hybrid cloud:** This component provides you with a cohesive way to monitor, manage, and secure the environment, which enables increased visibility and control.
- **A consistent data platform for the datacenter and the cloud:** This component creates data portability, combined with seamless access to on-premises and cloud data services for deep insight into all data sources.
- **Unified development and DevOps across the cloud and on-premises datacenters:** This component allows you to move applications between the two environments as needed. Developer productivity improves because both locations now have the same development environment.

Here are some examples of these components from an Azure perspective:

- Azure Active Directory (Azure AD) works with on-premises Active Directory to provide common identity for all users. SSO across on-premises and via the cloud makes it simple for users to safely access the applications and assets they need. Admins can manage security and governance controls and also have the flexibility to adjust permissions without affecting the user experience.
- Azure provides integrated management and security services for both cloud and on-premises infrastructure. These services include an integrated set of tools that are used to monitor, configure, and protect hybrid clouds. This end-to-end approach to management specifically addresses real-world challenges that face organizations considering a hybrid cloud solution.
- Azure hybrid cloud provides common tools that ensure secure access to all data, seamlessly and efficiently. Azure data services combine with Microsoft SQL Server to create a consistent data platform. A consistent hybrid cloud model allows users to work with both operational and analytical data. The same services are provided on-premises and in the cloud for data warehousing, data analysis, and data visualization.
- Azure Cloud Services, combined with Azure Stack on-premises, provide unified development and DevOps. Consistency across the cloud and on-premises means that your DevOps team can build applications that run in either environment and can easily deploy to the right location. You also can reuse templates across the hybrid solution, which can further simplify DevOps processes.

Azure Stack in a hybrid cloud environment

Azure Stack is a hybrid cloud solution that allows organizations to run Azure-consistent services in their datacenter. It provides a simplified development, management, and security experience that's consistent with Azure public cloud services. Azure Stack is an extension of Azure. You can use it to run Azure services from your on-premises environments and then move to the Azure cloud if and when required.

With Azure Stack, you can deploy and operate both IaaS and PaaS by using the same tools and offering the same experience as the Azure public cloud. Management of Azure Stack, whether through the web UI portal or through PowerShell, has a consistent look and feel for IT administrators and end users with Azure.

Azure and Azure Stack open up new hybrid use cases for both customer-facing and internal line-of-business applications:

- **Edge and disconnected solutions.** To address latency and connectivity requirements, customers can process data locally in Azure Stack and then aggregate it in Azure for further analytics. They can use common application logic across both. Many customers are interested in this edge scenario across different contexts, like factory floors, cruise ships, and mine shafts.
- **Cloud applications that meet various regulations.** Customers can develop and deploy applications in Azure, with full flexibility to deploy on-premises on Azure Stack to meet regulatory or policy requirements. No code changes are needed. Application examples include global audit, financial reporting, foreign exchange trading, online gaming, and expense reporting. Customers sometimes look to deploy different instances of the same application to Azure or Azure Stack, based on business and technical requirements. While Azure meets most requirements, Azure Stack complements the deployment approach where needed.
- **Cloud application model on-premises.** Customers can use Azure web services, containers, microservices, and serverless architectures to update and extend existing applications or build new ones. You can use consistent DevOps processes across Azure in the cloud and Azure Stack on-premises. There's a growing interest in application modernization, even for core mission-critical applications.

Azure Stack is offered via two deployment options:

- **Azure Stack integrated systems:** Azure Stack integrated systems are offered through Microsoft and hardware partners to create a solution that provides cloud-paced innovation balanced with simple management. Because Azure Stack is offered as an integrated system of hardware and software, you get flexibility and control while still adopting innovation from the cloud. Azure Stack integrated systems range in size from 4 to 12 nodes. They're jointly supported by the hardware partner and Microsoft. Use Azure Stack integrated systems to enable new scenarios for your production workloads.

- **Azure Stack Development Kit:** The Microsoft Azure Stack Development Kit is a single-node deployment of Azure Stack. You can use it to evaluate and learn about Azure Stack. You can also use the kit as a developer environment, where you can develop by using APIs and tooling that are consistent with Azure. The Azure Stack Development Kit isn't intended for use as a production environment.

Azure Stack one-cloud ecosystem

You can speed up Azure Stack initiatives by using the complete Azure ecosystem:

- Azure ensures that most applications and services that are certified for Azure will work on Azure Stack. Several ISVs are extending their solutions to Azure Stack. These ISVs include Bitnami, Docker, Kemp Technologies, Pivotal Cloud Foundry, Red Hat Enterprise Linux, and SUSE Linux.
- You can opt to have Azure Stack delivered and operated as a fully managed service. Several partners will have managed service offerings across Azure and Azure Stack shortly. These partners include Tieto, Yourhosting, Revera, Pulsant, and NTT. These partners deliver managed services for Azure via the Cloud Solution Provider (CSP) program. They're extending their offerings to include hybrid solutions.
- As an example of a complete, fully managed hybrid cloud solution, Avanade delivers an all-in-one offer. It includes cloud transformation services, software, infrastructure, setup and configuration, and ongoing managed services. This way customers can consume Azure Stack just as they do with Azure today.
- Providers can help accelerate application modernization initiatives by building end-to-end Azure solutions for customers. Each provider brings a deep Azure skill set, domain and industry knowledge, and process expertise such as DevOps. Every Azure Stack implementation is an opportunity for a provider to design the solution and lead and influence system deployment. They can also customize the included capabilities and deliver operational activities. Examples of providers include Avanade, DXC, Dell EMC Services, Infront Consulting Group, HPE Pointnext, and PWC (formerly PricewaterhouseCoopers).

Improve landing zone operations

5/18/2021 • 2 minutes to read • Edit Online

Landing zone operations provide the initial foundation for operations management. As operations scale, these improvements will refactor landing zones to meet growing operational excellence, reliability, and performance requirements.

Landing zone operations best practices

The following links provide best practices for improving landing zone operations.

- Azure server management: An onboarding guide to incorporating the cloud-native tools and services needed to manage operations.
- Hybrid monitoring: Many customers have already made a substantial investment in System Center Operations Manager. For those customers, this guide to hybrid monitoring can help them compare and contrast the cloud-native reporting tools with Operations Manager tooling. This comparison makes it easier to decide which tools to use for operational management.
- Centralize management operations: Use Azure Lighthouse to centralize operations management across multiple Azure tenants.
- Establish an operational fitness review: Review an environment for operational fitness.
- Workload specific operations best practices:
  - Resiliency checklist
  - Failure mode analysis
  - Recover from a region wide service disruption
  - Recover from data corruption or accidental deletion

Four steps to improve operations beyond a single landing zone

The Manage methodology provides overall guidance for building out operations management capacity, see the Manage methodology. We will use the basic structure of that methodology and the following steps from that methodology to improve landing zone operations and operations across all landing zones.

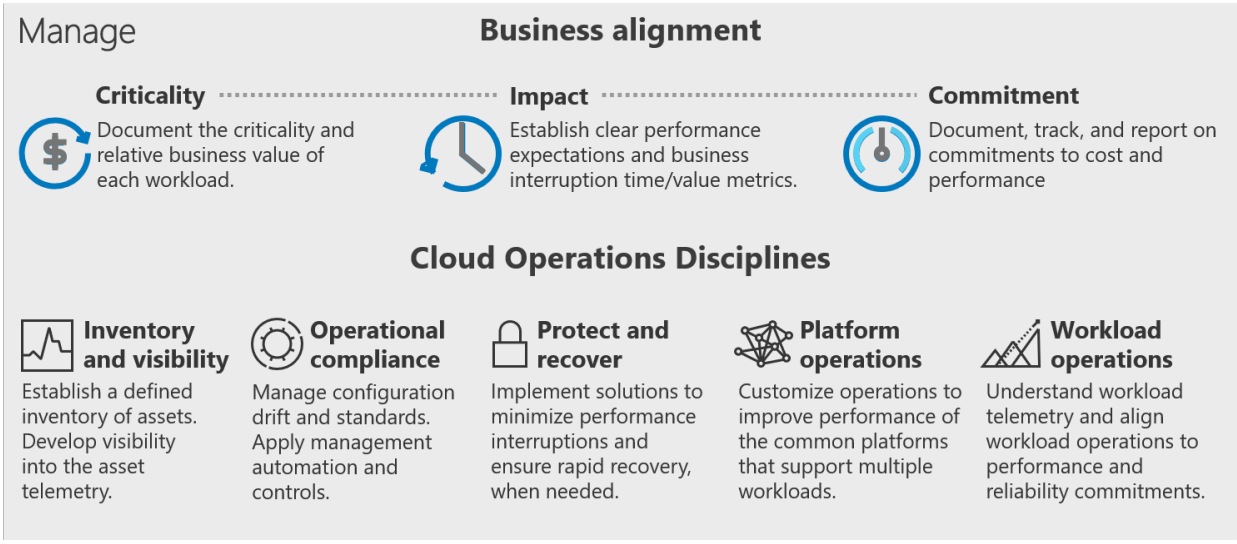


Figure 1: The Manage methodology of the Cloud Adoption Framework.

1. Establish a management baseline: A management baseline establishes the foundation for operations

management. The guidance under this first step can be applied to any landing zone to improve initial operations.

2. Define business commitments: Understanding the criticality and impact of each workload within a landing zone will establish a "definition of done" for any ongoing management improvements for any landing zone. This process will also identify the reliability, performance, and operations requirements of each workload.
3. Expand the management baseline: This set of best practices can be applied to improve landing zone operations beyond the initial baseline.
4. Advanced operations and design principles: Review the design and operations of specific workloads, platforms, or full landing zones to meet deeper requirements.

Test-driven development cycle

Before beginning any security improvements, it's important to understand the "definition of done" and all "acceptance criteria". For more information, see the articles on test-driven development of landing zones and test-driven development in Azure.

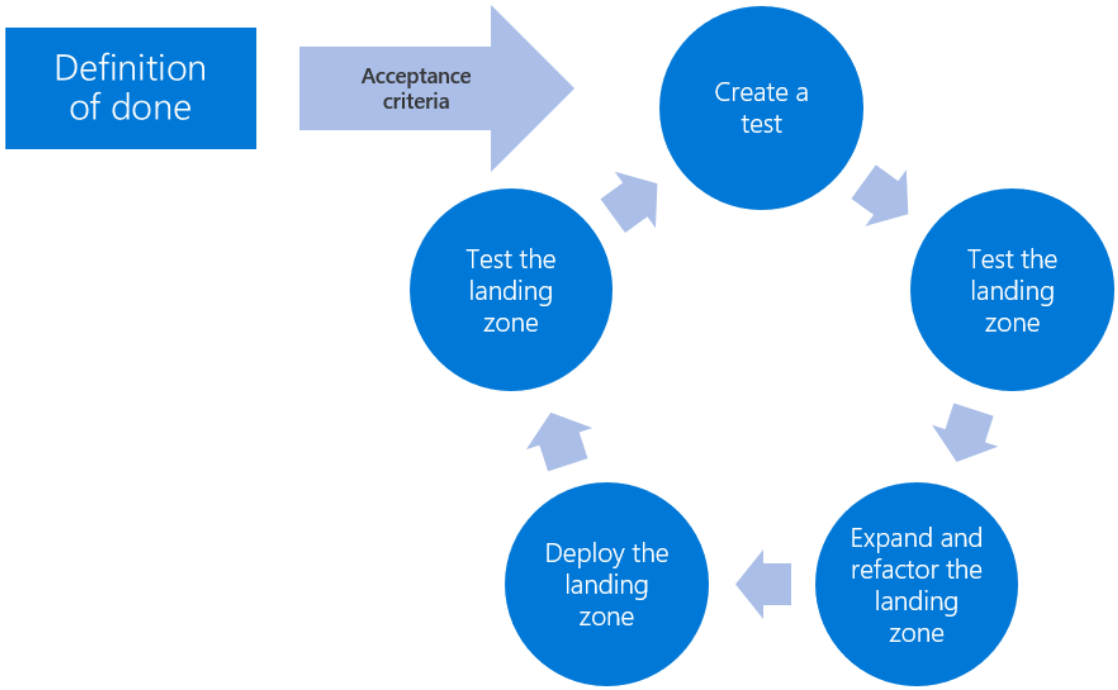


Figure 2: The test-driven development process for cloud landing zones.

Next steps

Understand how to improve landing zone governance to support adoption at scale.

Improve landing zone governance

Improve landing zone governance

5/18/2021 • 2 minutes to read • Edit Online

Landing zone governance is the smallest unit of overall governance. Establishing a sound governance foundation within your first few landing zones will reduce the amount of refactoring required later in the adoption lifecycle. Improving landing zone governance will integrate cost controls, establish basic tooling to allow for scale, and will make it easier for the cloud governance team to deliver on the Five Disciplines of Cloud Governance.

Landing zone governance best practices

- **Initial landing zone governance:** The article on establishing an initial governance foundation can assist with adding initial governance tooling to the first few landing zones. These practices will aid in scaling adoption and governance, along with implementing sound cost management. This approach starts with: resource organization, policy definitions, Azure roles, and blueprint definitions.
- Naming and tagging standards: Ensure consistency in naming and tagging, which is the foundational data for establishing sound governance practices.
- Track costs across workloads: Begin tracking costs in your first landing zone. Evaluate how you will apply consistency across multiple workloads and roles.
- Scale with multiple subscriptions: Evaluate how this landing zone and other landing zones will scale, as multiple subscriptions become a requirement.
- Organize subscriptions: Understand how to organize and manage multiple subscriptions.

Four steps to improve overall governance

The Govern methodology provides overall guidance for building out governance policies, processes, and disciplines. We will use the basic structure of that methodology and the following steps from that methodology to improve landing zone governance and governance across all landing zones.

1. Understand the methodology: Understand the basic methodology to guide end-state governance design.
2. Benchmark: Assess both current and future state to establish a vision and take action.
3. Initial governance foundation: Small, easily implemented set of governance tools to establish an initial foundation for all landing zones.
4. Improve the governance foundation: Iteratively add governance controls to strengthen all landing zone governance.

Next steps

Cloud adoption will continue to expand with each wave or release of new workloads. To stay ahead of these requirements, cloud platform teams should periodically review additional landing zone best practices.

Review additional landing zone best practices

Improve landing zone security

3/6/2021 • 2 minutes to read • Edit Online

When a workload or the landing zones that hosts it require access to any sensitive data or critical systems, it's important to protect the data and assets. Improving landing zone security builds on the test-driven development approach to landing zones by expanding or refactoring the landing zone to account for heightened security requirements.

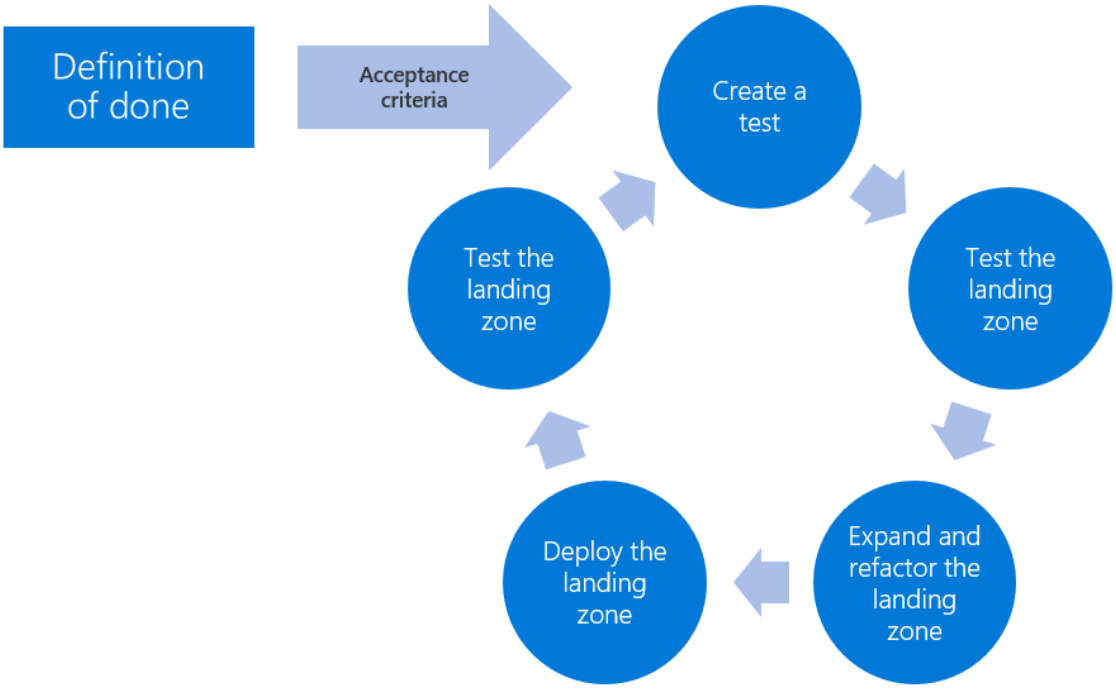
Landing zone security best practices

The following list of reference architectures and best practices provides examples of ways to improve landing zone security:

- Azure Security Center: Onboard a subscription to Security Center.
- Azure Sentinel: Onboard Azure Sentinel to provide a **security information event management (SIEM)** and **security orchestration automated response (SOAR)** solution.
- Network boundary security: Several reference patterns for developing a network, similar to how the network boundary is secured in a datacenter.
- Secure network architecture: Reference architecture for implementing a perimeter network and secure network architecture.
- Identity management and access control: Series of best practices for implementing identity and access to secure a landing zone in Azure.
- Network security practices: Provides additional best practices for securing the network.
- Operational security provides best practices for increasing operational security in Azure.
- The Security Baseline discipline: Example of developing a governance-driven security baseline to enforce security requirements.

Test-driven development cycle

Before beginning any security improvements, it's important to understand the "definition of done" and all "acceptance criteria". For more information, see the articles on test-driven development of landing zones and test-driven development in Azure.



Next steps

Understand how to improve landing zone operations to support critical applications.

Improve landing zone operations

Evaluate a Microsoft partner's Azure landing zone

3/6/2021 • 8 minutes to read • Edit Online

The Cloud Adoption Framework approaches cloud adoption as a self-service activity. The objective is to empower each team that supports adoption through standardized approaches. In practice, you can't assume that a self-service approach is sufficient for all adoption activities.

Successful cloud adoption programs typically involve at least one level of third-party support. Many cloud adoption efforts require support from a systems integrator (SI) or consulting partner who provides services that accelerate cloud adoption. Managed service providers (MSPs) provide enduring value by supporting landing zones and cloud adoption, but they also provide post-adoption operations management support. Additionally, successful cloud adoption efforts tend to engage one or more independent software vendors (ISV) who provide software-based services that accelerate cloud adoption. The rich partner ecosystems of SIs, ISVs, MSPs, and other forms of Microsoft partners have aligned their offerings to specific methodologies found in the Cloud Adoption Framework. When a partner is aligned to the Ready methodology of this framework, they will likely offer their own Azure landing zone implementation option.

This article provides a set of questions that help create an understanding of the scope of the partner's Azure landing zone implementation options.

> **IMPORTANT**
>
> Partner offers and Azure landing zone implementation options are defined by the partner, based on their extensive experience helping customers adopt the cloud.
>
> Partners might choose to omit the implementation of specific design areas in their initial landing zone implementation. However, they should be able to communicate when and how each design area is implemented, as well as a range of costs for completing that design area whenever possible.
>
> Other partner solutions might be flexible enough to support multiple options for each of the questions below. Use these questions to ensure you're comparing partner offers and self-service options equally.

Find a partner

If you need a partner to implement your Azure landing zones, start with the approved list of Cloud Adoption Framework aligned partners. Specifically, start with partners who have offers aligned to the Ready methodology.

Additionally, all Azure expert managed service providers (MSPs) have been audited to validate their ability to deliver each methodology of the Cloud Adoption Framework. While a particular partner might not have an aligned offer, all partners have demonstrated alignment during technical delivery.

Validate a partner offer

Once a partner is selected, use the remainder of this article to guide your validation of the partner offer. Each section includes a summary of what to look for and a list of questions to ask the partner. The partner's answers to these questions shouldn't be considered as right or wrong. Instead, the questions are designed to help you evaluate whether the partner offer will meet your business requirements.

Platform development velocity

As outlined in the Azure landing zone implementation options, there are two high-level approaches to landing zone implementation based on how you want to develop your landing zones.

Question for the partner: Which of the following approaches are supported by the partner's Azure landing zone solution?

- **Start small and expand:** Begin with a lightweight template. The landing zone solution is matured over time as your desired cloud operating model becomes clearer.
- **Start with enterprise-scale:** Begin with a more comprehensive reference implementation. The reference architecture builds on a well-defined cloud operating model that requires less iteration to reach a mature solution.
- **Other:** The partner has a modified approach and should be able to describe the approach.

Design principles

All Azure landing zones must consider the following set of common design areas. We refer to the way those design areas are implemented as design principles. The following sections will help validate the partner's design principles that define the Azure landing zone implementation.

Deployment options

Partners who offer an Azure landing zone solution might support one or more options to deploy (or modify/expand the landing zone) the solution to your Azure tenant.

Question for the partner: Which of the following does your Azure landing zone solution support?

- **Configuration automation:** Does the solution deploy the landing zone from a deployment pipeline or deployment tool?
- **Manual configuration:** Does the solution empower the IT team to manually configure the landing zone, without injecting errors into the landing zone source code?

Question for the partner: Which of the Azure landing zone implementation options are supported by the partner's solution? See the Azure landing zone implementation options article for a full list of options.

Identity

Identity is perhaps the most important design area to evaluate in the partner solution.

Question for the partner: Which of the following identity management options does the partner solution support?

- **Azure AD:** The suggested best practice is to use Azure AD and Azure role-based access control to manage identity and access in Azure.
- **Active Directory:** If required, does the partner solution provide an option to deploy Active Directory as an infrastructure as a service solution?
- **Third-party identity provider:** If your company uses a third-party identity solution, determine whether and how the partner's Azure landing zone integrates with the third-party solution.

Network topology and connectivity

Networking is arguably the second most important design area to evaluate. There are several best practice approaches to network topology and connectivity.

Question for the partner: Which of the following options is included with the partner's Azure landing zone solution? Are any of the following options incompatible with the partner's solution?

- **Virtual network:** Does the partner solution configure a virtual network? Can its topology be modified to meet your technical or business constraints?
- **Virtual private network (VPN):** Is VPN configuration included in the partner's landing zone design to

connect the cloud to existing datacenters or offices?

- **High-speed connectivity:** Is a high-speed connection such as Azure ExpressRoute included in the landing zone design?
- **Virtual network peering:** Does the design include connectivity between different subscriptions or virtual networks in Azure?

Resource organization

Sound governance and operational management of the cloud starts with best practice resource organization.

Question for the partner: Does the partner's landing zone design include considerations for the following resource organization practices?

- **Naming standards:** What naming standards will this offering follow and is that standard automatically enforced through policy?
- **Tagging standards:** Does the landing zone configuration follow and enforce specific standards for tagging assets?
- **Subscription design:** What subscription design strategies are supported by the partner offer?
- **Management group design:** Does the partner offer follow a defined pattern for the Azure management group hierarchy to organize subscriptions?
- **Resource group alignment:** How are resource groups used to group assets deployed to the cloud? In the partner offer, are resource groups used to group assets into workloads, deployment packages, or other organization standards?

Question for the partner: Does the partner provide onboarding documentation to track foundational decisions and educate staff? See the initial decision template for an example of such documentation.

Governance disciplines

Your governance requirements can heavily influence any complex landing zone designs. Many partners provide a separate offering to fully implement governance disciplines after landing zones are deployed. The following questions will help create clarity around the aspects of governance that will be built into any landing zones.

Question for the partner: What governance tooling does the partner solution include as part of the landing zone implementation?

- **Policy compliance monitoring:** Does the partner's landing zone solution includes defined governance policies along with tools and processes to monitor compliance? Does the offer include customization of policies to fit your governance needs?
- **Policy enforcement:** Does the partner's landing zone solution include automated enforcement tools and processes?
- **Cloud platform governance:** Does the partner offer include a solution for maintaining compliance to a common set of policies across all subscriptions? Or is the scope limited to individual subscriptions?
- **N/A:** Start-small approaches intentionally postpone governance decisions until the team has deployed low-risk workloads to Azure. This can be addressed in a separate offer after the landing zone solution has been deployed.

Question for the partner: Does the partner offer go beyond governance tooling to also include processes and practices for delivering any of the following cloud governance disciplines?

- **Cost management:** Does the partner offer prepare the team to evaluate, monitor, and optimize spend while creating cost accountability with workload teams?
- **Security baseline:** Does the partner offer prepare the team to maintain compliance as security requirements change and mature?
- **Resource consistency:** Does the partner offer prepare the team to ensure that all assets in the cloud are onboarded into relevant operations management processes?

- **Identity baseline:** Does the partner offer prepare the team to maintain identity, role definitions, and assignments after the initial landing zone is deployed?

Operations baseline

Your operations management requirements could influence configuration of specific Azure products during landing zone implementation. Many partners provide a separate offering to fully implement the operations baseline and advanced operations later in the cloud adoption journey, but before your first workload is released for production use. But, the partner's landing zone solution might include configuration for a number of operations management tools by default.

Question for the partner: Does the partner solution include design options to support any of the cloud operations disciplines?

- **Inventory and visibility:** Does the landing zone include tooling to ensure that 100% of assets are centrally monitored?
- **Operational compliance:** Does the architecture include tooling and automated processes to enforce patching or other operational compliance requirements?
- **Protect and recover:** Does the partner offer include tooling and configuration to ensure a minimal standard of backup and recovery for 100% of assets deployed?
- **Platform operations:** Does the landing zone offering include tooling or processes to optimize operations across the portfolio?
- **Workload operations:** Does the landing zone offering include tooling to manage workload-specific operations requirements and ensure that each workload is well-architected?

Take action

After reviewing the partner's Azure landing zone offer or solution using the questions above, your team will be better equipped to choose the partner whose Azure landing zone most closely aligns to your cloud operating model.

If you determine that a self-service approach to Azure landing zone deployment is a better fit, review or revisit the Azure landing zone implementation options to find the templated landing zone approach that best aligns with your cloud operating model.

Next steps

Learn about the process for refactoring landing zones.

Refactor landing zones

Refactor landing zones

5/26/2021 • 7 minutes to read • Edit Online

A landing zone is an environment for hosting your workloads that's **preprovisioned through code**. Since landing zone infrastructure is defined in code, it can be refactored similar to any other codebase. Refactoring is the process of modifying or restructuring source code to optimize the output of that code without changing its purpose or core function.

The Ready methodology uses the concept of refactoring to accelerate migration and remove common blockers. The steps in the ready overview discuss a process that starts with predefined landing zone template that aligns best with your hosting function. Then refactor or add to the source code to expand the landing zones ability to deliver that function through improved security, operations, or governance. The following image illustrates the concept of refactoring.

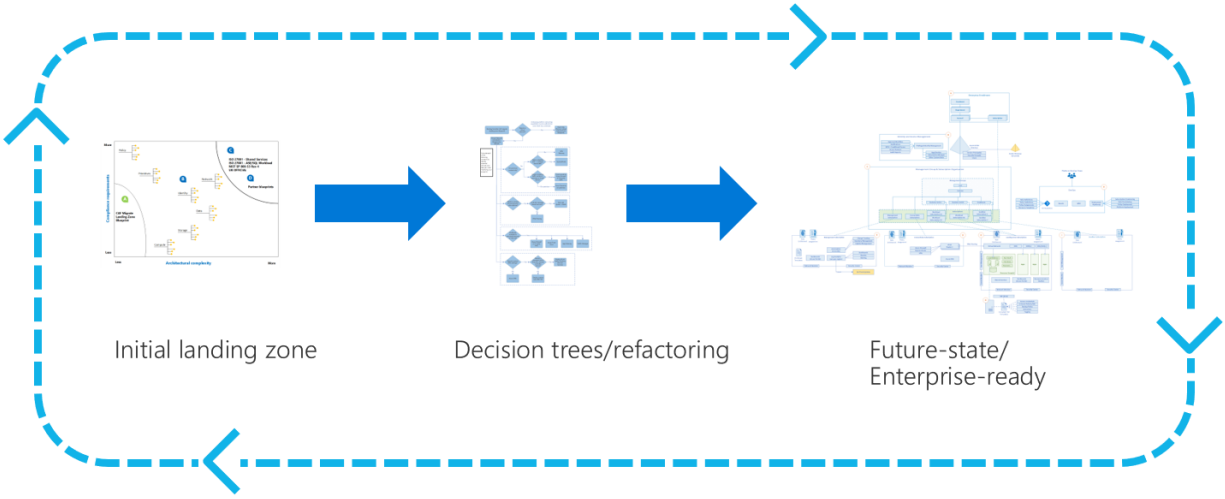


Figure 1: Landing zone refactoring.

Common blockers

When customers adopt the cloud, landing zone considerations are the single most common blocker to adoption and cloud-related business results. Customers tend to lean towards one of the following two blockers. Various teams often lean towards one of these two blockers, resulting in cultural deadlocks that make adoption difficult.

Both of the primary blockers are rooted in one belief, the cloud environment and the existing datacenters should be at or near feature parity regarding operations, governance, and security. This is a wise long-term goal. But the pain comes from the delicate balance between the timing to achieve that goal and the speed required to deliver business results.

Blocker: Acting too soon

It took years and significant effort to reach the current state of security governance and operations in the current datacenter. It also required observations, learning, and customization to meet the unique constraints of that environment. Replicating those same procedures and configurations will take time. Reaching complete feature parity may also result in an environment that underperforms in the cloud. This parity approach also commonly leads to significant unplanned overspending in the cloud environment. Don't try to apply current-state requirements to a future-state environment as an early stage gate. Such an approach rarely proves to be profitable.



Figure 2: Acting too soon is a common blocker.

In the image above, the customer has an objective of 100 workloads running in the cloud. To get there, the customer will likely deploy their first workload and then their first ten or so workloads before they're ready to release one of them to production. Eventually, they'll reach the objective of the adoption plan and have a robust portfolio in the cloud. But the red *X* in the image shows where customers commonly get stuck. Waiting for total alignment can delay the first workload by weeks, months, or even years.

Blocker: Acting too late

On the other hand, acting too late can have significant long-term consequences on the success of the cloud adoption effort. If the team waits to reach feature parity until the adoption efforts are complete, they will encounter unnecessary roadblocks and require several escalations to keep the efforts on track.



Figure 3: Acting too late is a common blocker.

Similar to acting too soon, in this image, the customer waits too long to reach enterprise readiness across landing zones. By waiting too long, the customer will be constrained on the amount of refactoring and expansion they can do in the environment. Those constraints will limit their ability to drive continued success.

Finding balance

To avoid these common blockers, we suggest an iterative approach based on well-structured cloud adoption plan, which maximizes learning opportunities and minimizes time to business success. Refactoring and parallel efforts are critical to this approach.

> **WARNING**
> Adoption teams who have a mid-term objective (within 24 months) to **host more than 1,000 assets (applications, infrastructure, or data assets) in the cloud** are highly unlikely to be successful using a refactoring approach. The learning curve is too high and the timeline too tight to allow for organic approaches to skills attainment. A more complete starting point requiring less customization will be a better path to achieve your objectives. Your implementation partners will likely be able to guide you through a better approach.

The remainder of this article will focus on some key constraints that can empower a refactoring approach, while minimizing risk.

Theory

The concept of refactoring a landing zone is simple, but execution requires proper guardrails. The concept shown above outlines the basic flow:

- When you're ready to build your first landing zone, start with an initial landing zone defined via a template.
- Once that landing zone is deployed, use the decision trees in the subsequent articles under the

`Expand your landing zone` section of the table of contents to refactor and add to your initial landing zone.

- Repeat decision trees and refactoring until you have an enterprise-ready environment that meets the enhanced requirements of your security, operations, and governance teams.

Development approach

The advantage of a refactoring-based approach, is the ability to create parallel iteration paths for development. The image below provides an example of two parallel iteration paths: cloud adoption and cloud platform. Both progress at their own pace, with minimal risk of becoming a blocker to either team's daily efforts. Alignment on the adoption plan and refactoring guardrails can lead to agreement about milestones and clarity about future-state dependencies.

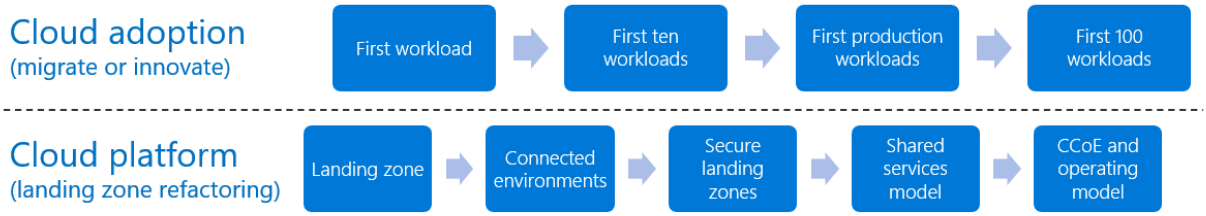


Figure 4: Landing zone parallel iteration.

In the example iteration paths above, the cloud adoption team is migrating their portfolio of 100 workloads to the cloud. In parallel, the cloud platform team is focused on staying ahead of the cloud adoption plan to ensure the environment is prepared for those workloads.

In this example, the planned iterations run as follows:

- The cloud platform team starts the development efforts by deploying an initial landing zone. That landing zone allows the cloud adoption team to deploy and begin testing their first workload.
- To prepare for the cloud adoption team's next deployment of 10 workloads, the cloud platform team works ahead to refactor and add a connected environment, treating the cloud as a perimeter network.
- Before the adoption team can release their first production workload, the security team requires a security review. While the adoption team deploys their first 10 workloads, the platform team moves ahead to define and implement security requirements.
- By the time the first workload is released to production, both teams should have enough learnings to prepare for a longer term shared service model. Centralizing core service architectures will help align governance and operations team. Centralizing core services will help prepare the adoption team to scale and release the next several waves of production workloads.
- As the team approaches their goal of migrating 100 workloads, the team will natural begin to move towards more of a cloud center of excellence collaboration model and team structure.

Configuring an enterprise-ready environment will take time. This approach will not eliminate that requirement. Instead, this approach is designed to remove early blockers and create opportunities for the platform and adoption teams to learn together.

Landing zone refactoring guardrails

All initial landing zone templates have limitations. Guardrails or policies during refactoring should reflect those limitations. Before beginning a landing zone refactoring process, it is important to understand the long-term requirements of the cloud adoption plan and classification of the candidate workloads, compared to the initial template limitations.

As an example of establishing refactoring guardrails, lets compare the development approach in the prior example and the CAF Migration landing zone blueprint.

- Per the assumptions of the CAF Migration landing zone blueprint, this initial landing zone is not designed for sensitive data or mission-critical workloads. Those features will have to be added through refactoring.
- In this example, lets assume that the portfolio of 100 workloads will require both mission critical and sensitive data hosting capabilities.

To balance these two competing requirements, the adoption team and platform team will agree to and operate under the following conditions:

- The cloud adoption team will prioritize production workloads that don't have access to sensitive data and are not deemed mission critical.
- Prior to production release, the security and operations team will validate alignment to the prior policy.
- The cloud platform team will work with the security and governance teams to implement a security baseline. Once security approves the implementation, the adoption team will be cleared to migrate workloads that have access to some sensitive data.
- The cloud platform team will work with the operations team to implement a management baseline. Once the operations team approves the implementation, the adoption team will be cleared to migrate workloads with a higher level of criticality.

For this example, the above set of agreed upon conditions will allow the adoption team get started on their migration effort. It also helps the platform team shape their interactions with other teams, as they build towards a longer-term enterprise ready environment.

Meeting long-term requirements while refactoring

The section of the Ready methodology on expanding your landing zone will aid in moving towards the longer term requirements. As the cloud adoption team progresses with their adoption plan, review Expand your landing zone for guidance to help make decisions and refactor to meet the evolving requirements of various teams.

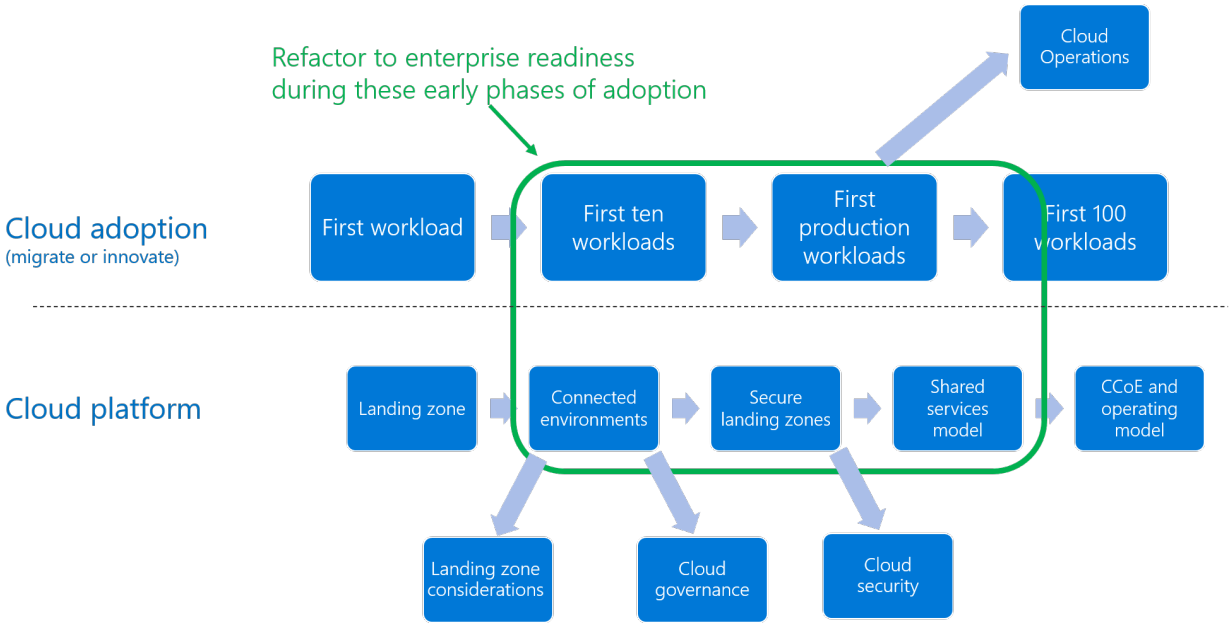


Figure 5: Deeper methodologies assisting a parallel landing zone iteration.

Each subsection of expand your landing zone maps to one of the additions outlined in the image above. Beyond those basic expansions, the deeper methodologies (such as govern or manage) of this framework will aid in going beyond basic landing zone modifications to implement long-term disciplines.

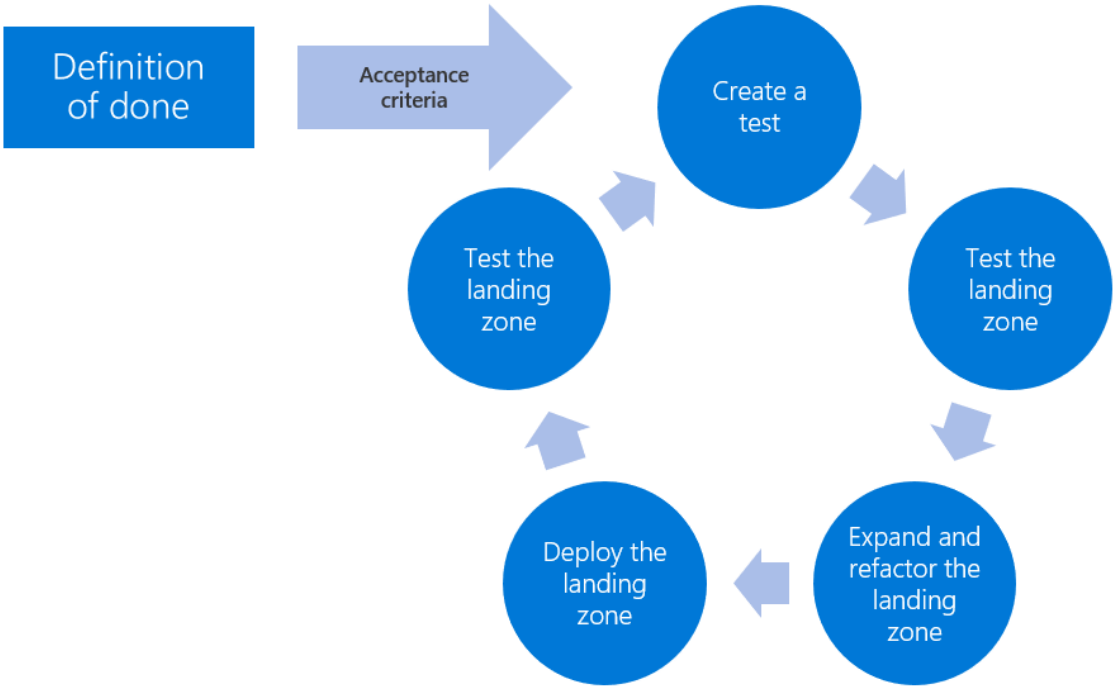
Next steps

To get started on a refactoring process, get started using Azure landing zones.

Test-driven development (TDD) for landing zones

3/6/2021 • 5 minutes to read • Edit Online

Test-driven development is a common software development and DevOps process that improves the quality of new features and improvements in any code-based solution. Cloud-based infrastructure, and the underlying source code can use this process to ensure landing zones meet core requirements and are of high quality. This process is especially useful when landing zones are being developed and refactored in a parallel development effort.



In the cloud, infrastructure is the output of code execution. Well-structured, tested, and verified code produces a viable landing zone. A landing zone is an environment for hosting your workloads, preprovisioned through code. It includes foundational capabilities using a defined set of cloud services and best practices that set you up for success. This guidance describes an approach that uses test-driven development to fulfill the last part of that definition, while meeting quality, security, operations, and governance requirements.

This approach can be used to meet simple feature requests during early development. Later in the cloud adoption lifecycle, this process can be used to meet security, operations, governance, or compliance requirements.

Definition of done

"Set up for success" is a subjective statement. This statement provides the cloud platform team with little actionable information during landing zone development or refactoring efforts. This lack of clarity can lead to missed expectations and vulnerabilities in a cloud environment. Before refactoring or expanding any landing zone, the cloud platform team should seek clarity regarding the "definition of done" for each landing zone.

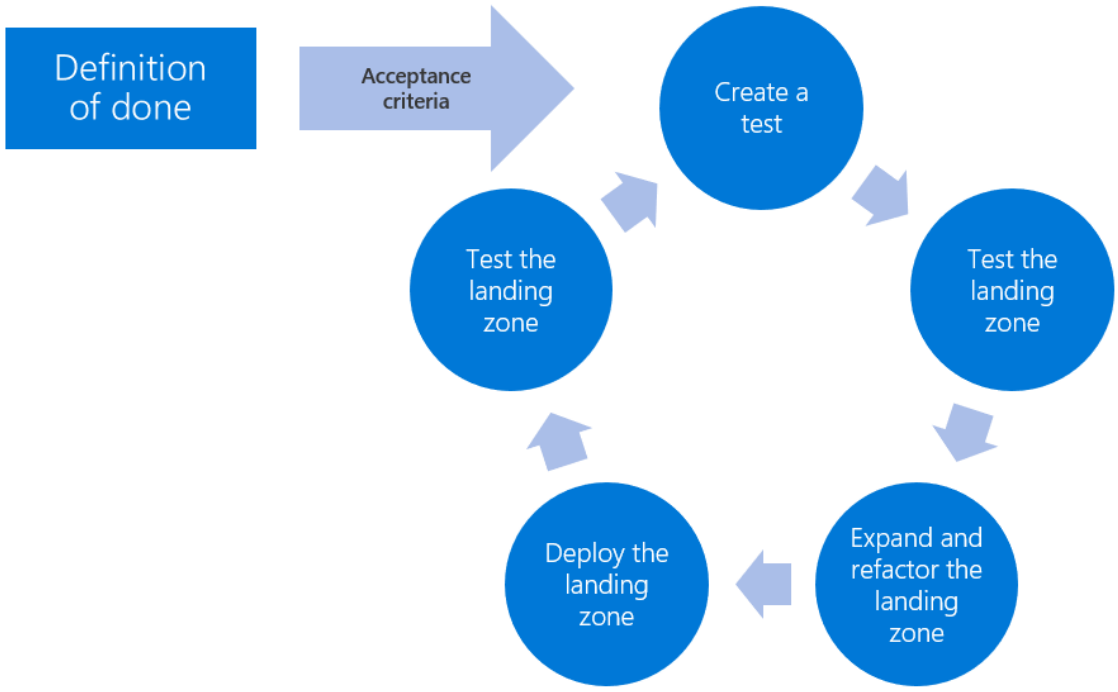
Definition of done is a simple agreement between the cloud platform team and other affected teams. This agreement outlines the expected value added features, which should be included in any landing zone development effort. The definition of done is often a checklist that's aligned with the short-term cloud adoption plan. In mature processes, those expected features in the checklist will each have their own acceptance criteria to

create even more clarity. When the value-added features each meet the acceptance criteria, the landing zone is sufficiently configured to enable the success of the current wave or release of adoption effort.

As teams adopt additional workloads and cloud features, the definition of done and acceptance criteria will become increasingly more complex.

Test-driven development cycle

The cycle that makes test-driven development effective is often referred to as a red/green test. In this approach, the cloud platform team starts with a failed test (red test) based on the definition of done and defined acceptance criteria. For each feature or acceptance criteria, the cloud platform team would complete development tasks until the test passes (green test). A test-driven development cycle (or red/green test) would repeat the basic steps in the following image and list below until the full definition of done can be met.



- **Create a test:** Define a test to validate that acceptance criteria for a specific value-add feature has been met. Automate the test whenever possible.
- **Test the landing zone:** Run the new test and any existing tests. If the required feature hasn't already been met by prior development efforts and isn't inclusive to the cloud provider's offering, the test should fail. Running existing tests will help validate that your new test doesn't reduce reliability of landing zone features delivered by existing code.
- **Expand and refactor the landing zone:** Add or modify the source code to fulfill the requested value-add feature and improve the general quality of the code base. To meet the fullest spirit of test-driven development, the cloud platform team would only add code to meet the requested feature and nothing more. At the same time, code quality and maintenance is a shared effort. When fulfilling new feature requests, the cloud platform team should seek to improve the code by removing duplication and clarifying the code. Running tests between new code creation and refactoring of source code is highly suggested.
- **Deploy the landing zone:** Once the source code is capable of fulfilling the feature request, deploy the modified landing zone to the cloud provider in a controlled testing or sandbox environment.
- **Test the landing zone:** Retesting the landing zone should validate that the new code meets the acceptance criteria for the requested feature. Once all tests pass, the feature is considered complete and the acceptance criteria are considered to be met.

When all value-added features and acceptance criteria pass their associated tests, the landing zone is ready to support the next wave of the cloud adoption plan.

Simple example of a definition of done

For an initial migration effort, definition of done may be overly simple. The following is an example of one of these overly simple examples.

- The initial landing zone will be used to host 10 workloads for initial learning purposes. These workloads are not critical to the business and have no access to sensitive data. In the future, it's likely these workloads will be released to production but criticality and sensitive is not expected to change. To support these workloads, the cloud adoption team will need the following criteria met:
- Network segmentation to align with proposed network design.
- Access to compute, storage, and networking resources to host the workloads aligned to the digital estate discovery.
- Naming and tagging schema for ease of use.
- This environment should be treated as a perimeter network with access to the public internet.
- During adoption efforts, the cloud adoption team would like temporary access to the environment to change service configurations.
- For awareness only: prior to production release, these workloads will require integration with the corporate identity provider to govern ongoing identity and access for operations management purposes. At which time the cloud adoption team's access should be revoked.

The last point above is not a feature or acceptance criteria. But it is an indicator that additional expansions will be required and should be explored with other teams early.

Additional examples of a definition of done

The Govern methodology within the Cloud Adoption Framework provides a narrative journey through the natural maturity of a governance team. Embedded in that journey are several examples of "definition of done" and "acceptance criteria", in the form of policy statements.

- Initial policy statements: Example of corporate policies governing and initial definition of done based on early stage adoption requirements.
- Identity expansion: Example of corporate policies governing *definition of done* to meet requirements to expand identity management for a landing zone.
- Security expansion: Example of corporate policies governing *definition of done* to meet security requirements aligned to the reference cloud adoption plan.
- Operations expansion: Example of corporate policies governing *definition of done* to meet basic operations management requirements.
- Cost expansion: Example of corporate policies governing *definition of done* to meet cost management requirements.

The above examples are basic samples to help develop a *definition of done* for your landing zones. Additional sample policies are available for each of the Five Disciplines of Cloud Governance.

Next steps

To accelerate test-driven development in Azure, review test-driven development features of Azure.

Test-driven development in Azure

Test-driven development for landing zones in Azure

5/18/2021 • 3 minutes to read • Edit Online

As outlined in the previous article on test-driven development (TDD) for landing zones, TDD cycles begin with a test that validates the acceptance criteria of a specific feature required to deliver the cloud adoption plan. Expanding or refactoring the landing zone can then be tested to validate that the acceptance criteria have been met. This article outlines a cloud-native toolchain in Azure to automate test-driven development cycles.

Azure tools to support landing zone TDD cycles

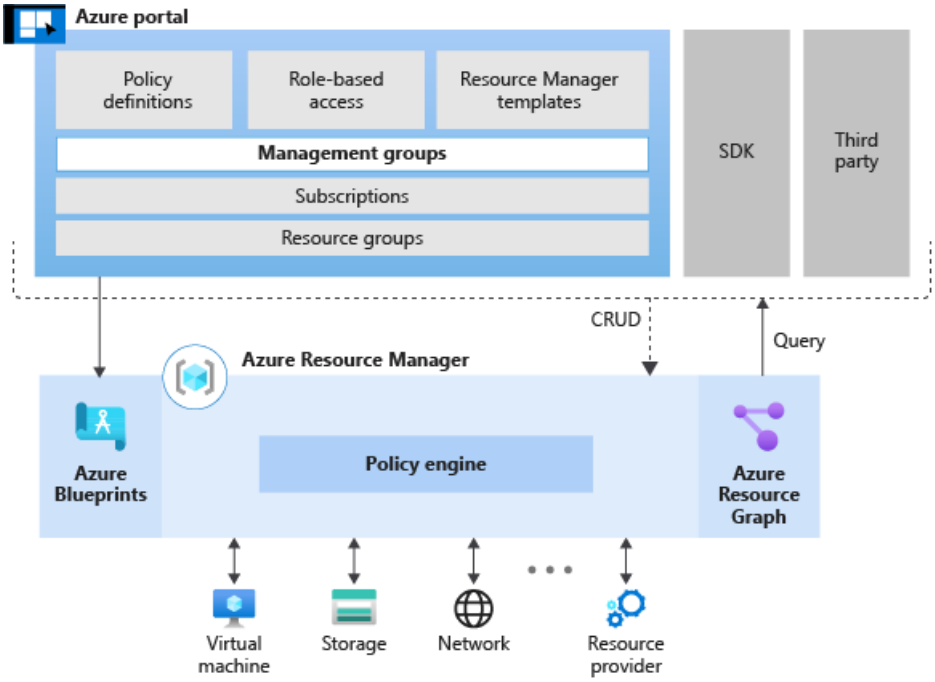


Figure 1: Test-driven development tools in Azure.

The toolchain of Azure-native governance products and services can be easily integrated into test-driven development for the creation of landing zones. Each of these tools serves a specific purpose, making it easier to develop, test, and deploy your landing zone in alignment with TDD cycles.

Microsoft-provided test and deployment templates to accelerate TDD

The following examples are provided by Microsoft for governance purposes. Each can be used as a test or series of tests in a test-driven development cycle for landing zones. The following sections provide more information on each tool:

- Azure Blueprints provides various blueprint samples, including policies for testing and templates for deployment. These blueprint samples can accelerate development, deployment, and testing efforts in TDD cycles.
- Azure Policy also includes built-in policy initiatives, which could be used to test and enforce the full definition of done for a landing zone. Azure Policy includes built-in policy definitions that can meet individual acceptance criteria within the definition of done.
- Azure Graph includes advanced query samples, which can be used to understand how the workloads are deployed within a landing zone for advanced testing scenarios.
- Azure quickstart templates provides of source code templates to aid in accelerating landing zone and

workload deployment.

The examples listed above can be used as tools for accelerating TDD cycles. They run on the governance tools in the following sections and allow cloud platform teams to create their own source code and tests.

Azure governance tools that can accelerate TDD cycles

Azure Policy: When deployments or attempted deployments deviate from governance policies, Azure Policy can provide automated detection, protection, and resolution. But Azure Policy also provides the primary mechanism for testing acceptance criteria in your definition of done. In a TDD cycle, a policy definition can be created to test a single acceptance criteria. Likewise, all acceptance criteria can be added to a policy initiative assigned to the entire subscription. This approach provides a mechanism for red tests before modifying the landing zone. Once the landing zone meets the definition of done, it can be used to enforce the test criteria to avoid code changes that would cause the test to fail in future releases.

Azure Blueprints: Azure blueprint groups policies and other deployment tools into a repeatable package that can be assigned to multiple landing zones. Blueprints prove useful when multiple adoption efforts share common definitions of done, which you may want to update over time. It can also help with deployment during subsequent efforts to expand and refactor landing zones.

Azure Resource Graph: Resource Graph provides a query language for creating data-driven tests based on information about the assets deployed within a landing zone. Later in the adoption plan, this tool can also define complex tests based on the interactions between workload assets and the underlying cloud environment.

Azure Resource Manager templates: These templates provide the primary source code for any environment deployed in Azure. Some third-party tools like Terraform generate their own ARM templates, which are then submitted to Azure Resource Manager.

Azure Resource Manager: Resource Manager provides a consistent platform for build and deploy functions. This platform can deploy landing zones based on source code definitions.

Next steps

To begin refactoring your first landing zone, evaluate basic landing zone considerations.

Basic landing zone considerations

Best practices for Azure readiness

5/19/2021 • 3 minutes to read • Edit Online

Cloud readiness requires equipping staff with the technical skills needed to start a cloud adoption effort and prepare your migration target environment for the assets and workloads you'll move to the cloud. Read these best practices and more guidance to help your team prepare your Azure environment.

Azure fundamentals

Organize and deploy your assets in the Azure environment.

- Azure fundamental concepts. Learn key Azure concepts and terms, and how these concepts relate to one another.
- Create your initial subscriptions. Establish an initial set of Azure subscriptions to begin your cloud adoption.
- Scale your Azure environment using multiple subscriptions. Understand reasons and strategies for creating additional subscriptions to scale your Azure environment.
- Organize your resources with Azure management groups. Learn how Azure management groups can manage resources, roles, policies, and deployment across multiple subscriptions.
- Follow recommended naming and tagging conventions. Review detailed recommendations for naming and tagging your resources. These recommendations support enterprise cloud adoption efforts.
- Create hybrid cloud consistency. Create hybrid cloud solutions that provide the benefits of cloud innovation while maintaining many of the conveniences of on-premises management.

Networking

Prepare your cloud networking infrastructure to support your workloads.

- Networking decisions. Choose the networking services, tools, and architectures that will support your organization's workload, governance, and connectivity requirements.
- Virtual network planning. Plan virtual networks based on your isolation, connectivity, and location requirements.
- Best practices for network security. Learn best practices for addressing common network security issues using built-in Azure capabilities.
- Perimeter networks. Enable secure connectivity between your cloud networks and your on-premises or physical datacenter networks, along with any connectivity to and from the internet.
- Hub and spoke network topology. Efficiently manage common communication or security requirements for complicated workloads and address potential Azure subscription limitations.

Identity and access control

Design your identity and access control infrastructure to improve the security and management efficiency of your workloads.

- Azure identity management and access control security best practices. Learn best practices for identity management and access control using built-in Azure capabilities.
- Best practices for Azure role-based access control. Enable fine-grained and group-based access management for resources organized around user roles.
- Securing privileged access for hybrid and cloud deployments in Azure Active Directory. Ensure that your organization's administrative access and privileged accounts are secure across your cloud and on-premises

environment.

Storage

- Azure Storage guidance. Select the right Azure Storage solution to support your usage scenarios.
- Azure Storage security guide. Learn about security features in Azure Storage.

Databases

- Choose the correct SQL Server option in Azure. Choose the PaaS or IaaS solution that best supports your SQL Server workloads.
- Database security best practices. Learn best practices for database security on the Azure platform.
- Choose the right data store. Select the right data store to meet your requirements. Hundreds of implementation choices are available among SQL and NoSQL databases. Data stores are often categorized by how they structure data and the types of operations they support. This article describes several common storage models.

AI + Machine Learning

- Organize and set up Azure Machine Learning environments. Learn about key decision points in setting up Azure Machine Learning for your organization.

Cost management

- Tracking costs across business units, environments, and projects. Learn best practices for creating proper cost-tracking mechanisms.
- How to optimize your cloud investment with Azure Cost Management + Billing. Implement a strategy for cost management and learn about the tools available for addressing cost challenges.
- Create and manage budgets. Learn to create and manage budgets using Azure Cost Management + Billing.
- Export cost data. Learn to export cost data using Azure Cost Management + Billing.
- Optimize costs based on recommendations. Learn to identify underutilized resources and reduce costs by using Azure Cost Management + Billing and Azure Advisor.
- Use cost alerts to monitor usage and spending. Learn to use Azure Cost Management + Billing alerts to monitor your Azure usage and spending.

Create your initial Azure subscriptions

3/6/2021 • 2 minutes to read • Edit Online

Start your Azure adoption by creating an initial set of subscriptions. Learn what subscriptions you should begin with based on your initial requirements.

Your first two subscriptions

Start by creating two subscriptions:

- Create one Azure subscription to contain your production workloads.
- Create a second subscription to serve as your nonproduction environment, using an Azure Dev/Test offer for lower pricing.



Figure 1: An initial subscription model with keys next to boxes labeled "production" and "nonproduction".

This approach has many benefits:

- Using separate subscriptions for your production and nonproduction environments creates a boundary that makes management of your resources simpler and safer.
- Azure Dev/Test subscription offerings are available for nonproduction workloads. These offerings provide discounted rates on Azure services and software licensing.
- Your production and nonproduction environments will likely have different sets of Azure policies. Using separate subscriptions makes it simple to apply each distinct policy at the subscription level.
- You can allow certain types of Azure resources in your nonproduction subscription for testing purposes. You can enable those resource providers in your nonproduction subscription without making them available in your production environment.
- You can use Azure Dev/Test subscriptions as isolated sandbox environments. These sandboxes allow administrators and developers to rapidly build up and tear down entire sets of Azure resources. This isolation can also help with data protection and security concerns.
- The acceptable cost thresholds that you define will likely vary between production and nonproduction environments.

Sandbox subscriptions

If innovation goals are part of your cloud adoption strategy, consider creating one or more sandbox subscriptions. You can apply security policies to keep these test subscriptions isolated from your production and nonproduction environments. Users can easily experiment with Azure capabilities in these isolated environments. Use an Azure Dev/Test offer to create these subscriptions.



Figure 2: A subscription model with sandbox subscriptions.

Shared services subscription

If you're planning to host **more than 1,000 VMs or compute instances in the cloud within 24 months**, create another Azure subscription to host shared services. This will prepare you to support your end-state enterprise architecture.

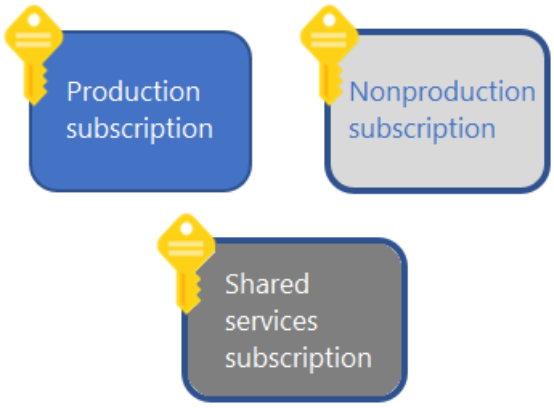


Figure 3: A subscription model with shared services.

Next steps

Review the reasons why you might want to create additional Azure subscriptions to meet your requirements.

Create additional subscriptions to scale your Azure environment

Create additional subscriptions to scale your Azure environment

3/6/2021 • 2 minutes to read • Edit Online

Organizations often use multiple Azure subscriptions to avoid per-subscription resource limits and to better manage and govern their Azure resources. It's important to define a strategy for scaling your subscriptions.

Review fundamental concepts

As you expand your Azure environment beyond your initial subscriptions, it's important to understand Azure concepts such as accounts, tenants, directories, and subscriptions. For more information, see Azure fundamental concepts.

Other considerations might necessitate additional subscriptions. Keep the following in mind as you expand your cloud estate.

Technical considerations

Subscription limits: Subscriptions have defined limits for some resource types. For example, the number of virtual networks in a subscription is limited. When a subscription approaches these limits, you'll need to create another subscription and put additional resources there. For more information, see Azure subscription and service limits.

Classic model resources: If you've been using Azure for a long time, you may have resources that were created using the classic deployment model. Azure policies, Azure role-based access control, resource grouping, and tags cannot be applied to classic model resources. You should move these resources into subscriptions that contain only classic model resources.

Costs: There might be some additional costs for data ingress and egress between subscriptions.

Business priorities

Your business priorities might lead you to create additional subscriptions. These priorities include:

- Innovation
- Migration
- Cost
- Operations
- Security
- Governance

For other considerations about scaling your subscriptions, review the subscription decision guide in the Cloud Adoption Framework.

Moving resources between subscriptions

As your subscription model grows, you might decide that some resources belong in other subscriptions. Many types of resources can be moved between subscriptions. You can also use automated deployments to re-create resources in another subscription. For more information, see Move Azure resources to another resource group or subscription.

Tips for creating new subscriptions

- Identify who is responsible for creating new subscriptions.
- Decide which resource types are available in a subscription by default.
- Decide what all standard subscriptions should look like. Considerations include Azure RBAC access, policies, tags, and infrastructure resources.
- If possible, programmatically create new subscriptions via a service principal. You must grant permission to the service principal to create subscriptions. Define a security group that can request new subscriptions via an automated workflow.
- If you're an Enterprise Agreement (EA) customer, ask Azure support to block creation of non-EA subscriptions for your organization.

Next steps

Create a management group hierarchy to help organize and manage your subscriptions and resources.

Organize and manage your subscriptions and resources

Organize and manage multiple Azure subscriptions

8/25/2021 • 2 minutes to read • Edit Online

If you have only a few subscriptions, then managing them independently is relatively simple. However, if you have many subscriptions, create a management group hierarchy to help manage your subscriptions and resources.

Azure management groups

Azure management groups help you efficiently manage access, policies, and compliance for your subscriptions. Each management group is a container for one or more subscriptions.

Management groups are arranged in a single hierarchy. You define this hierarchy in your Azure Active Directory (Azure AD) tenant to align with your organization's structure and needs. The top level is called the *root management group*. You can define up to six levels of management groups in your hierarchy. Each subscription is contained by only one management group.

Azure provides four levels of management scope:

- Management groups
- Subscriptions
- Resource groups
- Resources

Any access or policy applied at one level in the hierarchy is inherited by the levels below it. A resource owner or subscription owner can't alter an inherited policy. This limitation helps improve governance.

> **NOTE**
> Tag inheritance is not yet supported but will be available soon.

This inheritance model lets you arrange the subscriptions in your hierarchy so that each subscription follows appropriate policies and security controls.

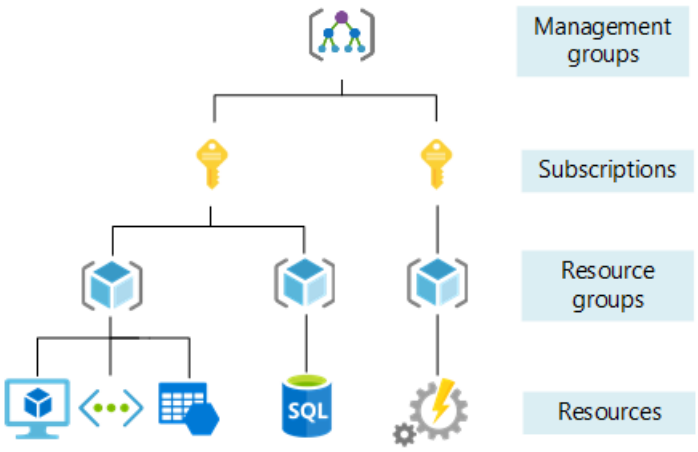


Figure 1: The four scope levels for organizing your Azure resources.

Any access or policy assignment on the root management group applies to all resources in the directory. Carefully consider which items you define at this scope. Include only the assignments you must have.

Create your management group hierarchy

When you define your management group hierarchy, first create the root management group. Then move all existing subscriptions in the directory into the root management group. New subscriptions are always created in the root management group. Later, you can move them to another management group.

When you move a subscription to an existing management group, it inherits the policies and role assignments from the management group hierarchy above it. Once you have established multiple subscriptions for your Azure workloads, you can create additional subscriptions to contain Azure services that other subscriptions share.

If you expect your Azure environment to grow, you should create management groups for production and nonproduction now, and apply appropriate policies and access controls at the management group level. New subscriptions will inherit the appropriate controls as they're added to each management group.

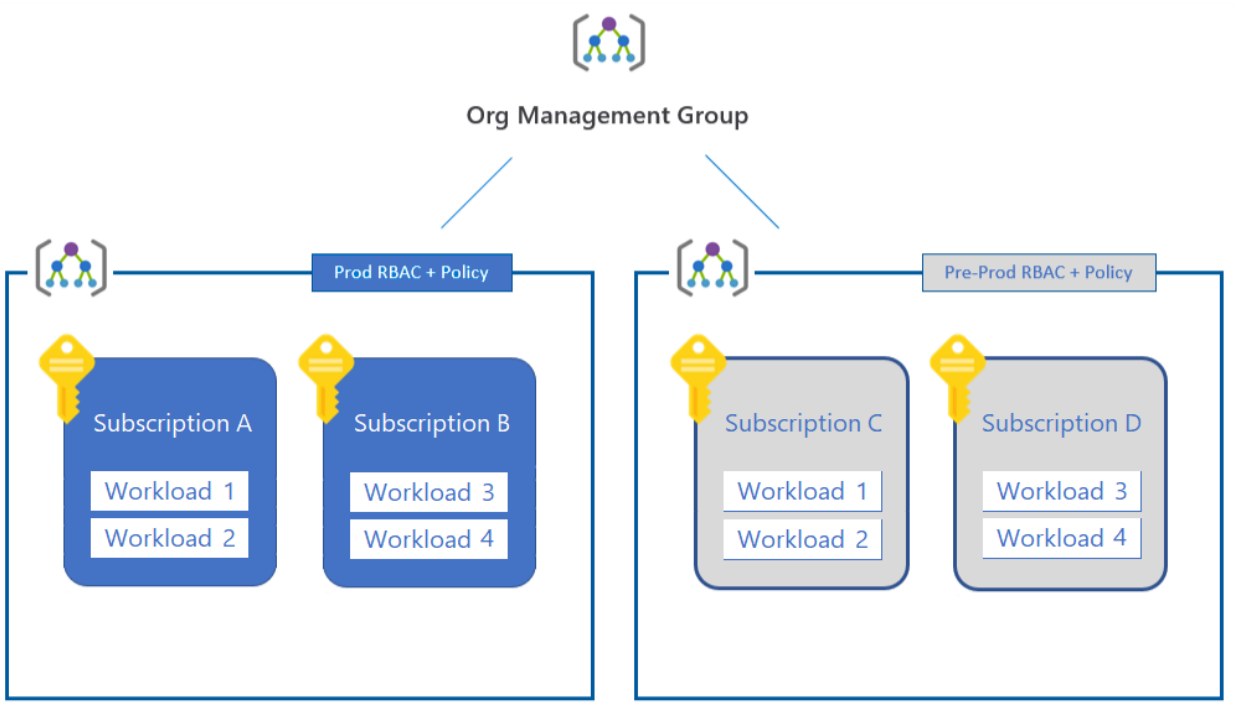


Figure 2: An example of a management group hierarchy.

Example use cases

Some basic examples of using management groups to separate different workloads include:

Production versus nonproduction workloads: Use management groups to more easily manage different roles and policies between production and nonproduction subscriptions. For example, developers might have contributor access in nonproduction subscriptions but only reader access in production subscriptions.

Internal services versus external services: Enterprises often have different requirements, policies, and roles for internal services versus external customer-facing services.

Related resources

Review the following resources to learn more about organizing and managing your Azure resources.

- Organize your resources with Azure management groups
- Elevate access to manage all Azure subscriptions and management groups
- Move Azure resources to another resource group or subscription

Next steps

Review recommended naming and tagging conventions to follow when deploying your Azure resources.

Recommended naming and tagging conventions

Develop your naming and tagging strategy for Azure resources

7/21/2021 • 2 minutes to read • Edit Online

Organize your cloud assets to support governance, operational management, and accounting requirements. Well-defined naming and metadata tagging conventions help to quickly locate and manage resources. These conventions also help associate cloud usage costs with business teams via chargeback and showback accounting mechanisms.

Define your naming and tagging strategy as early as possible. Use the following links to help you define and implement your strategy:

- Define your naming convention
- Recommended abbreviations for Azure resource types
- Define your tagging strategy
- Resource naming and tagging decision guide
- Naming rules and restrictions for Azure resources

> **NOTE**
> Every business has its own organizational and management requirements. These recommendations help start a discussion with your cloud adoption teams. As the discussion proceeds, use the template below to document the naming and tagging decisions you make when aligning these recommendations to your specific business needs.
>
> Download the naming and tagging conventions tracking template.

Purpose of naming and tagging

Accurately representing and naming your resources is essential for security purposes. In the event of a security incident, it's critical to quickly identify affected systems, what functions those systems support, and the potential business impact. Security services such as Azure Security Center and Azure Sentinel reference resources and their associated logging and alert information by resource name.

Azure defines naming rules and restrictions for Azure resources. This guidance provides detailed recommendations to support enterprise cloud adoption efforts.

Changing resource names can be difficult. Establish a comprehensive naming convention before you begin any large cloud deployment.

Naming and tagging strategy

A naming and tagging strategy includes business and operational details as components of resource names and metadata tags:

- The business side of this strategy ensures that resource names and tags include the organizational information needed to identify the teams. Use a resource along with the business owners who are responsible for resource costs.
- The operational side ensures that names and tags include information that IT teams use to identify the workload, application, environment, criticality, and other information useful for managing resources.

Next steps

Learn about the considerations for defining your naming convention your Azure resources and assets, and review example names for resources and assets in Azure.

Name your Azure resources and assets

Define your naming convention

8/13/2021 • 6 minutes to read • Edit Online

An effective naming convention composes resource names from important information about each resource. A well-chosen name helps you quickly identify the resource's type, its associated workload, its deployment environment, and the Azure region hosting it. For example, a public IP resource for a production SharePoint workload residing in the West US region might be `pip-sharepoint-prod-westus-001`.

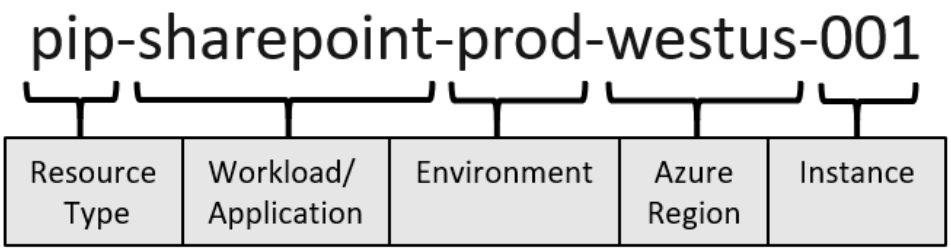


Diagram 1: Components of an Azure resource name.

Naming scope

All Azure resource types have a scope that defines the level that resource names must be unique. A resource must have a unique name within its scope.

For example, a virtual network has a resource group scope, which means that there can be only one network named `vnet-prod-westus-001` in a given resource group. Other resource groups could have their own virtual network named `vnet-prod-westus-001`. Subnets are scoped to virtual networks, so each subnet within a virtual network must have a distinct name.

Some resource names, such as PaaS services with public endpoints or virtual machine DNS labels, have global scopes, so they must be unique across the entire Azure platform.

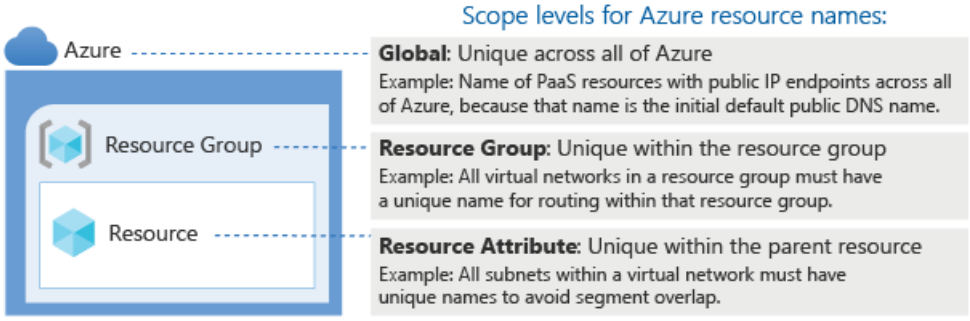


Diagram 2: Scope levels for Azure resource names.

Resource names have length limits. Balancing the context embedded in a name with its scope and length limit is important when you develop your naming conventions. For more information, see naming rules and restrictions for Azure resources.

Recommended naming components

When you construct your naming convention, identify the key pieces of information that you want to reflect in a resource name. Different information is relevant for different resource types. The following list provides examples of information that are useful when you construct resource names.

Keep the length of naming components short to prevent exceeding resource name length limits.

NAMING COMPONENT	DESCRIPTION
Resource type	An abbreviation that represents the type of Azure resource or asset. This component is often used as a prefix or suffix in the name. For more information, see Recommended abbreviations for Azure resource types. Examples: `rg`, `vm`
Business unit	Top-level division of your company that owns the subscription or workload the resource belongs to. In smaller organizations, this component might represent a single corporate top-level organizational element. Examples: `fin`, `mktg`, `product`, `it`, `corp`
Application or service name	Name of the application, workload, or service that the resource is a part of. Examples: `navigator`, `emissions`, `sharepoint`, `hadoop`
Subscription type	Summary description of the purpose of the subscription that contains the resource. Often broken down by deployment environment type or specific workloads. Examples: `prod`, `shared`, `client`
Deployment environment	The stage of the development lifecycle for the workload that the resource supports. Examples: `prod`, `dev`, `qa`, `stage`, `test`
Region	The Azure region where the resource is deployed. Examples: `westus`, `eastus2`, `westeu`, `usva`, `ustx`

Example names for common Azure resource types

The following section provides some example names for common Azure resource types in an enterprise cloud deployment.

> **NOTE**
>
> Some of these example names use a three-digit padding scheme (`###`), such as `mktg-prod-001`.
>
> Padding improves readability and sorting of assets when those assets are managed in a configuration management database (CMDB), IT Asset Management tool, or traditional accounting tools. When the deployed asset is managed centrally as part of a larger inventory or portfolio of IT assets, the padding approach aligns with interfaces those systems use to manage inventory naming.
>
> Unfortunately, the traditional asset padding approach can prove problematic in infrastructure-as-code approaches which may iterate through assets based on a non-padded number. This approach is common during deployment or automated configuration management tasks. Those scripts would have to routinely strip the padding and convert the padded number to a real number, which slows script development and run time.
>
> Choose an approach that's suitable for your organization. The padding shown here illustrates the importance of using a consistent approach to inventory numbering, rather than which approach is superior. Before choosing a numbering scheme (with or without padding), evaluate what will affect long-term operations more: CMDB/asset management solutions or code-based inventory management. Then consistently follow the padding option that best fits your operational needs.

Example names: General

ASSET TYPE	SCOPE	FORMAT AND EXAMPLES
Management group	Business unit and/or environment type	*mg-<business unit>[-<environment type>]* • `mg-mktg` • `mg-hr` • `mg-corp-prod` • `mg-fin-client`
Subscription	Account / enterprise agreement	*<business unit>-<subscription type>-<###>* • `mktg-prod-001` • `corp-shared-001` • `fin-client-001`
Resource group	Subscription	*rg-<app or service name>-<subscription type>-<###>* • `rg-mktgsharepoint-prod-001` • `rg-acctlookupsvc-shared-001` • `rg-ad-dir-services-shared-001`
API management service instance	Global	*apim-<app or service name>* `apim-navigator-prod`
Managed identity	Resource group	*id-<app or service name>-<environment type>-<region name>-<###>* • `id-appcn-keda-prod-eastus2-001`

Example names: Networking

ASSET TYPE	SCOPE	FORMAT AND EXAMPLES
Virtual network	Resource group	*vnet-<subscription type>-<region>-<###>* • `vnet-shared-eastus2-001` • `vnet-prod-westus-001` • `vnet-client-eastus2-001`
Subnet	Virtual network	*snet-<subscription>-<region>-<###>* • `snet-shared-eastus2-001` • `snet-prod-westus-001` • `snet-client-eastus2-001`

ASSET TYPE	SCOPE	FORMAT AND EXAMPLES
Network interface (NIC)	Resource group	*nic-<##>-<vm name>-<subscription>-<###>* • `nic-01-dc1-shared-001` • `nic-02-vmhadoop1-prod-001` • `nic-02-vmtest1-client-001`
Public IP address	Resource group	*pip-<vm name or app name>-<environment>-<region>-<###>* • `pip-dc1-shared-eastus2-001` • `pip-hadoop-prod-westus-001`
Load balancer	Resource group	*lb-<app name or role>--<###>* • `lb-navigator-prod-001` • `lb-sharepoint-dev-001`
Network security group (NSG)	Subnet or NIC	*nsg-<policy name or app name>-<###>* • `nsg-weballow-001` • `nsg-rdpallow-001` • `nsg-sqlallow-001` • `nsg-dnsblocked-001`
Local network gateway	Virtual gateway	*lgw-<subscription type>-<region>-<###>* • `lgw-shared-eastus2-001` • `lgw-prod-westus-001` • `lgw-client-eastus2-001`
Virtual network gateway	Virtual network	*vgw-<subscription type>-<region>-<###>* • `vgw-shared-eastus2-001` • `vgw-prod-westus-001` • `vgw-client-eastus2-001`
Site-to-Site connection	Resource group	*cn-<local gateway name>-to-<virtual gateway name>* • `cn-lgw-shared-eastus2-001-to-vgw-shared-eastus2-001` • `cn-lgw-shared-eastus2-001-to-vgw-shared-westus-001`

ASSET TYPE	SCOPE	FORMAT AND EXAMPLES
VPN connection	Resource group	*cn-<subscription1>-<region1>-to-<subscription2>-<region2>-* • `cn-shared-eastus2-to-shared-westus` • `cn-prod-eastus2-to-prod-westus`
Route table	Resource group	*route-<route table name>* • `route-navigator` • `route-sharepoint`
DNS label	Global	*<DNS A record for VM>.<region>.cloudapp.azure.com* • `dc1.westus.cloudapp.azure.com` • `web1.eastus2.cloudapp.azure.com`

Example names: Compute and Web

ASSET TYPE	SCOPE	FORMAT AND EXAMPLES
Virtual machine	Resource group	*vm<policy name or app name><###>* • `vmnavigator001` • `vmsharepoint001` • `vmsqlnode001` • `vmhadoop001`
VM storage account	Global	*stvm<performance type><app name or prod name><region><###>* • `stvmstcoreeastus2001` • `stvmpmcoreeastus2001` • `stvmstplmeastus2001` • `stvmsthadoopeastus2001`
Web app	Global	*app-<app name>-<environment>-<###>.azurewebsites.net* • `app-navigator-prod-001.azurewebsites.net` • `app-accountlookup-dev-001.azurewebsites.net`

ASSET TYPE	SCOPE	FORMAT AND EXAMPLES
Function app	Global	*func-<app name>-<environment>-<###>.azurewebsites.net* • `func-navigator-prod-001.azurewebsites.net` • `func-accountlookup-dev-001.azurewebsites.net`
Cloud service	Global	*cld-<app name>-<environment>-<###>.cloudapp.net}* • `cld-navigator-prod-001.azurewebsites.net` • `cld-accountlookup-dev-001.azurewebsites.net`
Notification Hubs namespace	Global	*ntfns-<app name>-<environment>* • `ntfns-navigator-prod` • `ntfns-emissions-dev`
Notification hub	Notification Hubs namespace	*ntf-<app name>-<environment>* • `ntf-navigator-prod` • `ntf-emissions-dev`

Example names: Databases

ASSET TYPE	SCOPE	FORMAT AND EXAMPLES
Azure SQL Database server	Global	*sql-<app name>-<environment>* • `sql-navigator-prod` • `sql-emissions-dev`
Azure SQL database	Azure SQL Database	*sqldb-<database name>-<environment>* • `sqldb-users-prod` • `sqldb-users-dev`
Azure Cosmos DB database	Global	*cosmos-<app name>-<environment>* • `cosmos-navigator-prod` • `cosmos-emissions-dev`
Azure Cache for Redis instance	Global	*redis-<app name>-<environment>* • `redis-navigator-prod` • `redis-emissions-dev`

ASSET TYPE	SCOPE	FORMAT AND EXAMPLES
MySQL database	Global	*mysql-<app name>-<environment>* • `mysql-navigator-prod` • `mysql-emissions-dev`
PostgreSQL database	Global	*psql-<app name>-<environment>* • `psql-navigator-prod` • `psql-emissions-dev`
Azure Synapse Analytics	Global	*syn-<app name>-<environment>* • `syn-navigator-prod` • `syn-emissions-dev`
SQL Server Stretch Database	Azure SQL Database	*sqlstrdb-<app name>-<environment>* • `sqlstrdb-navigator-prod` • `sqlstrdb-emissions-dev`

Example names: Storage

ASSET TYPE	SCOPE	FORMAT AND EXAMPLES
Storage account (general use)	Global	*st<storage name><###>* • `stnavigatordata001` • `stemissionsoutput001`
Storage account (diagnostic logs)	Global	*stdiag<first 2 letters of subscription name and number><region><###>* • `stdiagsh001eastus2001` • `stdiagsh001westus001`
Azure StorSimple	Global	*ssimp<app name>-<environment>* • `ssimpnavigatorprod` • `ssimpemissionsdev`
Azure Container Registry	Global	*cr<app name><environment><###>* • `crnavigatorprod001`

Example names: AI and machine learning

ASSET TYPE	SCOPE	FORMAT AND EXAMPLES

ASSET TYPE	SCOPE	FORMAT AND EXAMPLES
Azure Cognitive Search	Global	*srch-<app name>-<environment>* • `srch-navigator-prod` • `srch-emissions-dev`
Azure Cognitive Services	Resource group	*cog-<app name>-<environment>* • `cog-navigator-prod` • `cog-emissions-dev`
Azure Machine Learning workspace	Resource group	*mlw-<app name>-<environment>* • `mlw-navigator-prod` • `mlw-emissions-dev`

Example names: Analytics and IoT

ASSET TYPE	SCOPE	FORMAT AND EXAMPLES
Azure Data Factory	Global	*adf-<app name>-<environment>* • `adf-navigator-prod` • `adf-emissions-dev`
Azure Stream Analytics	Resource group	*asa-<app name>-<environment>* • `asa-navigator-prod` • `asa-emissions-dev`
Data Lake Analytics account	Global	*dla<app name><environment>* • `dlanavigatorprod` • `dlanavigatorprod`
Data Lake Storage account	Global	*dls<app name><environment>* • `dlsnavigatorprod` • `dlsemissionsdev`
Event hub	Global	*evh-<app name>-<environment>* • `evh-navigator-prod` • `evh-emissions-dev`
HDInsight - HBase cluster	Global	*hbase-<app name>-<environment>* • `hbase-navigator-prod` • `hbase-emissions-dev`

ASSET TYPE	SCOPE	FORMAT AND EXAMPLES
HDInsight - Hadoop cluster	Global	*hadoop-<app name>-<environment>* • `hadoop-navigator-prod` • `hadoop-emissions-dev`
HDInsight - Spark cluster	Global	*spark-<app name>-<environment>* • `spark-navigator-prod` • `spark-emissions-dev`
IoT hub	Global	*iot-<app name>-<environment>* • `iot-navigator-prod` • `iot-emissions-dev`
Power BI Embedded	Global	*pbi-<app name>-<environment>* • `pbi-navigator-prod` • `pbi-emissions-dev`

Example names: Integration

ASSET TYPE	SCOPE	FORMAT AND EXAMPLES
Service Bus	Global	*sb-<app name>-<environment>.servicebus.windows.net* • `sb-navigator-prod` • `sb-emissions-dev`
Service Bus queue	Service Bus	*sbq-<query descriptor>* • `sbq-messagequery`
Service Bus topic	Service Bus	*sbt-<query descriptor>* • `sbt-messagequery`

Next steps

Review recommended abbreviations to use for various Azure resource types when naming your resources and assets.

Recommended abbreviations for Azure resource types

Recommended abbreviations for Azure resource types

7/21/2021 • 4 minutes to read • Edit Online

Azure workloads are typically composed of multiple resources and services. Including a naming component in your resource names that represents the type of the Azure resource makes it easier to visually recognize application or service components.

This list provides recommended abbreviations for various Azure resource types to include in your naming conventions. These abbreviations are often used as prefixes in resource names, so each abbreviation is shown below followed by a hyphen (`-`), except for resource types that disallow hyphens in the resource name. Your naming convention might place the resource type abbreviation in a different location of the name if it's more suitable for your organization's needs.

General

ASSET TYPE	RESOURCE PROVIDER NAMESPACE/ENTITY	ABBREVIATION
API management service instance	`Microsoft.ApiManagement/service`	`apim-`
Managed Identity	`Microsoft.ManagedIdentity/userAssigne`	`id-` `tities`
Management group	`Microsoft.Management/managementGroups`	`mg-`
Policy definition	`Microsoft.Authorization/policyDefinit`	`policy-`
Resource group	`Microsoft.Resources/resourceGroups`	`rg-`

Networking

ASSET TYPE	RESOURCE PROVIDER NAMESPACE/ENTITY	ABBREVIATION
Application gateway	`Microsoft.Network/applicationGateways`	`agw-`
Application security group (ASG)	`Microsoft.Network/applicationSecurity`	`asg-`
Bastion	`Microsoft.Network/bastionHosts`	`bas-`
CDN profile	`Microsoft.Cdn/profiles`	`cdnp-`
CDN endpoint	`Microsoft.Cdn/profiles/endpoints`	`cdne-`
Connections	`Microsoft.Network/connections`	`con-`
DNS	`Microsoft.Network/dnsZones`	`dnsz-`

ASSET TYPE	RESOURCE PROVIDER NAMESPACE/ENTITY	ABBREVIATION
DNS zone	`Microsoft.Network/privateDnsZones`	`pdnsz-`
Firewall	`Microsoft.Network/azureFirewalls`	`afw-`
Firewall policy	`Microsoft.Network/firewallPolicies`	`afwp-`
ExpressRoute circuit	`Microsoft.Network/expressRouteCircuit`	`erc-`
Front Door instance	`Microsoft.Network/frontDoors`	`fd-`
Front Door firewall policy	`Microsoft.Network/frontdoorWebApplica`	`fdfp-` `ewallPolicies`
Load balancer (internal)	`Microsoft.Network/loadBalancers`	`lbi-`
Load balancer (external)	`Microsoft.Network/loadBalancers`	`lbe-`
Load balancer rule	`Microsoft.Network/loadBalancers/inbou`	`rule-` `les`
Local network gateway	`Microsoft.Network/localNetworkGateway`	`lgw-`
NAT gateway	Microsoft.Network/natGateways	`ng`
Network interface (NIC)	`Microsoft.Network/networkInterfaces`	`nic-`
Network security group (NSG)	`Microsoft.Network/networkSecurityGrou`	`nsg-`
Network security group (NSG) security rules	`Microsoft.Network/networkSecurityGrou`	`nsgsr-` `ityRules`
Network Watcher	`Microsoft.Network/networkWatchers`	`nw-`
Private Link	`"Microsoft.Network/privateLinkService`	`pl-`
Public IP address	`Microsoft.Network/publicIPAddresses`	`pip-`
Public IP address prefix	`Microsoft.Network/publicIPPrefixes`	`ippre-`
Route filter	`Microsoft.Network/routeFilters`	`rf-`
Route table	`Microsoft.Network/routeTables`	`rt-`
Service endpoint	`Microsoft.serviceEndPointPolicies`	`se-`
Traffic Manager profile	`Microsoft.Network/trafficManagerProfi`	`traf-`
User defined route (UDR)	`Microsoft.Network/routeTables/routes`	`udr-`
Virtual network	`Microsoft.Network/virtualNetworks`	`vnet-`

ASSET TYPE	RESOURCE PROVIDER NAMESPACE/ENTITY	ABBREVIATION
Virtual network peering	`Microsoft.Network/virtualNetworks/vir`	`peer-` `workPeerings`
Virtual network subnet	`Microsoft.Network/virtualNetworks/subı`	`snet-`
Virtual WAN	`Microsoft.Network/virtualWans`	`vwan-`
VPN Gateway	`Microsoft.Network/vpnGateways`	`vpng-`
VPN connection	`Microsoft.Network/vpnGateways/vpnConn`	`vcn-` `s`
VPN site	`Microsoft.Network/vpnGateways/vpnSites`	`vst-`
Virtual network gateway	`Microsoft.Network/virtualNetworkGatewa`	`vgw-`
Web Application Firewall (WAF) policy	`Microsoft.Network/firewallPolicies`	`waf`
Web Application Firewall (WAF) policy rule group	`Microsoft.Network/firewallPolicies/ru`	`wafrg` `s`

Compute and Web

ASSET TYPE	RESOURCE PROVIDER NAMESPACE/ENTITY	ABBREVIATION
App Service environment	`Microsoft.Web/sites`	`ase-`
App Service plan	`Microsoft.Web/serverFarms`	`plan-`
Availability set	`Microsoft.Compute/availabilitySets`	`avail-`
Azure Arc enabled server	`Microsoft.HybridCompute/machines`	`arcs-`
Azure Arc enabled Kubernetes cluster	`Microsoft.Kubernetes/connectedClusters`	`arck`
Cloud service	`Microsoft.Compute/cloudServices`	`cld-`
Disk encryption set	`Microsoft.Compute/diskEncryptionSets`	`des`
Function app	`Microsoft.Web/sites`	`func-`
Gallery	`Microsoft.Compute/galleries`	`gal`
Managed disk (OS)	`Microsoft.Compute/disks`	`osdisk`
Managed disk (data)	`Microsoft.Compute/disks`	`disk`
Notification Hubs	`Microsoft.NotificationHubs/namespaces/`	`ntf-` `icationHubs`

ASSET TYPE	RESOURCE PROVIDER NAMESPACE/ENTITY	ABBREVIATION
Notification Hubs namespace	`Microsoft.NotificationHubs/namespaces`	`ntfns-`
Snapshot	`Microsoft.Compute/snapshots`	`snap-`
Static web app	`Microsoft.Web/staticSites`	`stapp-`
Virtual machine	`Microsoft.Compute/virtualMachines`	`vm`
Virtual machine scale set	`Microsoft.Compute/virtualMachineScale`	`vmss-`
VM storage account	`Microsoft.Storage/storageAccounts`	`stvm`
Web app	`Microsoft.Web/sites`	`app-`

Containers

ASSET TYPE	RESOURCE PROVIDER NAMESPACE/ENTITY	ABBREVIATION
AKS cluster	`Microsoft.ContainerService/managedClu`	`aks-`
Container registry	`Microsoft.ContainerRegistry/registrie`	`cr`
Container instance	`Microsoft.ContainerInstance/containerG`	`ci` `ps`
Service Fabric cluster	`Microsoft.ServiceFabric/clusters`	`sf-`

Databases

ASSET TYPE	RESOURCE PROVIDER NAMESPACE/ENTITY	ABBREVIATION
Azure Cosmos DB database	`Microsoft.DocumentDB/databaseAccounts`	`cosmos-` `ases`
Azure Cache for Redis instance	`Microsoft.Cache/Redis`	`redis-`
Azure SQL Database server	`Microsoft.Sql/servers`	`sql-`
Azure SQL database	`Microsoft.Sql/servers/databases`	`sqldb-`
Azure Synapse Analytics	`Microsoft.Synapse/workspaces`	`syn`
Azure Synapse Analytics Workspaces	`Microsoft.Synapse/workspaces`	`synw`
Azure Synapse Analytics SQL Dedicated Pool	`Microsoft.Synapse/workspaces/sqlPools`	`syndp`
Azure Synapse Analytics Spark Pool	`Microsoft.Synapse/workspaces/sqlPools`	`synsp`

ASSET TYPE	RESOURCE PROVIDER NAMESPACE/ENTITY	ABBREVIATION
MySQL database	`Microsoft.DBforMySQL/servers`	`mysql-`
PostgreSQL database	`Microsoft.DBforPostgreSQL/servers`	`psql-`
SQL Server Stretch Database	`Microsoft.Sql/servers/databases`	`sqlstrdb-`
SQL Managed Instance	`Microsoft.Sql/managedInstances`	`sqlmi-`

Storage

ASSET TYPE	RESOURCE PROVIDER NAMESPACE/ENTITY	ABBREVIATION
Storage account	`Microsoft.Storage/storageAccounts`	`st`
Azure StorSimple	`Microsoft.StorSimple/managers`	`ssimp`

AI and Machine Learning

ASSET TYPE	RESOURCE PROVIDER NAMESPACE/ENTITY	ABBREVIATION
Azure Cognitive Search	`Microsoft.Search/searchServices`	`srch-`
Azure Cognitive Services	`Microsoft.CognitiveServices/accounts`	`cog-`
Azure Machine Learning workspace	`Microsoft.MachineLearningServices/wor`	`mlw-` `s`

Analytics and IoT

ASSET TYPE	RESOURCE PROVIDER NAMESPACE/ENTITY	ABBREVIATION
Azure Analysis Services server	`Microsoft.AnalysisServices/servers`	`as`
Azure Databricks workspace	`Microsoft.Databricks/workspaces`	`dbw-`
Azure Stream Analytics	`Microsoft.StreamAnalytics/cluster`	`asa-`
Azure Data Explorer cluster	`Microsoft.Kusto/clusters`	`dec`
Azure Data Explorer cluster database	`Microsoft.Kusto/clusters/databases`	`dedb`
Azure Data Factory	`Microsoft.DataFactory/factories`	`adf-`
Data Lake Store account	`Microsoft.DataLakeStore/accounts`	`dls`

ASSET TYPE	RESOURCE PROVIDER NAMESPACE/ENTITY	ABBREVIATION
Data Lake Analytics account	`Microsoft.DataLakeAnalytics/accounts`	`dla`
Event Hubs namespace	`Microsoft.EventHub/namespaces`	`evhns-`
Event hub	`Microsoft.EventHub/namespaces/eventHu`	`evh-`
Event Grid domain	`Microsoft.EventGrid/domains`	`evgd-`
Event Grid subscriptions	`Microsoft.EventGrid/eventSubscription`	`evgs-`
Event Grid topic	`Microsoft.EventGrid/domains/topics`	`evgt-`
HDInsight - Hadoop cluster	`Microsoft.HDInsight/clusters`	`hadoop-`
HDInsight - HBase cluster	`Microsoft.HDInsight/clusters`	`hbase-`
HDInsight - Kafka cluster	`Microsoft.HDInsight/clusters`	`kafka-`
HDInsight - Spark cluster	`Microsoft.HDInsight/clusters`	`spark-`
HDInsight - Storm cluster	`Microsoft.HDInsight/clusters`	`storm-`
HDInsight - ML Services cluster	`Microsoft.HDInsight/clusters`	`mls-`
IoT hub	`Microsoft.Devices/IotHubs`	`iot-`
Provisioning services	`Microsoft.Devices/provisioningService`	`provs-`
Provisioning services certificate	`Microsoft.Devices/provisioningService`	`pcert-` `icates`
Power BI Embedded	`Microsoft.PowerBIDedicated/capacities`	`pbi-`
Time Series Insights environment	`Microsoft.TimeSeriesInsights/environm`	`tsi-`

Azure Virtual Desktop

ASSET TYPE	RESOURCE PROVIDER NAMESPACE/ENTITY	ABBREVIATION
Virtual desktop host pool		vdpool-
Virtual desktop application group		vdag-
Virtual desktop workspace		vdws-

Developer tools

ASSET TYPE	RESOURCE PROVIDER NAMESPACE/ENTITY	ABBREVIATION
App Configuration store	`Microsoft.AppConfiguration/configurat`	`appcs-` `s`
SignalR	`Microsoft.SignalRService/SignalR`	`sigr`

Integration

ASSET TYPE	RESOURCE PROVIDER NAMESPACE/ENTITY	ABBREVIATION
Integration account	`Microsoft.Logic/integrationAccounts`	`ia-`
Logic apps	`Microsoft.Logic/workflows`	`logic-`
Service Bus	`Microsoft.ServiceBus/namespaces`	`sb-`
Service Bus queue	`Microsoft.ServiceBus/namespaces/queue`	`sbq-`
Service Bus topic	`Microsoft.ServiceBus/namespaces/topic`	`sbt-`

Management and governance

ASSET TYPE	RESOURCE PROVIDER NAMESPACE/ENTITY	ABBREVIATION
Automation account	`Microsoft.Automation/automationAccoun`	`aa-`
Application Insights	`Microsoft.Insights/components`	`appi-`
Azure Monitor action group	`Microsoft.Insights/actionGroups`	`ag-`
Azure Purview instance	`Microsoft.Purview/accounts`	`pview-`
Blueprint	`Microsoft.Blueprint/blueprints`	`bp-`
Blueprint assignment	`Microsoft.Blueprint/blueprints/artifa`	`bpa-`
Key vault	`Microsoft.KeyVault/vaults`	`kv-`
Log Analytics workspace	`Microsoft.OperationalInsights/workspa`	`log-`

Migration

ASSET TYPE	RESOURCE PROVIDER NAMESPACE/ENTITY	ABBREVIATION
Azure Migrate project	`Microsoft.Migrate/assessmentProjects`	`migr-`

ASSET TYPE	RESOURCE PROVIDER NAMESPACE/ENTITY	ABBREVIATION
Database Migration Service instance	`Microsoft.DataMigration/services`	`dms-`
Recovery Services vault	`Microsoft.RecoveryServices/vaults`	`rsv-`

Deprecated product names

ASSET TYPE	RESOURCE PROVIDER NAMESPACE/ENTITY	ABBREVIATION
Azure SQL Data Warehouse	`Microsoft.Sql/servers`	`sqldw-`

Next steps

Review recommendations for tagging your Azure resources and assets.

Define your tagging strategy

Define your tagging strategy

3/6/2021 • 2 minutes to read • Edit Online

When you apply metadata tags to your cloud resources, you can include information about those assets that couldn't be included in the resource name. You can use that information to perform more sophisticated filtering and reporting on resources. You want these tags to include context about the resource's associated workload or application, operational requirements, and ownership information. This information can be used by IT or business teams to find resources or generate reports about resource usage and billing.

What tags you apply to resources and what tags are required or optional differs among organizations. The following list provides examples of common tags that capture important context and information about a resource. Use this list as a starting point to establish your own tagging conventions.

Minimum suggested tags

The following tags will guide implementation and processes in all subsequent Cloud Adoption Framework methodologies. Many of the best practices in those methodologies demonstrate automation of cloud operations and governance based on the following tags.

TAG NAME	DESCRIPTION	KEY AND EXAMPLE VALUES
Workload name	Name of the workload the resource supports.	*WorkloadName* • `ControlCharts`
Data classification	Sensitivity of data hosted by this resource.	*DataClassification* • `Non-business` • `Public` • `General` • `Confidential` • `Highly confidential`
Business criticality	Business impact of the resource or supported workload.	*Criticality* • `Low` • `Medium` • `High` • `Business unit-critical` • `Mission-critical`
Business unit	Top-level division of your company that owns the subscription or workload that the resource belongs to. In smaller organizations, this tag might represent a single corporate or shared top-level organizational element.	*BusinessUnit* • `Finance` • `Marketing` • `Product XYZ` • `Corp` • `Shared`

TAG NAME	DESCRIPTION	KEY AND EXAMPLE VALUES
Operations commitment	Level of operations support provided for this workload or resource.	*OpsCommitment* • `Baseline only` • `Enhanced baseline` • `Platform operations` • `Workload operations`
Operations team	Team accountable for day-to-day operations.	*OpsTeam* • `Central IT` • `Cloud operations` • `ControlCharts team` • `MSP-{Managed Service Provider name}`

Additional common tagging examples

The following are a number of tags commonly used across Azure to increase visibility into the usage of Azure resources.

TAG NAME	DESCRIPTION	KEY AND EXAMPLE VALUES
Application name	Added granularity, if the workload is subdivided across multiple applications or services.	*ApplicationName* • `IssueTrackingSystem`
Approver name	Person responsible for approving costs related to this resource.	*Approver* • `chris@contoso.com`
Budget required/approved	Money allocated for this application, service, or workload.	*BudgetAmount* • `$200,000`
Cost center	Accounting cost center associated with this resource.	*CostCenter* • `55332`
Disaster recovery	Business criticality of the application, workload, or service.	*DR* • `Mission-critical` • `Critical` • `Essential`
End date of the project	Date when the application, workload, or service is scheduled for retirement.	*EndDate* • `2023-10-15`

TAG NAME	DESCRIPTION	KEY AND EXAMPLE VALUES
Environment	Deployment environment of the application, workload, or service.	*Env* • `Prod` • `Dev` • `QA` • `Stage` • `Test`
Owner name	Owner of the application, workload, or service.	*Owner* • `jane@contoso.com`
Requester name	User who requested the creation of this application.	*Requester* • `john@contoso.com`
Service class	Service level agreement level of the application, workload, or service.	*ServiceClass* • `Dev` • `Bronze` • `Silver` • `Gold`
Start date of the project	Date when the application, workload, or service was first deployed.	*StartDate* • `2020-10-15`

Take action

Review the resource naming and tagging decision guide.

Next steps

Learn how to move resource groups and assets between subscriptions in Azure.

Move resource groups and assets between subscriptions

The virtual datacenter: A network perspective

5/18/2021 • 43 minutes to read • Edit Online

Applications migrated from on-premises will benefit from Azure's secure cost-efficient infrastructure, even with minimal application changes. Even so, enterprises should adapt their architectures to improve agility and take advantage of Azure's capabilities.

Microsoft Azure delivers hyperscale services and infrastructure with enterprise-grade capabilities and reliability. These services and infrastructure offer many choices in hybrid connectivity, so customers can choose to access them over the internet or a private network connection. Microsoft partners can also provide enhanced capabilities by offering security services and virtual appliances that are optimized to run in Azure.

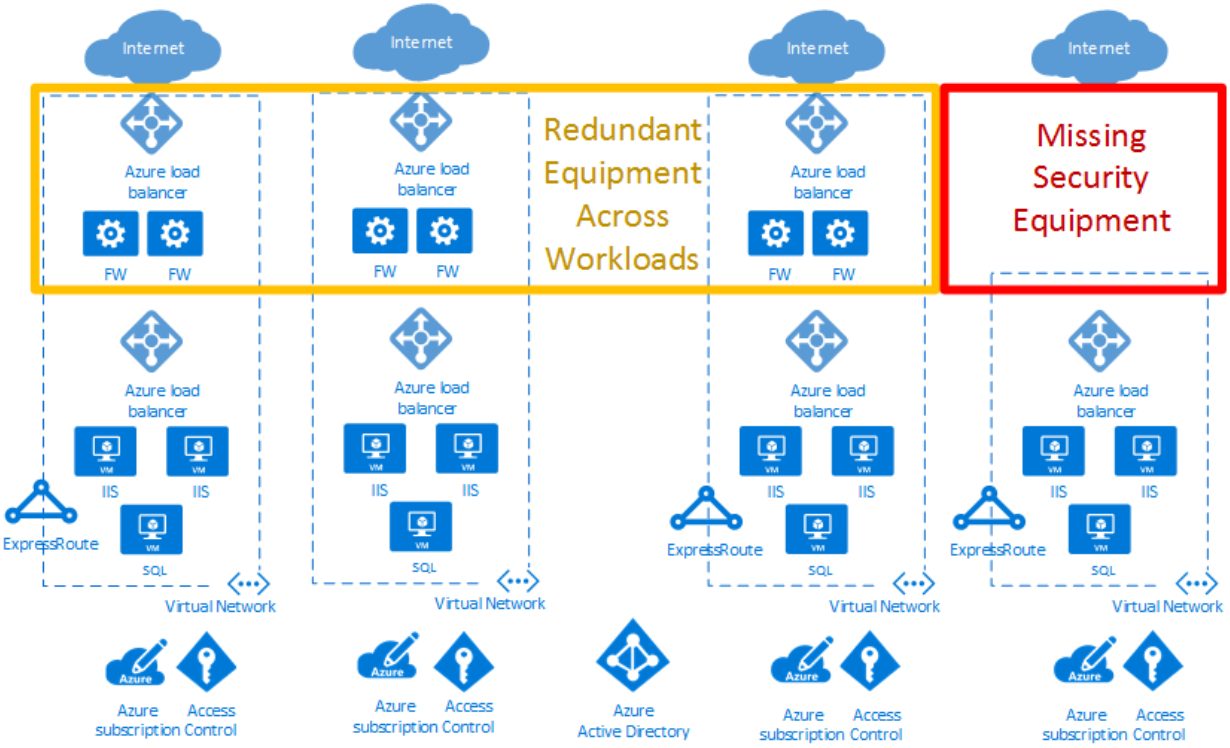
Customers can use Azure to seamlessly extend their infrastructure into the cloud and build multitier architectures.

What is a virtual datacenter?

The cloud began as a platform for hosting public-facing applications. Enterprises recognized the value of the cloud and began migrating internal line-of-business applications. These applications brought additional security, reliability, performance, and cost considerations that required additional flexibility when delivering cloud services. New infrastructure and networking services were designed to provide this flexibility, and new features provided for elastic scale, disaster recovery, and other considerations.

Cloud solutions were initially designed to host single, relatively isolated applications in the public spectrum. This approach worked well for a few years. As the benefits of cloud solutions became clear, multiple large-scale workloads were hosted on the cloud. Addressing security, reliability, performance, and cost concerns of deployments in one or more regions became vital throughout the lifecycle of the cloud service.

In the example cloud deployment diagram below, the red box highlights a security gap. The yellow box shows an opportunity to optimize network virtual appliances across workloads.



Virtual datacenters help achieve the scale required for enterprise workloads. This scale must address the challenges introduced when running large-scale applications in the public cloud.

A virtual datacenter (VDC) implementation includes more than the application workloads in the cloud. It also provides the network, security, management, and other infrastructure such as DNS and Active Directory services. As enterprises migrate additional workloads to Azure, consider the infrastructure and objects that support these workloads. Carefully structuring your resources helps avoid proliferation of hundreds of separately managed "workload islands" with independent data flows, security models, and compliance challenges.

The virtual datacenter concept provides recommendations and high-level designs for implementing a collection of separate but related entities. These entities often have common supporting functions, features, and infrastructure. Viewing your workloads as a virtual datacenter helps realize reduced cost from economies of scale, optimized security via component and data flow centralization, and easier operations, management, and compliance audits.

> **NOTE**
>
> A virtual datacenter is **not** a specific Azure service. Rather, various Azure features and capabilities are combined to meet your requirements. A virtual datacenter is a way of thinking about your workloads and Azure usage to optimize your resources and capabilities in the cloud. It provides a modular approach to providing IT services in Azure while respecting the enterprise's organizational roles and responsibilities.

A virtual datacenter helps enterprises deploy workloads and applications in Azure for the following scenarios:

- Host multiple related workloads.
- Migrate workloads from an on-premises environment to Azure.
- Implement shared or centralized security and access requirements across workloads.
- Mix DevOps and centralized IT appropriately for a large enterprise.

Who should implement a virtual datacenter?

Any customer that has decided to adopt Azure can benefit from the efficiency of configuring a set of resources for common use by all applications. Depending on the size, even single applications can benefit from using the patterns and components used to build a VDC implementation.

Some organizations have centralized teams or departments for IT, networking, security, or compliance. Implementing a VDC can help enforce policy points, separate responsibilities, and ensure the consistency of the underlying common components. Application teams can retain the freedom and control that is suitable for their requirements.

Organizations with a DevOps approach can also use VDC concepts to provide authorized pockets of Azure resources. This method can ensure the DevOps groups have total control within that grouping, at either the subscription level or within resource groups in a common subscription. At the same time, the network and security boundaries stay compliant as defined by a centralized policy in the hub network and centrally managed resource group.

Considerations for implementing a virtual datacenter

When designing a virtual datacenter, consider these pivotal issues:

Identity and directory service

Identity and directory services are key capabilities of both on-premises and cloud datacenters. Identity covers all aspects of access and authorization to services within a VDC implementation. To ensure that only authorized

users and processes access your Azure resources, Azure uses several types of credentials for authentication, including account passwords, cryptographic keys, digital signatures, and certificates. Azure Multi-Factor Authentication provides an additional layer of security for accessing Azure services using strong authentication with a range of easy verification options (phone call, text message, or mobile app notification) that allow customers to choose the method they prefer.

Any large enterprise needs to define an identity management process that describes the management of individual identities, their authentication, authorization, roles, and privileges within or across their VDC. The goals of this process should be to increase security and productivity while reducing cost, downtime, and repetitive manual tasks.

Enterprise organizations may require a demanding mix of services for different lines of business, and employees often have different roles when involved with different projects. The VDC requires good cooperation between different teams, each with specific role definitions, to get systems running with good governance. The matrix of responsibilities, access, and rights can be complex. Identity management in the VDC is implemented through Azure Active Directory (Azure AD) and Azure role-based access control (Azure RBAC).

A directory service is a shared information infrastructure that locates, manages, administers, and organizes everyday items and network resources. These resources can include volumes, folders, files, printers, users, groups, devices, and other objects. Each resource on the network is considered an object by the directory server. Information about a resource is stored as a collection of attributes associated with that resource or object.

All Microsoft online business services rely on Azure Active Directory (Azure AD) for sign-on and other identity needs. Azure Active Directory is a comprehensive, highly available identity and access management cloud solution that combines core directory services, advanced identity governance, and application access management. Azure AD can integrate with on-premises Active Directory to enable single sign-on for all cloud-based and locally hosted on-premises applications. The user attributes of on-premises Active Directory can be automatically synchronized to Azure AD.

A single global administrator isn't required to assign all permissions in a VDC implementation. Instead, each specific department, group of users, or services in the Directory Service can have the permissions required to manage their own resources within a VDC implementation. Structuring permissions requires balancing. Too many permissions can impede performance efficiency, and too few or loose permissions can increase security risks. Azure role-based access control (Azure RBAC) helps to address this problem, by offering fine-grained access management for resources in a VDC implementation.

Security infrastructure

Security infrastructure refers to the segregation of traffic in a VDC implementation's specific virtual network segment. This infrastructure specifies how ingress and egress are controlled in a VDC implementation. Azure is based on a multitenant architecture that prevents unauthorized and unintentional traffic between deployments by using virtual network isolation, access control lists, load balancers, IP filters, and traffic flow policies. Network address translation (NAT) separates internal network traffic from external traffic.

The Azure fabric allocates infrastructure resources to tenant workloads and manages communications to and from virtual machines (VMs). The Azure hypervisor enforces memory and process separation between VMs and securely routes network traffic to guest OS tenants.

Connectivity to the cloud

A virtual datacenter requires connectivity to external networks to offer services to customers, partners, or internal users. This need for connectivity refers not only to the Internet, but also to on-premises networks and datacenters.

Customers control which services can access and be accessed from the public internet. This access is controlled by using Azure Firewall or other types of virtual network appliances (NVAs), custom routing policies by using user-defined routes, and network filtering by using network security groups. We recommend that all internet-facing resources also be protected by the Azure DDoS Protection Standard.

Enterprises may need to connect their virtual datacenter to on-premises datacenters or other resources. This connectivity between Azure and on-premises networks is a crucial aspect when designing an effective architecture. Enterprises have two different ways to create this interconnection: transit over the Internet or via private direct connections.

An Azure Site-to-Site VPN connects on-premises networks to your virtual datacenter in Azure. The link is established through secure encrypted connections (IPsec tunnels). Azure Site-to-Site VPN connections are flexible, quick to create, and typically don't require any additional hardware procurement. Based on industry standard protocols, most current network devices can create VPN connections to Azure over the internet or existing connectivity paths.

ExpressRoute enables private connections between your virtual datacenter and any on-premises networks. ExpressRoute connections don't go over the public Internet, and offer higher security, reliability, and higher speeds (up to 100 Gbps) along with consistent latency. ExpressRoute provides the benefits of compliance rules associated with private connections. With ExpressRoute Direct, you can connect directly to Microsoft routers at either 10 Gbps or 100 Gbps.

Deploying ExpressRoute connections usually involves engaging with an ExpressRoute service provider (ExpressRoute Direct being the exception). For customers that need to start quickly, it's common to initially use Site-to-Site VPN to establish connectivity between a virtual datacenter and on-premises resources. Once your physical interconnection with your service provider is complete, then migrate connectivity over your ExpressRoute connection.

For large numbers of VPN or ExpressRoute connections, Azure Virtual WAN is a networking service that provides optimized and automated branch-to-branch connectivity through Azure. Virtual WAN lets you connect to and configure branch devices to communicate with Azure. Connecting and configuring can be done either manually or by using preferred provider devices through a Virtual WAN partner. Using preferred provider devices allows ease of use, simplification of connectivity, and configuration management. The Azure WAN built-in dashboard provides instant troubleshooting insights that can help save you time, and gives you an easy way to view large-scale site-to-site connectivity. Virtual WAN also provides security services with an optional Azure Firewall and Firewall Manager in your Virtual WAN hub.

Connectivity within the cloud

Azure Virtual Networks and virtual network peering are the basic networking components in a virtual datacenter. A virtual network guarantees an isolation boundary for virtual datacenter resources. Peering allows intercommunication between different virtual networks within the same Azure region, across regions, and even between networks in different subscriptions. Both inside and between virtual networks, traffic flows can be controlled by sets of security rules specified for network security groups, firewall policies (Azure Firewall or network virtual appliances), and custom user-defined routes.

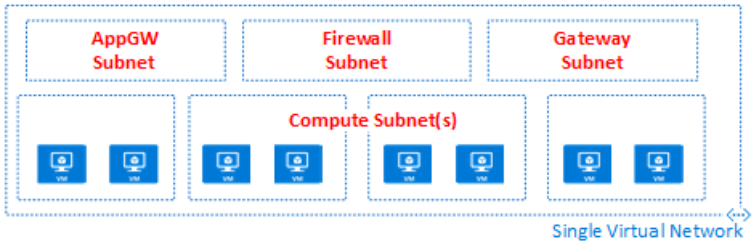
Virtual networks are also anchor points for integrating platform as a service (PaaS) Azure products like Azure Storage, Azure SQL, and other integrated public services that have public endpoints. With service endpoints and Azure Private Link, you can integrate your public services with your private network. You can even take your public services private, but still enjoy the benefits of Azure-managed PaaS services.

Virtual datacenter overview

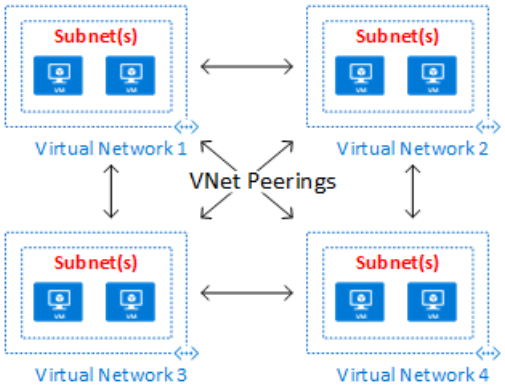
Topologies

A virtual datacenter can be built using one of these high-level topologies, based on your needs and scale requirements:

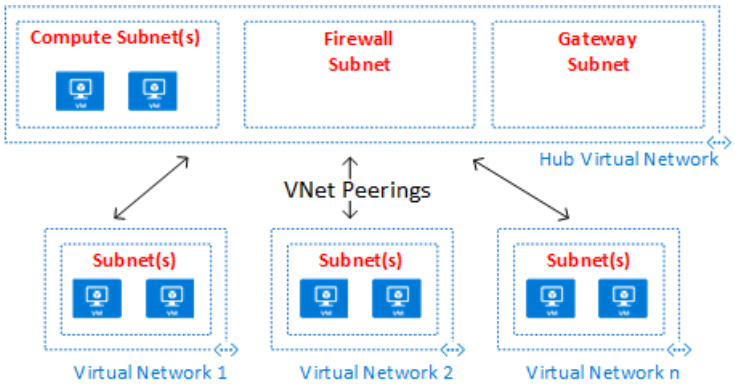
In a *Flat topology*, all resources are deployed in a single virtual network. Subnets allow for flow control and segregation.



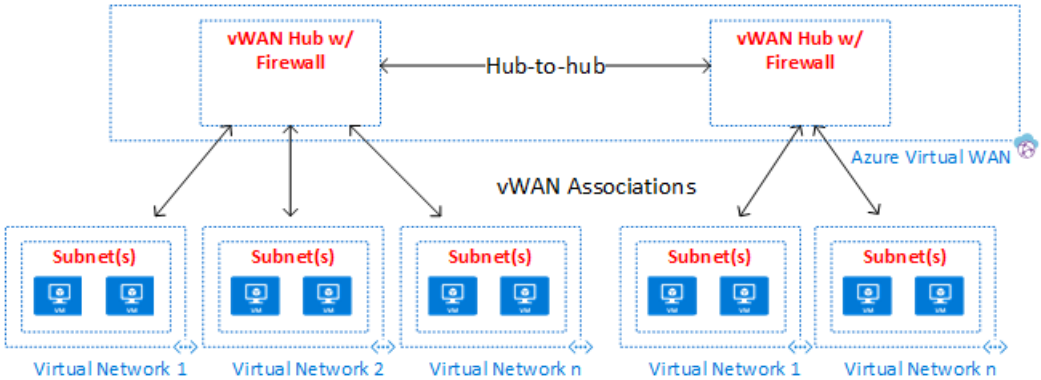
In a *Mesh topology*, virtual network peering connects all virtual networks directly to each other.



A *Peering hub and spoke topology* is well suited for distributed applications and teams with delegated responsibilities.



An *Azure Virtual WAN topology* can support large-scale branch office scenarios and global WAN services.



The peering hub and spoke topology and the Azure Virtual WAN topology both use a hub and spoke design, which is optimal for communication, shared resources, and centralized security policy. Hubs are built using either a virtual network peering hub (labeled as `Hub Virtual Network` in the diagram) or a Virtual WAN hub (labeled as `Azure Virtual WAN` in the diagram). Azure Virtual WAN is designed for large-scale branch-to-branch and branch-to-Azure communications, or for avoiding the complexities of building all the components individually in a virtual networking peering hub. In some cases, your requirements might mandate a virtual network peering hub design, such as the need for network virtual appliances in the hub.

In both of the hub and spoke topologies, the hub is the central network zone that controls and inspects all traffic between different zones: internet, on-premises, and the spokes. The hub and spoke topology helps the IT department centrally enforce security policies. It also reduces the potential for misconfiguration and exposure.

The hub often contains the common service components consumed by the spokes. The following examples are common central services:

- The Windows Active Directory infrastructure, required for user authentication of third parties that access from untrusted networks before they get access to the workloads in the spoke. It includes the related Active Directory Federation Services (AD FS).
- A Distributed Name System (DNS) service to resolve naming for the workload in the spokes, to access resources on-premises and on the internet if Azure DNS isn't used.
- A public key infrastructure (PKI), to implement single sign-on on workloads.
- Flow control of TCP and UDP traffic between the spoke network zones and the internet.
- Flow control between the spokes and on-premises.
- If needed, flow control between one spoke and another.

A virtual datacenter reduces overall cost by using the shared hub infrastructure between multiple spokes.

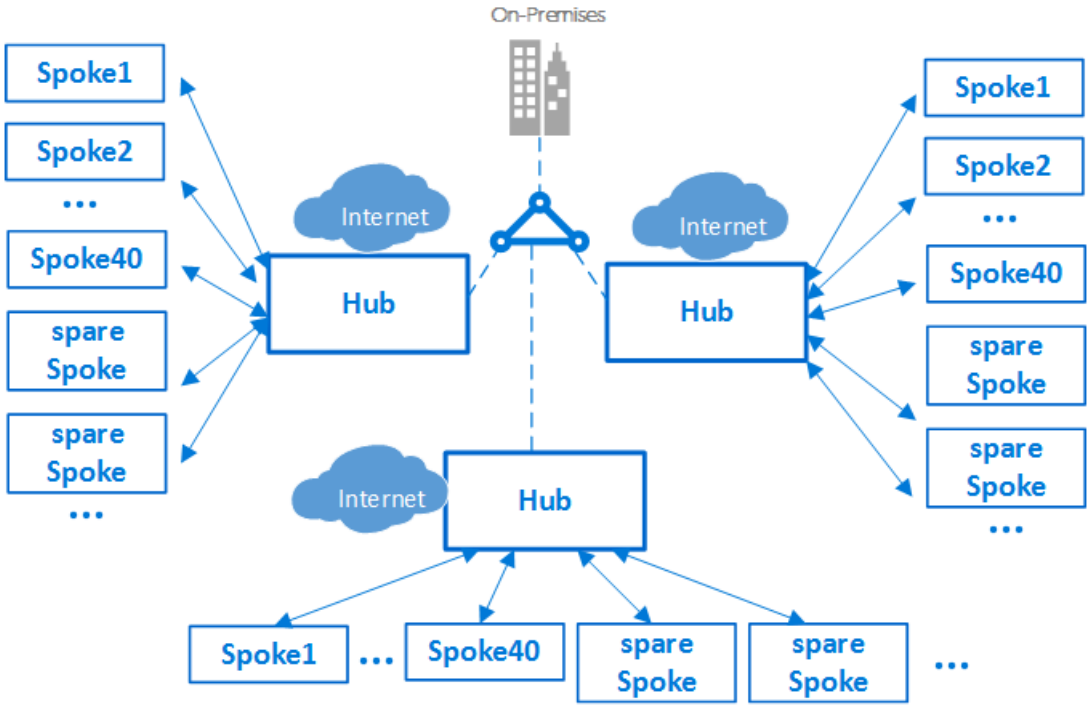
The role of each spoke can be to host different types of workloads. The spokes also provide a modular approach for repeatable deployments of the same workloads. Examples include dev/test, user acceptance testing, preproduction, and production. The spokes can also segregate and enable different groups within your organization. An example is DevOps groups. Inside a spoke, it's possible to deploy a basic workload or complex multitier workloads with traffic control between the tiers.

Subscription limits and multiple hubs

> **IMPORTANT**
>
> Based on the size of your Azure deployments, a multiple hub strategy may be needed. When designing your hub and spoke strategy, ask "can this design scale to use another hub virtual network in this region?", also, "can this design scale to accommodate multiple regions?" It's far better to plan for a design that scales and not need it, than to fail to plan and need it.
>
> When to scale to a secondary (or more) hub will depend on myriad factors, usually based on inherent limits on scale. Be sure to review the subscription, virtual network, and virtual machine limits when designing for scale.

In Azure, every component, whatever the type, is deployed in an Azure subscription. The isolation of Azure components in different Azure subscriptions can satisfy the requirements of different lines of business, such as setting up differentiated levels of access and authorization.

A single VDC implementation can scale up to large number of spokes, although, as with every IT system, there are platform limits. The hub deployment is bound to a specific Azure subscription, which has restrictions and limits (for example, a maximum number of virtual network peerings. For details, see Azure subscription and service limits, quotas, and constraints). In cases where limits may be an issue, the architecture can scale up further by extending the model from a single hub-spokes to a cluster of hub and spokes. Multiple hubs in one or more Azure regions can be connected using virtual network peering, ExpressRoute, Virtual WAN, or Site-to-Site VPN.

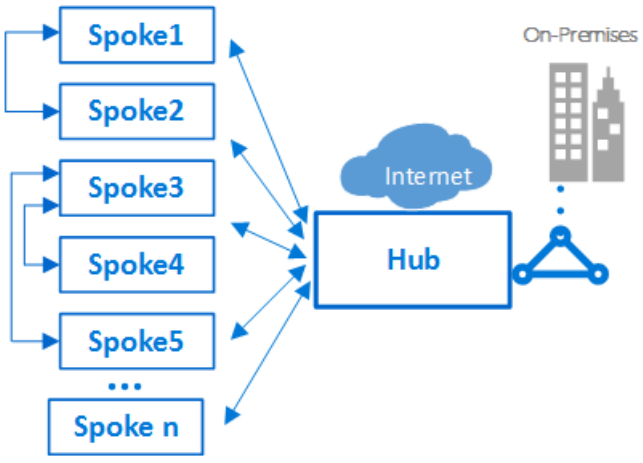


The introduction of multiple hubs increases the cost and management effort of the system. It is only justified due to scalability, system limits, redundancy, regional replication for end-user performance, or disaster recovery. In scenarios requiring multiple hubs, all the hubs should strive to offer the same set of services for operational ease.

Interconnection between spokes

Inside a single spoke, or a flat network design, it's possible to implement complex multitier workloads. Multitier configurations can be implemented using subnets, one for every tier or application, in the same virtual network. Traffic control and filtering are done using network security groups and user-defined routes.

An architect might want to deploy a multitier workload across multiple virtual networks. With virtual network peering, spokes can connect to other spokes in the same hub or different hubs. A typical example of this scenario is the case where application processing servers are in one spoke, or virtual network. The database deploys in a different spoke, or virtual network. In this case, it's easy to interconnect the spokes with virtual network peering and, by doing that, avoid transiting through the hub. A careful architecture and security review should be done to ensure that bypassing the hub doesn't bypass important security or auditing points that might exist only in the hub.



Spokes can also be interconnected to a spoke that acts as a hub. This approach creates a two-level hierarchy: the spoke in the higher level (level 0) becomes the hub of lower spokes (level 1) of the hierarchy. The spokes of a VDC implementation are required to forward the traffic to the central hub so that the traffic can transit to its destination in either the on-premises network or the public internet. An architecture with two levels of hubs

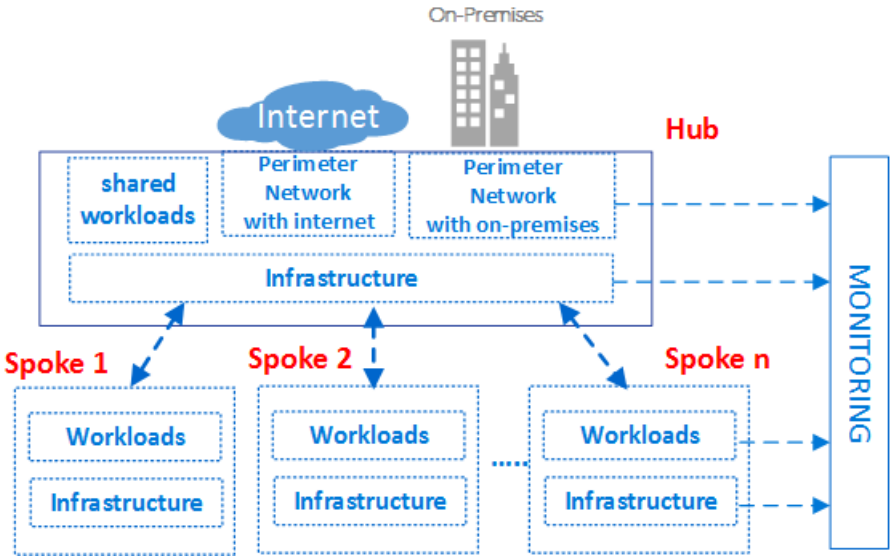
introduces complex routing that removes the benefits of a simple hub-spoke relationship.

Although Azure allows complex topologies, one of the core principles of the VDC concept is repeatability and simplicity. To minimize management effort, the simple hub-spoke design is the VDC reference architecture that we recommend.

Components

The virtual datacenter is made up of four basic component types: **Infrastructure**, **Perimeter Networks**, **Workloads**, and **Monitoring**.

Each component type consists of various Azure features and resources. Your VDC implementation is made up of instances of multiple components types and multiple variations of the same component type. For instance, you may have many different, logically separated workload instances that represent different applications. You use these different component types and instances to ultimately build the VDC.



The preceding high-level conceptual architecture of the VDC shows different component types used in different zones of the hub-spokes topology. The diagram shows infrastructure components in various parts of the architecture.

As good practice in general, access rights and privileges should be group-based. Dealing with groups rather than individual users eases maintenance of access policies, by providing a consistent way to manage it across teams, and aids in minimizing configuration errors. Assigning and removing users to and from appropriate groups helps keeping the privileges of a specific user up to date.

Each role group should have a unique prefix on their names. This prefix makes it easy to identify which group is associated with which workload. For example, a workload hosting an authentication service might have groups named **AuthServiceNetOps**, **AuthServiceSecOps**, **AuthServiceDevOps**, and **AuthServiceInfraOps**. Centralized roles, or roles not related to a specific service, might be prefaced with **Corp**. An example is **CorpNetOps**.

Many organizations use a variation of the following groups to provide a major breakdown of roles:

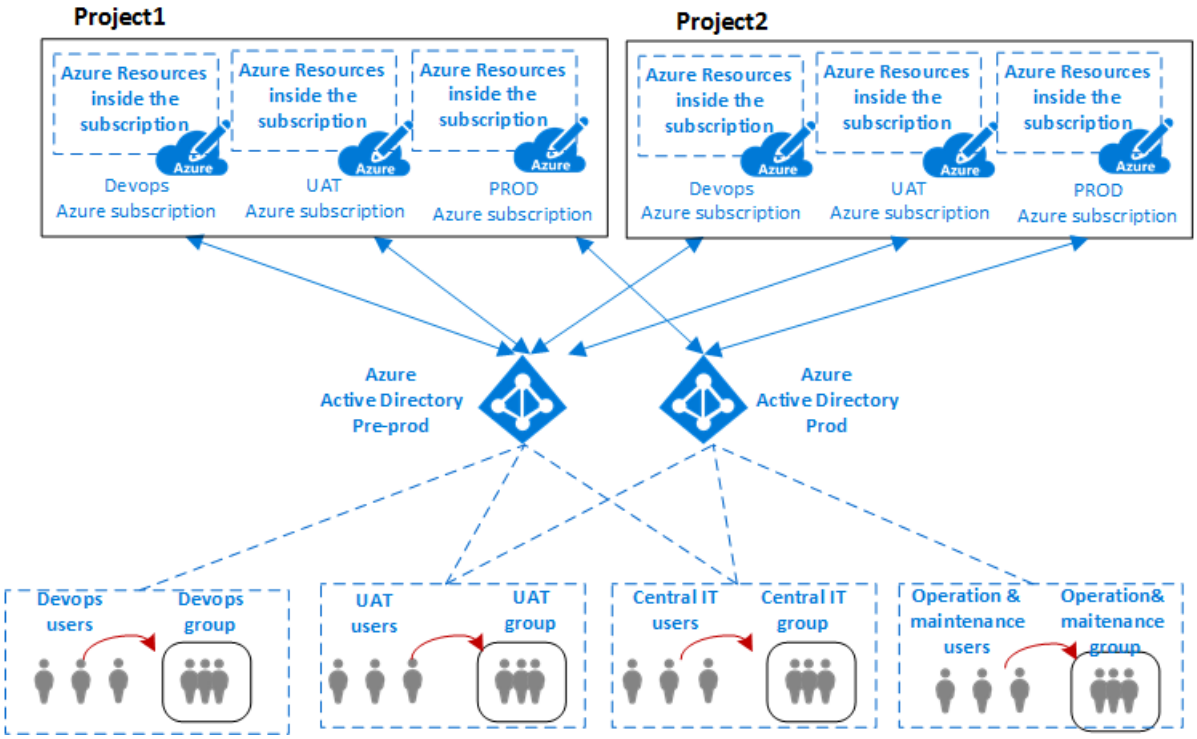
- The central IT team named **Corp** has the ownership rights to control infrastructure components. Examples are networking and security. The group needs to have the role of contributor on the subscription, control of the hub, and network contributor rights in the spokes. Large organizations frequently split up these management responsibilities between multiple teams. Examples are a network operations **CorpNetOps** group with exclusive focus on networking and a security operations **CorpSecOps** group responsible for the firewall and security policy. In this specific case, two different groups need to be created for assignment of these custom roles.
- The dev/test group named **AppDevOps** has the responsibility to deploy app or service workloads. This

group takes the role of virtual machine contributor for IaaS deployments or one or more PaaS contributor's roles. For more information, see Azure built-in roles. Optionally, the dev/test team might need visibility on security policies (network security groups) and routing policies (user-defined routes) inside the hub or a specific spoke. In addition to the role of contributor for workloads, this group would also need the role of network reader.

- The operation and maintenance group called **CorpInfraOps** or **AppInfraOps** has the responsibility of managing workloads in production. This group needs to be a subscription contributor on workloads in any production subscriptions. Some organizations might also evaluate if they need an additional escalation support team group with the role of subscription contributor in production and the central hub subscription. The additional group fixes potential configuration issues in the production environment.

The VDC is designed so that groups created for the central IT team, managing the hub, have corresponding groups at the workload level. In addition to managing hub resources only, the central IT team can control external access and top-level permissions on the subscription. Workload groups can also control resources and permissions of their virtual network independently from the central IT team.

The virtual datacenter is partitioned to securely host multiple projects across different lines of business. All projects require different isolated environments (dev, UAT, and production). Separate Azure subscriptions for each of these environments can provide natural isolation.



The preceding diagram shows the relationship between an organization's projects, users, and groups and the environments where the Azure components are deployed.

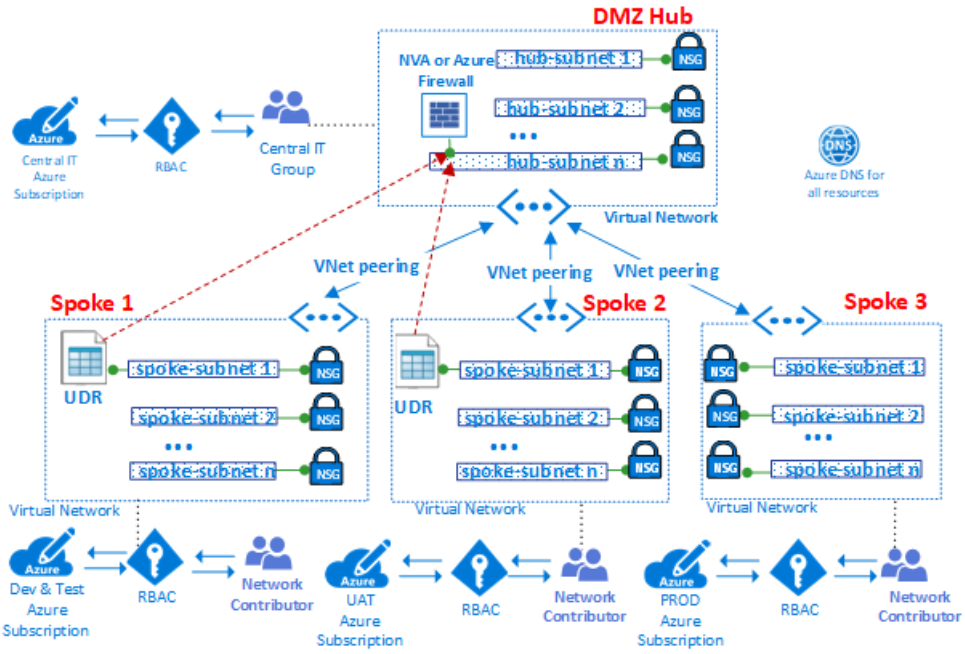
Typically in IT, an environment (or tier) is a system in which multiple applications are deployed and executed. Large enterprises use a development environment (where changes are made and tested) and a production environment (what end-users use). Those environments are separated, often with several staging environments in between them to allow phased deployment (rollout), testing, and rollback if problems arise. Deployment architectures vary significantly, but usually the basic process of starting at development (DEV) and ending at production (PROD) is still followed.

A common architecture for these types of multitier environments consists of DevOps for development and testing, UAT for staging, and production environments. Organizations can use single or multiple Azure AD tenants to define access and rights to these environments. The previous diagram shows a case where two different Azure AD tenants are used: one for DevOps and UAT, and the other exclusively for production.

The presence of different Azure AD tenants enforces the separation between environments. The same group of users, such as the central IT team, need to authenticate by using a different URI to access a different Azure AD tenant to modify the roles or permissions of either the DevOps or production environments of a project. The presence of different user authentications to access different environments reduces possible outages and other issues caused by human errors.

Component type: Infrastructure

This component type is where most of the supporting infrastructure resides. It's also where your centralized IT, security, and compliance teams spend most of their time.



Infrastructure components provide an interconnection for the different components of a VDC implementation, and are present in both the hub and the spokes. The responsibility for managing and maintaining the infrastructure components is typically assigned to the central IT team or security team.

One of the primary tasks of the IT infrastructure team is to guarantee the consistency of IP address schemas across the enterprise. The private IP address space assigned to a VDC implementation must be consistent and **not** overlapping with private IP addresses assigned on your on-premises networks.

While NAT on the on-premises edge routers or in Azure environments can avoid IP address conflicts, it adds complications to your infrastructure components. Simplicity of management is one of the key goals of the VDC, so using NAT to handle IP concerns, while a valid solution, is not a recommended solution.

Infrastructure components have the following functionality:

- Identity and directory services. Access to every resource type in Azure is controlled by an identity stored in a directory service. The directory service stores not only the list of users, but also the access rights to resources in a specific Azure subscription. These services can exist cloud-only, or they can be synchronized with on-premises identity stored in Active Directory.
- Virtual Network. Virtual networks are one of main components of the VDC, and enable you to create a traffic isolation boundary on the Azure platform. A virtual network is composed of a single or multiple virtual network segments, each with a specific IP network prefix (a subnet, either IPv4 or dual stack IPv4/IPv6). The virtual network defines an internal perimeter area where IaaS virtual machines and PaaS services can establish private communications. VMs (and PaaS services) in one virtual network can't communicate directly to VMs (and PaaS services) in a different virtual network, even if both virtual networks are created by the same customer, under the same subscription. Isolation is a critical property that ensures customer VMs and communication remains private within a virtual network. Where cross-network connectivity is desired, the following features describe how that can be accomplished.

- Virtual network peering. The fundamental feature used to create the infrastructure of a VDC is virtual network peering, which connects two virtual networks in the same region, either through the Azure datacenter network or using the Azure worldwide backbone across regions.
- Virtual Network service endpoints. Service endpoints extend your virtual network private address space to include your PaaS space. The endpoints also extend the identity of your virtual network to the Azure services over a direct connection. Endpoints allow you to secure your critical Azure service resources to only your virtual networks.
- Private Link. Azure Private Link enables you to access Azure PaaS Services (for example, Azure Storage, Azure Cosmos DB, and Azure SQL Database) and Azure hosted customer/partner services over a Private Endpoint in your virtual network. Traffic between your virtual network and the service traverses over the Microsoft backbone network, eliminating exposure from the public Internet. You can also create your own Private Link Service in your virtual network and deliver it privately to your customers. The setup and consumption experience using Azure Private Link is consistent across Azure PaaS, customer-owned, and shared partner services.
- User-defined routes. Traffic in a virtual network is routed by default based on the system routing table. A user-defined route is a custom routing table that network administrators can associate to one or more subnets to override the behavior of the system routing table and define a communication path within a virtual network. The presence of user-defined routes guarantees that egress traffic from the spoke transit through specific custom VMs or network virtual appliances and load balancers present in both the hub and the spokes.
- Network security groups. A network security group is a list of security rules that act as traffic filtering on IP sources, IP destinations, protocols, IP source ports, and IP destination ports (also called a Layer 4 five-tuple). The network security group can be applied to a subnet, a Virtual NIC associated with an Azure VM, or both. The network security groups are essential to implement a correct flow control in the hub and in the spokes. The level of security afforded by the network security group is a function of which ports you open, and for what purpose. Customers should apply additional per-VM filters with host-based firewalls such as iptables or the Windows Firewall.
- DNS. DNS provides name resolution for resources in a virtual datacenter. Azure provides DNS services for both public and private name resolution. Private zones provide name resolution both within a virtual network and across virtual networks. You can have private zones not only span across virtual networks in the same region, but also across regions and subscriptions. For public resolution, Azure DNS provides a hosting service for DNS domains, providing name resolution using Microsoft Azure infrastructure. By hosting your domains in Azure, you can manage your DNS records using the same credentials, APIs, tools, and billing as your other Azure services.
- Management group, subscription, and resource group management. A subscription defines a natural boundary to create multiple groups of resources in Azure. This separation can be for function, role segregation, or billing. Resources in a subscription are assembled together in logical containers known as resource groups. The resource group represents a logical group to organize the resources in a virtual datacenter. If your organization has many subscriptions, you may need a way to efficiently manage access, policies, and compliance for those subscriptions. Azure management groups provide a level of scope above subscriptions. You organize subscriptions into containers known as management groups and apply your governance conditions to the management groups. All subscriptions within a management group automatically inherit the conditions applied to the management group. To see these three features in a hierarchy view, see Organizing your resources in the Cloud Adoption Framework.
- Azure role-based access control (Azure RBAC). Azure RBAC can map organizational roles and rights to access specific Azure resources, allowing you to restrict users to only a certain subset of actions. If you're synchronizing Azure Active Directory with an on-premises Active Directory, you can use the same Active Directory groups in Azure that you use on-premises. With Azure RBAC, you can grant access by assigning the appropriate role to users, groups, and applications within the relevant scope. The scope of a role assignment can be an Azure subscription, a resource group, or a single resource. Azure RBAC allows inheritance of permissions. A role assigned at a parent scope also grants access to the children contained within it. Using

Azure RBAC, you can segregate duties and grant only the amount of access to users that they need to perform their jobs. For example, one employee can manage virtual machines in a subscription, while another can manage SQL Server databases in the same subscription.

Component Type: Perimeter Networks

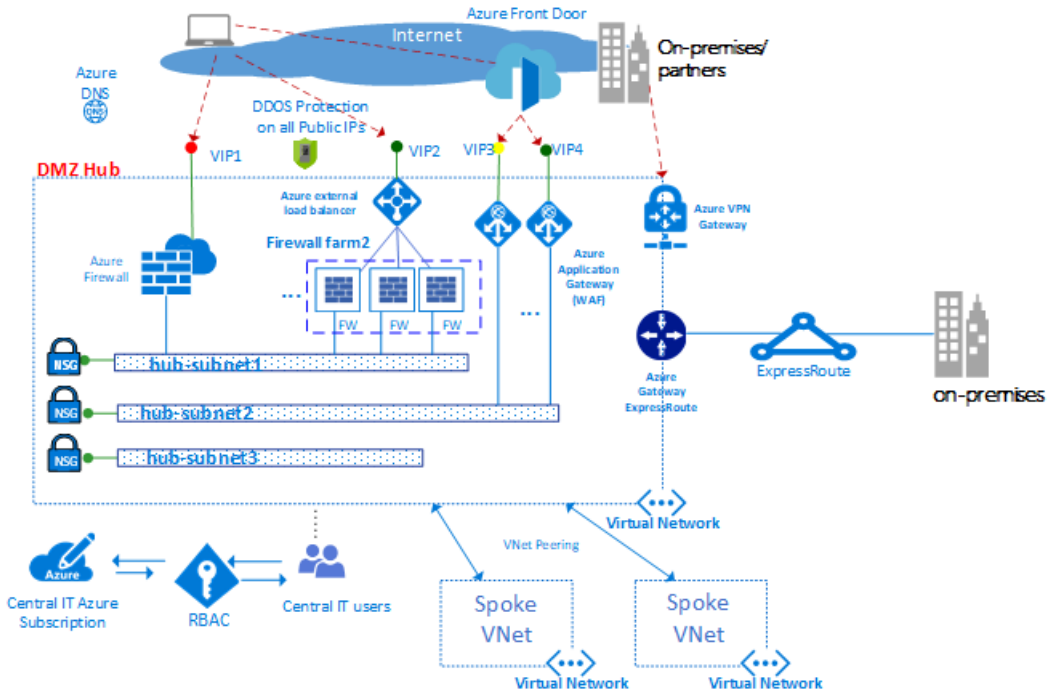
Components of a perimeter network (sometimes called a DMZ network) connect your on-premises or physical datacenter networks, along with any internet connectivity. The perimeter typically requires a significant time investment from your network and security teams.

Incoming packets should flow through the security appliances in the hub before reaching the back-end servers and services in the spokes. Examples include the firewall, IDS, and IPS. Before they leave the network, internet-bound packets from the workloads should also flow through the security appliances in the perimeter network. This flow enables policy enforcement, inspection, and auditing.

Perimeter network components include:

- Virtual networks, user-defined routes, and network security groups
- Network virtual appliances
- Azure Load Balancer
- Azure Application Gateway with web application firewall (WAF)
- Public IPs
- Azure Front Door with web application firewall (WAF)
- Azure Firewall and Azure Firewall Manager
- Standard DDoS Protection

Usually, the central IT team and security teams have responsibility for requirement definition and operation of the perimeter networks.



The preceding diagram shows the enforcement of two perimeters with access to the internet and an on-premises network, both resident in the DMZ hub. In the DMZ hub, the perimeter network to internet can scale up to support many lines of business, using multiple farms of Web Application Firewalls (WAFs) or Azure Firewalls. The hub also allows for on-premises connectivity via VPN or ExpressRoute as needed.

> **NOTE**
>
> In the preceding diagram, in the "DMZ Hub", many of the following features can be bundled together in an Azure Virtual WAN hub (such as virtual networks, user-defined routes, network security groups, VPN gateways, ExpressRoute gateways, Azure load balancers, Azure Firewalls, Firewall Manager, and DDOS). Using Azure Virtual WAN hubs can make the creation of the hub virtual network, and thus the VDC, much easier, since most of the engineering complexity is handled for you by Azure when you deploy an Azure Virtual WAN hub.

Virtual networks. The hub is typically built on a virtual network with multiple subnets to host the different types of services that filter and inspect traffic to or from the internet via Azure Firewall, NVAs, WAF, and Azure Application Gateway instances.

User-defined routes. Using user-defined routes, customers can deploy firewalls, IDS/IPS, and other virtual appliances, and route network traffic through these security appliances for security boundary policy enforcement, auditing, and inspection. User-defined routes can be created in both the hub and the spokes to guarantee that traffic transits through the specific custom VMs, Network Virtual Appliances, and load balancers used by a VDC implementation. To guarantee that traffic generated from virtual machines residing in the spoke transits to the correct virtual appliances, a user-defined route needs to be set in the subnets of the spoke by setting the front-end IP address of the internal load balancer as the next hop. The internal load balancer distributes the internal traffic to the virtual appliances (load balancer back-end pool).

Azure Firewall is a managed network security service that protects your Azure Virtual Network resources. It's a stateful managed firewall with high availability and cloud scalability. You can centrally create, enforce, and log application and network connectivity policies across subscriptions and virtual networks. Azure Firewall uses a static public IP address for your virtual network resources. It allows outside firewalls to identify traffic that originates from your virtual network. The service is fully integrated with Azure Monitor for logging and analytics.

If you use the Azure Virtual WAN topology, the Azure Firewall Manager is a security management service that provides central security policy and route management for cloud-based security perimeters. It works with Azure Virtual WAN hub, a Microsoft-managed resource that lets you easily create hub and spoke architectures. When security and routing policies are associated with such a hub, it's referred to as a secured virtual hub.

Network virtual appliances. In the hub, the perimeter network with access to the internet is normally managed through an Azure Firewall instance or a farm of firewalls or web application firewall (WAF).

Different lines of business commonly use many web applications, which tend to suffer from various vulnerabilities and potential exploits. Web application firewalls are a special type of product used to detect attacks against web applications, HTTP/HTTPS, in more depth than a generic firewall. Compared with tradition firewall technology, WAFs have a set of specific features to protect internal web servers from threats.

An Azure Firewall or NVA firewall both use a common administration plane, with a set of security rules to protect the workloads hosted in the spokes, and control access to on-premises networks. The Azure Firewall has scalability built in, whereas NVA firewalls can be manually scaled behind a load balancer. Generally, a firewall farm has less specialized software compared with a WAF, but has a broader application scope to filter and inspect any type of traffic in egress and ingress. If an NVA approach is used, they can be found and deployed from Azure Marketplace.

We recommend that you use one set of Azure Firewall instances, or NVAs, for traffic originating on the internet. Use another for traffic originating on-premises. Using only one set of firewalls for both is a security risk as it provides no security perimeter between the two sets of network traffic. Using separate firewall layers reduces the complexity of checking security rules and makes it clear which rules correspond to which incoming network request.

Azure Load Balancer offers a high availability Layer 4 (TCP/UDP) service, which can distribute incoming traffic

among service instances defined in a load-balanced set. Traffic sent to the load balancer from front-end endpoints (public IP endpoints or private IP endpoints) can be redistributed with or without address translation to a set of back-end IP address pool (such as network virtual appliances or virtual machines).

Azure Load Balancer can probe the health of the various server instances as well, and when an instance fails to respond to a probe, the load balancer stops sending traffic to the unhealthy instance. In a virtual datacenter, an external load balancer is deployed to the hub and the spokes. In the hub, the load balancer is used to efficiently route traffic across firewall instances, and in the spokes, load balancers are used to manage application traffic.

Azure Front Door (AFD) is Microsoft's highly available and scalable Web Application Acceleration Platform, Global HTTP Load Balancer, Application Protection, and Content Delivery Network. Running in more than 100 locations at the edge of Microsoft's Global Network, AFD enables you to build, operate, and scale out your dynamic web application and static content. AFD provides your application with world-class end-user performance, unified regional/stamp maintenance automation, BCDR automation, unified client/user information, caching, and service insights. The platform offers:

- Performance, reliability, and support service-level agreements (SLAs).
- Compliance certifications.
- Auditable security practices that are developed, operated, and natively supported by Azure.

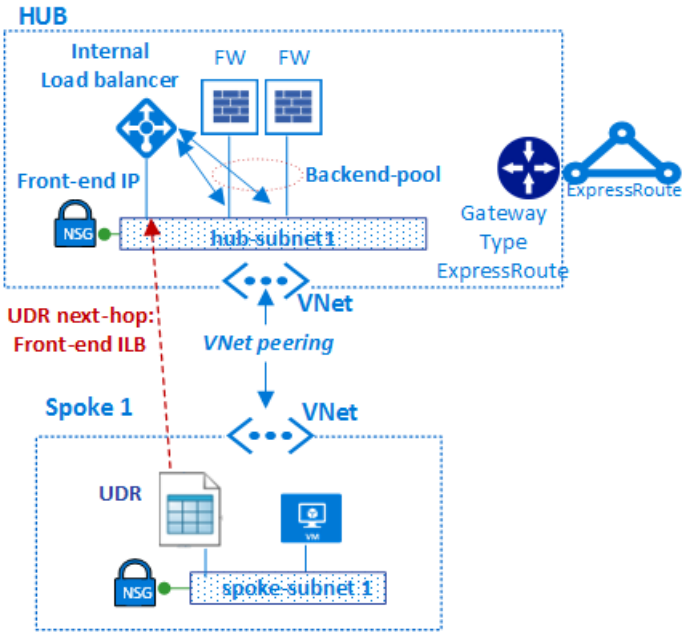
Azure Front Door also provides a web application firewall (WAF), which protects web applications from common vulnerabilities and exposures.

Azure Application Gateway is a dedicated virtual appliance providing a managed application delivery controller. It offers various Layer 7 load-balancing capabilities for your application. It allows you to optimize web farm performance by offloading CPU-intensive SSL termination to the application gateway. It also provides other Layer 7 routing capabilities, such as round-robin distribution of incoming traffic, cookie-based session affinity, URL-path-based routing, and the ability to host multiple websites behind a single application gateway. A web application firewall (WAF) is also provided as part of the application gateway WAF SKU. This SKU provides protection to web applications from common web vulnerabilities and exploits. Application Gateway can be configured as internet-facing gateway, internal-only gateway, or a combination of both.

Public IPs. With some Azure features, you can associate service endpoints to a public IP address so that your resource is accessible from the internet. This endpoint uses NAT to route traffic to the internal address and port on the virtual network in Azure. This path is the primary way for external traffic to pass into the virtual network. You can configure public IP addresses to determine which traffic is passed in and how and where it's translated onto the virtual network.

Azure DDoS Protection Standard provides additional mitigation capabilities over the Basic service tier that are tuned specifically to Azure Virtual Network resources. DDoS Protection Standard is simple to enable and requires no application changes. Protection policies are tuned through dedicated traffic monitoring and machine learning algorithms. Policies are applied to public IP addresses associated to resources deployed in virtual networks. Examples include Azure Load Balancer, Azure Application Gateway, and Azure Service Fabric instances. Near real-time, system-generated logs are available through Azure Monitor views during an attack and for history. Application layer protection can be added through the Azure Application Gateway web application firewall. Protection is provided for IPv4 and IPv6 Azure public IP addresses.

The hub and spoke topology uses virtual network peering and user-defined routes to route traffic properly.



In the diagram, the user-defined route ensures that traffic flows from the spoke to the firewall before passing to on-premises through the ExpressRoute gateway (if the firewall policy allows that flow).

Component type: Monitoring

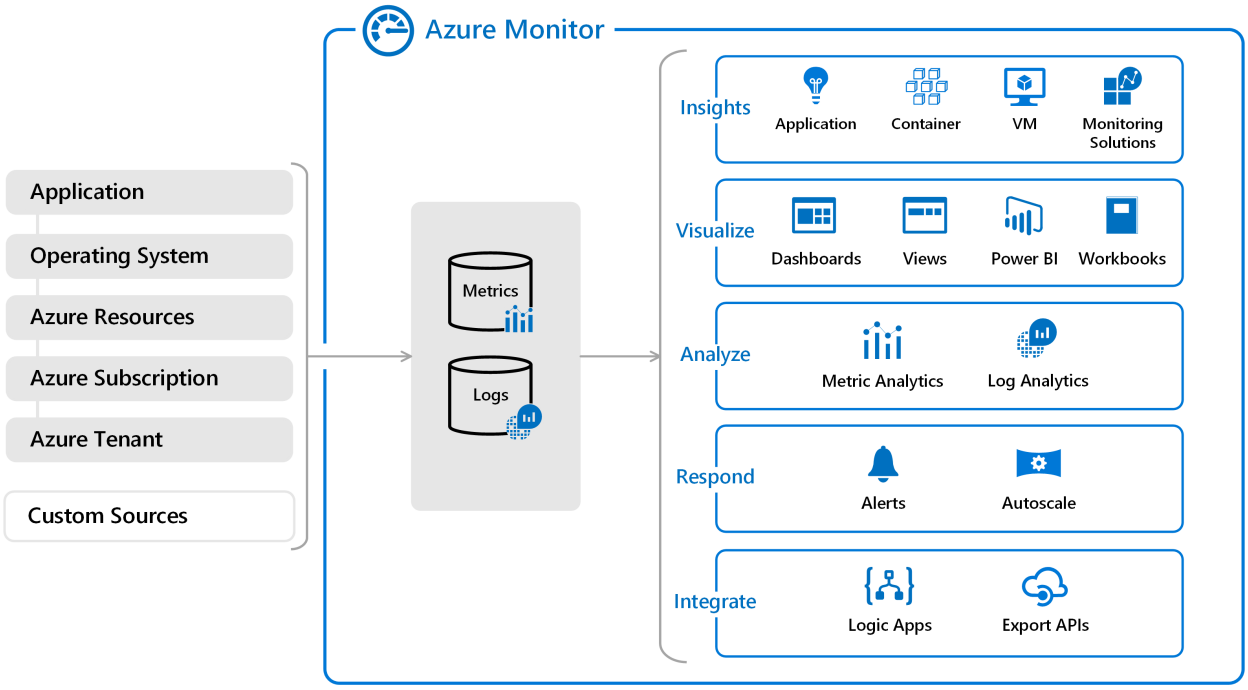
Monitoring components provide visibility and alerting from all the other component types. All teams should have access to monitoring for the components and services they have access to. If you have a centralized help desk or operations teams, they require integrated access to the data provided by these components.

Azure offers different types of logging and monitoring services to track the behavior of Azure-hosted resources. Governance and control of workloads in Azure is based not just on collecting log data but also on the ability to trigger actions based on specific reported events.

Azure Monitor. Azure includes multiple services that individually perform a specific role or task in the monitoring space. Together, these services deliver a comprehensive solution for collecting, analyzing, and acting on system-generated logs from your applications and the Azure resources that support them. They can also work to monitor critical on-premises resources in order to provide a hybrid monitoring environment. Understanding the tools and data that are available is the first step in developing a complete monitoring strategy for your applications.

There are two fundamental types of logs in Azure Monitor:

- Metrics are numerical values that describe some aspect of a system at a particular point in time. They are lightweight and capable of supporting near real-time scenarios. For many Azure resources, you'll see data collected by Azure Monitor right in their Overview page in the Azure portal. As an example, look at any virtual machine and you'll see several charts displaying performance metrics. Select any of the graphs to open the data in metrics explorer in the Azure portal, which allows you to chart the values of multiple metrics over time. You can view the charts interactively or pin them to a dashboard to view them with other visualizations.
- Logs contain different kinds of data organized into records with different sets of properties for each type. Events and traces are stored as logs along with performance data, which can all be combined for analysis. Log data collected by Azure Monitor can be analyzed with queries to quickly retrieve, consolidate, and analyze collected data. Logs are stored and queried from Log Analytics. You can create and test queries using Log Analytics in the Azure portal and then either directly analyze the data using these tools or save queries for use with visualizations or alert rules.



Azure Monitor can collect data from a variety of sources. You can think of monitoring data for your applications in tiers ranging from your application, any operating system, and the services it relies on, down to the Azure platform itself. Azure Monitor collects data from each of the following tiers:

- **Application monitoring data:** Data about the performance and functionality of the code you have written, regardless of its platform.
- **Guest OS monitoring data:** Data about the operating system on which your application is running. This OS could be running in Azure, another cloud, or on-premises.
- **Azure resource monitoring data:** Data about the operation of an Azure resource.
- **Azure subscription monitoring data:** Data about the operation and management of an Azure subscription, as well as data about the health and operation of Azure itself.
- **Azure tenant monitoring data:** Data about the operation of tenant-level Azure services, such as Azure Active Directory.
- **Custom sources:** Logs sent from on-premises sources can be included as well. Examples include on-premises server events or network device syslog output.

Monitoring data is only useful if it can increase your visibility into the operation of your computing environment. Azure Monitor includes several features and tools that provide valuable insights into your applications and other resources that they depend on. Monitoring solutions and features such as Application Insights and Azure Monitor for containers provide deep insights into different aspects of your application and specific Azure services.

Monitoring solutions in Azure Monitor are packaged sets of logic that provide insights for a particular application or service. They include logic for collecting monitoring data for the application or service, queries to analyze that data, and views for visualization. Monitoring solutions are available from Microsoft and partners to provide monitoring for various Azure services and other applications.

With all of this rich data collected, it's important to take proactive action on events happening in your environment where manual queries alone won't suffice. Alerts in Azure Monitor proactively notify you of critical conditions and potentially attempt to take corrective action. Alert rules based on metrics provide near real-time alerting based on numeric values, while rules based on logs allow for complex logic across data from multiple sources. Alert rules in Azure Monitor use action groups, which contain unique sets of recipients and actions that can be shared across multiple rules. Based on your requirements, action groups can perform such actions as using webhooks that cause alerts to start external actions or to integrate with your ITSM tools.

Azure Monitor also allows the creation of custom dashboards. Azure dashboards allow you to combine different

kinds of data, including both metrics and logs, into a single pane in the Azure portal. You can optionally share the dashboard with other Azure users. Elements throughout Azure Monitor can be added to an Azure dashboard in addition to the output of any log query or metrics chart. For example, you could create a dashboard that combines tiles that show a graph of metrics, a table of activity logs, a usage chart from Application Insights, and the output of a log query.

Finally, Azure Monitor data is a native source for Power BI. Power BI is a business analytics service that provides interactive visualizations across a variety of data sources and is an effective means of making data available to others within and outside your organization. You can configure Power BI to automatically import log data from Azure Monitor to take advantage of these additional visualizations.

Azure Network Watcher provides tools to monitor, diagnose, and view metrics and enable or disable logs for resources in a virtual network in Azure. It's a multifaceted service that allows the following functionalities and more:

- Monitor communication between a virtual machine and an endpoint.
- View resources in a virtual network and their relationships.
- Diagnose network traffic filtering problems to or from a VM.
- Diagnose network routing problems from a VM.
- Diagnose outbound connections from a VM.
- Capture packets to and from a VM.
- Diagnose problems with an virtual network gateway and connections.
- Determine relative latencies between Azure regions and internet service providers.
- View security rules for a network interface.
- View network metrics.
- Analyze traffic to or from a network security group.
- View diagnostic logs for network resources.

Component type: Workloads

Workload components are where your actual applications and services reside. It's where your application development teams spend most of their time.

The workload possibilities are endless. The following are just a few of the possible workload types:

Internal applications: Line-of-business applications are critical to enterprise operations. These applications have some common characteristics:

- **Interactive:** Data is entered, and results or reports are returned.
- **Data-driven:** Data intensive with frequent access to databases or other storage.
- **Integrated:** Offer integration with other systems within or outside the organization.

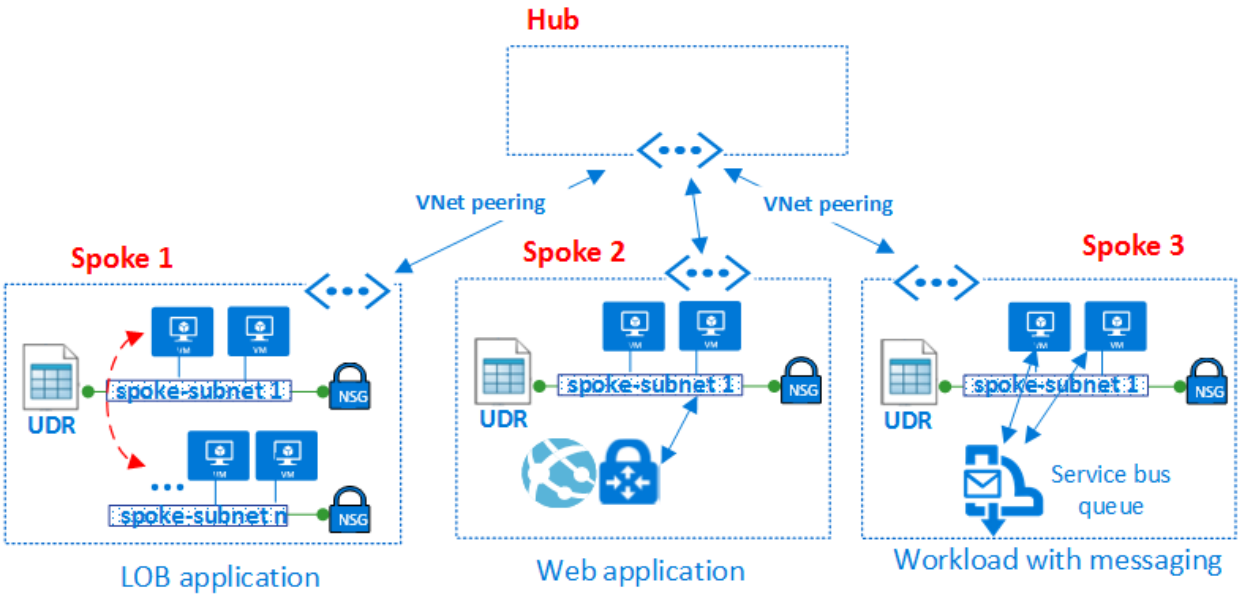
Customer-facing web sites (internet-facing or internally facing): Most internet applications are web sites. Azure can run a web site via either an IaaS virtual machine or an Azure Web Apps site (PaaS). Azure Web Apps integrates with virtual networks to deploy web apps in a spoke network zone. Internally facing web sites don't need to expose a public internet endpoint because the resources are accessible via private non-internet routable addresses from the private virtual network.

Big data analytics: When data needs to scale up to larger volumes, relational databases may not perform well under the extreme load or unstructured nature of the data. Azure HDInsight is a managed, full-spectrum, open-source analytics service in the cloud for enterprises. You can use open-source frameworks such as Hadoop, Apache Spark, Apache Hive, LLAP, Apache Kafka, Apache Storm, and R. HDInsight supports deploying into a location-based virtual network, can be deployed to a cluster in a spoke of the virtual datacenter.

Events and Messaging: Azure Event Hubs is a big data streaming platform and event ingestion service. It can receive and process millions of events per second. It provides low latency and configurable time retention,

enabling you to ingest massive amounts of data into Azure and read it from multiple applications. A single stream can support both real-time and batch-based pipelines.

You can implement a highly reliable cloud messaging service between applications and services through Azure Service Bus. It offers asynchronous brokered messaging between client and server, structured first-in-first-out (FIFO) messaging, and publishes and subscribe capabilities.



These examples barely scratch the surface of the types of workloads you can create in Azure—everything from a basic Web and SQL app to the latest in IoT, big data, machine learning, AI, and so much more.

Highly availability: multiple virtual datacenters

So far, this article has focused on the design of a single VDC, describing the basic components and architectures that contribute to resiliency. Azure features such as Azure load balancer, NVAs, availability zones, availability sets, scale sets, and other capabilities that help you include solid SLA levels into your production services.

However, because a virtual datacenter is typically implemented within a single region, it might be vulnerable to outages that affect the entire region. Customers that require high availability must protect the services through deployments of the same project in two or more VDC implementations deployed to different regions.

In addition to SLA concerns, several common scenarios benefit from running multiple virtual datacenters:

- Regional or global presence of your end users or partners.
- Disaster recovery requirements.
- A mechanism to divert traffic between datacenters for load or performance.

Regional/global presence

Azure datacenters exist in many regions worldwide. When selecting multiple Azure datacenters, consider two related factors: geographical distances and latency. To optimize user experience, evaluate the distance between each virtual datacenter as well as the distance from each virtual datacenter to the end users.

An Azure region that hosts your virtual datacenter must conform with regulatory requirements of any legal jurisdiction under which your organization operates.

Disaster recovery

The design of a disaster recovery plan depends on the types of workloads and the ability to synchronize state of those workloads between different VDC implementations. Ideally, most customers desire a fast fail-over mechanism, and this requirement may need application data synchronization between deployments running in multiple VDC implementations. However, when designing disaster recovery plans, it's important to consider that most applications are sensitive to the latency that can be caused by this data synchronization.

Synchronization and heartbeat monitoring of applications in different VDC implementations requires them to communicate over the network. Multiple VDC implementations in different regions can be connected through:

- Hub-to-hub communication built into Azure Virtual WAN hubs across regions in the same Virtual WAN.
- Virtual network peering to connect hubs across regions.
- ExpressRoute private peering, when the hubs in each VDC implementation are connected to the same ExpressRoute circuit.
- Multiple ExpressRoute circuits connected via your corporate backbone, and your multiple VDC implementations connected to the ExpressRoute circuits.
- Site-to-Site VPN connections between the hub zone of your VDC implementations in each Azure region.

Typically, Virtual WAN hubs, virtual network peering, or ExpressRoute connections are preferred for network connectivity, due to the higher bandwidth and consistent latency levels when passing through the Microsoft backbone.

Run network qualification tests to verify the latency and bandwidth of these connections, and decide whether synchronous or asynchronous data replication is appropriate based on the result. It's also important to weigh these results in view of the optimal recovery time objective (RTO).

Disaster recovery: diverting traffic from one region to another

Both Azure Traffic Manager and Azure Front Door periodically check the service health of listening endpoints in different VDC implementations and, if those endpoints fail, route automatically to the next closest VDC. Traffic Manager uses real-time user measurements and DNS to route users to the closest (or next closest during failure). Azure Front Door is a reverse proxy at over 100 Microsoft backbone edge sites, using anycast to route users to the closest listening endpoint.

Summary

A virtual datacenter approach to datacenter migration creates a scalable architecture that optimizes Azure resource use, lowers costs, and simplifies system governance. The virtual datacenter is typical based on hub and spoke network topologies (using either virtual network peering or Virtual WAN hubs). Common shared services provided in the hub, and specific applications and workloads are deployed in the spokes. The virtual datacenter also matches the structure of company roles, where different departments such as Central IT, DevOps, and Operations and Maintenance all work together while performing their specific roles. The virtual datacenter supports migrating existing on-premises workloads to Azure, but also provides many advantages to cloud-native deployments.

References

Learn more about the Azure capabilities discussed in this document.

Network features

Azure Virtual Networks
Network Security Groups
Service Endpoints
Private Link
User-Defined Routes
Network Virtual Appliances
Public IP Addresses
Azure DNS

Load balancing

Azure Front Door
Azure Load Balancer (Layer 4)
Application Gateway (Layer 7)
Azure Traffic Manager

Connectivity

Virtual Network Peering
Virtual Private Network
Virtual WAN
ExpressRoute
ExpressRoute Direct

Identity

Azure Active Directory
Multi-Factor Authentication
Azure role-based access control
Azure built-in roles

Monitoring

Network Watcher
Azure Monitor
Log Analytics

Best practices

Management Group
Subscription Management
Resource Group Management
Azure Subscription Limits

Security

Azure Firewall
Firewall Manager
Application Gateway WAF
Front Door WAF
Azure DDoS

Other Azure services

Azure Storage
Azure SQL
Azure Web Apps
Azure Cosmos DB
HDInsight
Event Hubs
Service Bus
Azure IoT
Azure Machine Learning

Next steps

- Learn more about virtual network peering, the core technology of hub and spoke topologies.
- Implement Azure Active Directory to use Azure role-based access control.
- Develop a subscription and resource management model using Azure role-based access control that fits the structure, requirements, and policies of your organization. The most important activity is planning. Analyze how reorganizations, mergers, new product lines, and other considerations will affect your initial models to ensure you can scale to meet future needs and growth.

Perimeter networks

8/25/2021 • 6 minutes to read • Edit Online

Perimeter networks enable secure connectivity between your cloud networks and your on-premises or physical datacenter networks, along with any connectivity to and from the internet. A perimeter network is sometimes called a demilitarized zone or DMZ.

For perimeter networks to be effective, incoming packets must flow through security appliances hosted in secure subnets before reaching back-end servers. Examples include the firewall, intrusion detection systems, and intrusion prevention systems. Before they leave the network, internet-bound packets from workloads should also flow through the security appliances in the perimeter network. The purposes of this flow are policy enforcement, inspection, and auditing.

Perimeter networks make use of the following Azure features and services:

- Virtual networks, user-defined routes, and network security groups
- Network virtual appliances (NVAs)
- Azure Load Balancer
- Azure Application Gateway and Web Application Firewall (WAF)
- Public IPs
- Azure Front Door with Web Application Firewall
- Azure Firewall

> **NOTE**
>
> Azure reference architectures provide example templates that you can use to implement your own perimeter networks:
>
> - Implement a perimeter network between Azure and your on-premises datacenter
> - Implement a perimeter network between Azure and the internet

Usually, your central IT team and security teams are responsible for defining requirements for operating your perimeter networks.

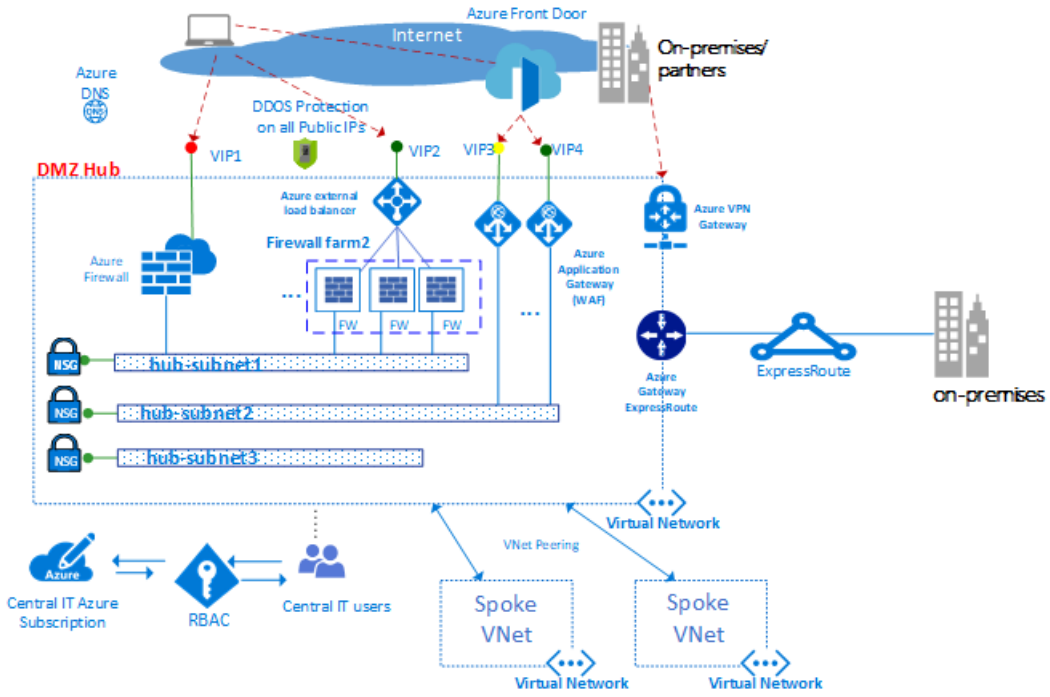


Figure 1: Example of a hub and spoke network topology.

The diagram above shows an example hub and spoke network topology that implements enforcement of two perimeters with access to the internet and an on-premises network. Both perimeters reside in the DMZ hub. In the DMZ hub, the perimeter network to the internet can scale up to support many lines of business via multiple farms of WAFs and Azure Firewall instances that help protect the spoke virtual networks. The hub also allows for connectivity via VPN or Azure ExpressRoute as needed.

Virtual networks

Perimeter networks are typically built using a virtual network with multiple subnets to host the different types of services that filter and inspect traffic to or from the internet via NVAs, WAFs, and Azure Application Gateway instances.

User-defined routes

By using user-defined routes, customers can deploy firewalls, intrusion detection systems, intrusion prevention systems, and other virtual appliances. Customers can then route network traffic through these security appliances for security boundary policy enforcement, auditing, and inspection. User-defined routes can be created to guarantee that traffic passes through the specified custom VMs, NVAs, and load balancers.

In a hub and spoke network example, guaranteeing that traffic generated by virtual machines that reside in the spoke passes through the correct virtual appliances in the hub requires a user-defined route defined in the subnets of the spoke. This route sets the front-end IP address of the internal load balancer as the next hop. The internal load balancer distributes the internal traffic to the virtual appliances (load balancer back-end pool).

Azure Firewall

Azure Firewall is a managed cloud-based service that helps protect your Azure Virtual Network resources. It's a fully stateful managed firewall with built-in high availability and unrestricted cloud scalability. You can centrally create, enforce, and log application and network connectivity policies across subscriptions and virtual networks.

Azure Firewall uses a static public IP address for your virtual network resources. It allows outside firewalls to identify traffic that originates from your virtual network. The service interoperates with Azure Monitor for logging and analytics.

Network virtual appliances

Perimeter networks with access to the internet are typically managed through an Azure Firewall instance or a farm of firewalls or web application firewalls.

Different lines of business commonly use many web applications. These applications tend to suffer from various vulnerabilities and potential exploits. A Web Application Firewall detects attacks against web applications (HTTP/S) in more depth than a generic firewall. Compared with tradition firewall technology, web application firewalls have a set of specific features to help protect internal web servers from threats.

An Azure Firewall instance and a [network virtual appliance][nva] firewall use a common administration plane with a set of security rules to help protect the workloads hosted in the spokes and control access to on-premises networks. Azure Firewall has built-in scalability, whereas NVA firewalls can be manually scaled behind a load balancer.

A firewall farm typically has less specialized software compared with a WAF, but it has a broader application scope to filter and inspect any type of traffic in egress and ingress. If you use an NVA approach, you can find and deploy the software from the Azure Marketplace.

Use one set of Azure Firewall instances (or NVAs) for traffic that originates on the internet and another set for traffic that originates on-premises. Using only one set of firewalls for both is a security risk because it provides no security perimeter between the two sets of network traffic. Using separate firewall layers reduces the complexity of checking security rules and makes clear which rules correspond to which incoming network requests.

Azure Load Balancer

Azure Load Balancer offers a high-availability Layer 4 (TCP/UDP) service, which can distribute incoming traffic among service instances defined in a load-balanced set. Traffic sent to the load balancer from front-end endpoints (public IP endpoints or private IP endpoints) can be redistributed with or without address translation to a pool of back-end IP addresses (such as NVAs or VMs).

Azure Load Balancer can also probe the health of the various server instances. When an instance fails to respond to a probe, the load balancer stops sending traffic to the unhealthy instance.

As an example of using a hub and spoke network topology, you can deploy an external load balancer to both the hub and the spokes. In the hub, the load balancer efficiently routes traffic to services in the spokes. In the spokes, load balancers manage application traffic.

Azure Front Door

Azure Front Door is Microsoft's highly available and scalable web application acceleration platform and global HTTPS load balancer. You can use Azure Front Door to build, operate, and scale out your dynamic web application and static content. It runs in more than 100 locations at the edge of Microsoft's global network.

Azure Front Door provides your application with unified regional/stamp maintenance automation, BCDR automation, unified client/user information, caching, and service insights. The platform offers performance, reliability, and support SLAs. It also offers compliance certifications and auditable security practices that are developed, operated, and supported natively by Azure.

Azure Application Gateway

Azure Application Gateway is a dedicated virtual appliance that provides a managed application delivery controller. It offers various Layer 7 load-balancing capabilities for your application.

Azure Application Gateway allows you to optimize web farm productivity by offloading CPU-intensive SSL

termination to the application gateway. It also provides other Layer 7 routing capabilities, including round-robin distribution of incoming traffic, cookie-based session affinity, URL path-based routing, and the ability to host multiple websites behind a single application gateway.

The Azure Application Gateway WAF SKU includes a Web Application Firewall. This SKU provides protection to web applications from common web vulnerabilities and exploits. You can configure Azure Application Gateway as an internet-facing gateway, an internal-only gateway, or a combination of both.

Public IPs

With some Azure features, you can associate service endpoints to a public IP address so that your resource can be accessed from the internet. This endpoint uses network address translation (NAT) to route traffic to the internal address and port on the Azure virtual network. This path is the primary way for external traffic to pass into the virtual network. You can configure public IP addresses to determine what traffic is passed in, and how and where it's translated onto the virtual network.

Azure DDoS Protection Standard

Azure DDoS Protection Standard provides extra mitigation capabilities over the basic service tier that are tuned specifically to Azure Virtual Network resources. DDoS protection standard is simple to enable and requires no application changes.

You can tune protection policies through dedicated traffic monitoring and machine-learning algorithms. Policies are applied to public IP addresses associated to resources deployed in virtual networks. Examples include Azure Load Balancer, Application Gateway, and Service Fabric instances.

Real-time telemetry is available through Azure Monitor views both during an attack and for historical purposes. You can add application-layer protection by using the Web Application Firewall in Azure Application Gateway. Protection is provided for IPv4 Azure public IP addresses.

Hub-and-spoke network topology

8/25/2021 • 5 minutes to read • Edit Online

Hub and spoke is a networking model for efficiently managing common communication or security requirements. It also helps avoid Azure subscription limitations. This model addresses the following concerns:

- **Saving on costs and efficient management:** Centralizing services that can be shared by multiple workloads, such as network virtual appliances (NVAs) and DNS servers, in a single location allows IT to minimize redundant resources and management effort.
- **Overcoming subscription limits:** Large cloud-based workloads might require using more resources than are allowed in a single Azure subscription. Peering workload virtual networks from different subscriptions to a central hub can overcome these limits. For more information, see Azure subscription limits.
- **A separation of concerns:** You can deploy individual workloads between central IT teams and workload teams.

Smaller cloud estates might not benefit from the added structure and capabilities that this model offers. But larger cloud adoption efforts should consider implementing a hub-and-spoke networking architecture if they have any of the concerns listed previously.

> **NOTE**
>
> The Azure reference architectures site contains example templates that you can use as the basis for implementing your own hub-and-spoke networks:
>
> - Implement a hub-and-spoke network topology in Azure
> - Implement a hub-and-spoke network topology with shared services in Azure

Overview

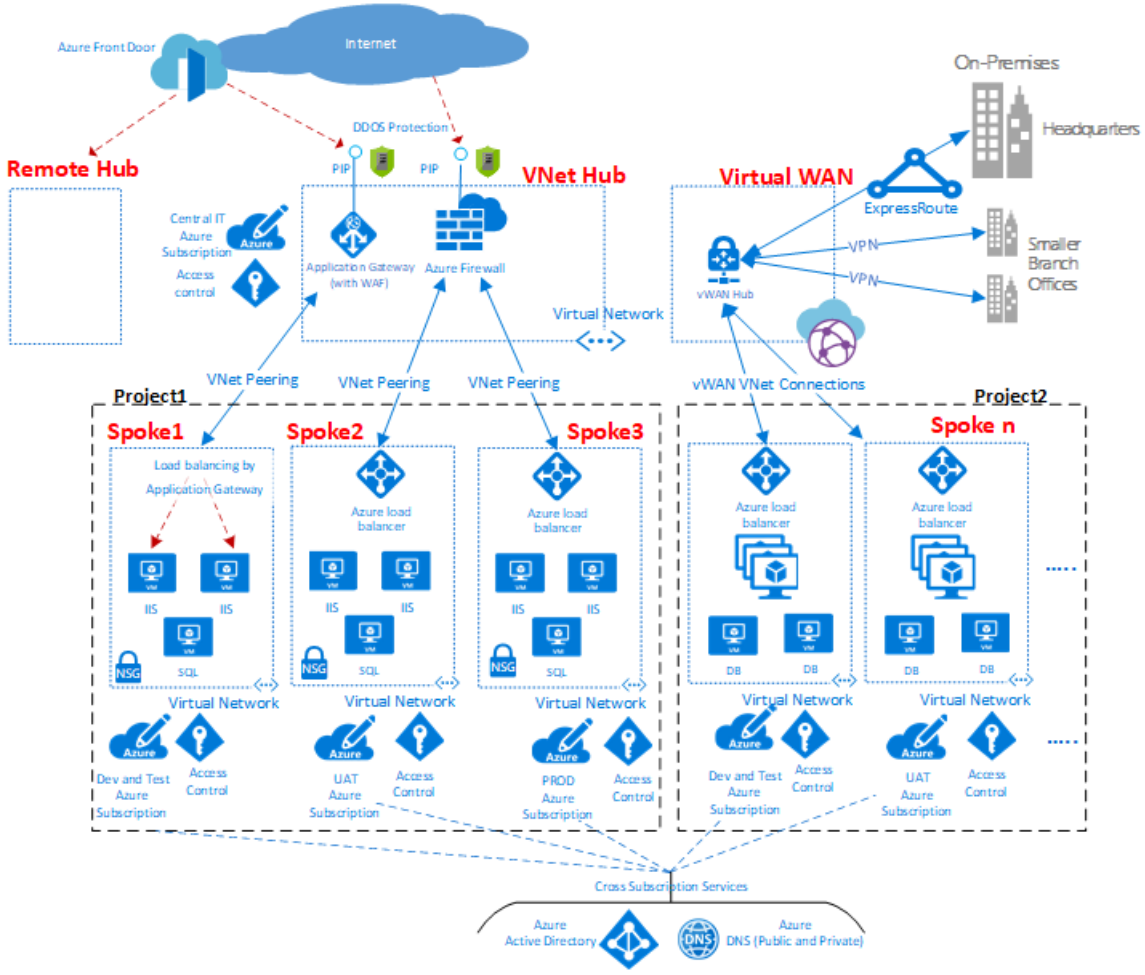


Figure 1: An example of a hub-and-spoke network topology.

As shown in the diagram, Azure supports two types of hub-and-spoke design. It supports communication, shared resources, and centralized security policy (labeled as `VNet hub` in the diagram), or a design based on Azure Virtual WAN (labeled as `Virtual WAN` in the diagram) for large-scale branch-to-branch and branch-to-Azure communications.

A hub is a central network zone that controls and inspects ingress or egress traffic between zones: internet, on-premises, and spokes. The hub-and-spoke topology gives your IT department an effective way to enforce security policies in a central location. It also reduces the potential for misconfiguration and exposure.

The hub often contains the common service components that the spokes consume. The following examples are common central services:

- The Windows Server Active Directory infrastructure is required to authenticate third-party users who access untrusted networks before they access workloads in the spoke. It includes the related Active Directory Federation Services (AD FS).
- A DNS service resolves naming the workload in the spokes to access resources on-premises and on the internet if Azure DNS isn't used.
- A public key infrastructure implements single sign-on for workloads.
- TCP and UDP traffic flow is controlled between the spoke network zones and the internet.
- Flow is controlled between the spokes and on-premises.
- Flow is controlled control between one spoke and another, if needed.

You can minimize redundancy, simplify management, and reduce overall cost by using the shared hub infrastructure to support multiple spokes.

The role of each spoke can be to host different types of workloads. The spokes also provide a modular approach for repeatable deployments of the same workloads. Examples include dev/test, user acceptance testing, staging,

and production.

The spokes can also segregate and enable different groups within your organization. An example is Azure DevOps groups. Inside a spoke, it's possible to deploy a basic workload or complex multitier workloads with traffic control between the tiers.

In many cases the Application Gateway shown in the diagram above can live in spoke with the application it's serving for better management and scale, however corporate policy may dictate the Application Gateway be placed in the hub for centralized management and segregation of duty.

Subscription limits and multiple hubs

In Azure, every type of component is deployed in an Azure subscription. The isolation of Azure components in different Azure subscriptions can satisfy the requirements of different lines of business, such as setting up differentiated levels of access and authorization.

A single hub-and-spoke implementation can scale up to a large number of spokes, but as with every IT system, there are platform limits. The hub deployment is bound to a specific Azure subscription, which has restrictions and limits. One example is a maximum number of virtual network peerings. For more information, see Azure subscription and service limits.

In cases where limits might be an issue, you can scale up the architecture further by extending the model from a single hub-and-spoke to a cluster of hubs and spokes. You can connect multiple hubs in one or more Azure regions by using virtual network peering, Azure ExpressRoute, Azure Virtual WAN, or a site-to-site VPN.

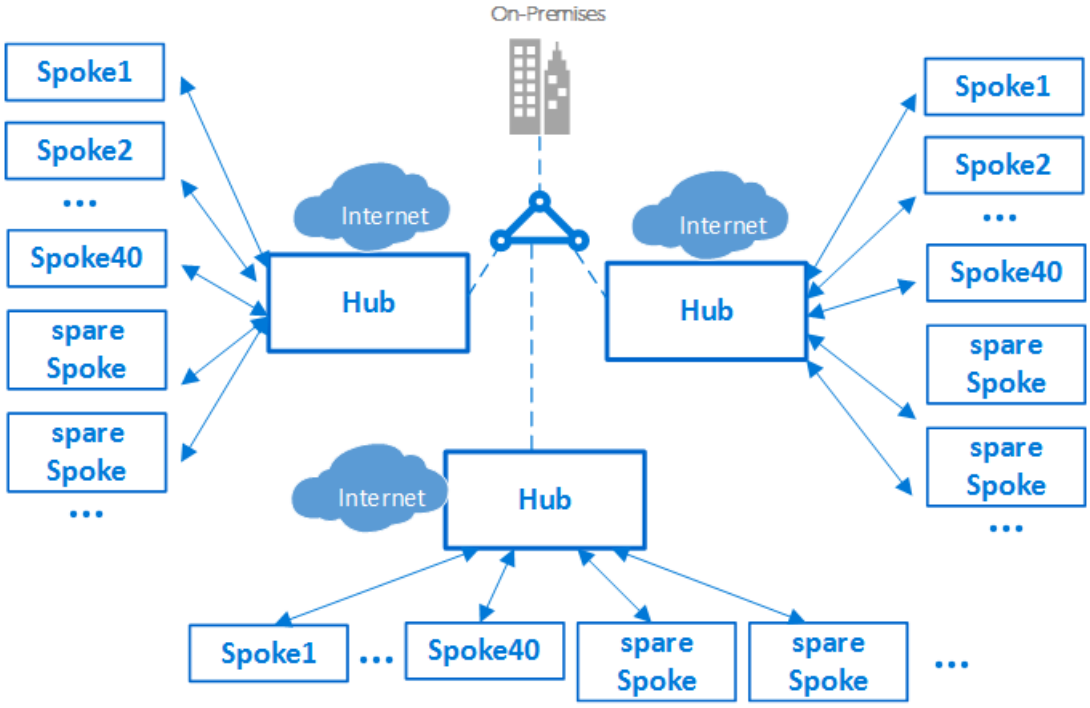


Figure 2: A cluster of hubs and spokes.

The introduction of multiple hubs increases the cost and management overhead of the system. This is only justified by scalability, system limits, or redundancy and regional replication for user performance or disaster recovery. In scenarios that require multiple hubs, all the hubs should strive to offer the same set of services for operational ease.

Interconnection between spokes

It's possible to implement complex multitier workloads in a single spoke. You can implement multitier configurations by using subnets (one for every tier) in the same virtual network and by using network security

groups to filter the flows.

An architect might want to deploy a multitier workload across multiple virtual networks. With virtual network peering, spokes can connect to other spokes in the same hub or in different hubs.

A typical example of this scenario is the case where application processing servers are in one spoke or virtual network. The database deploys in a different spoke or virtual network. In this case, it's easy to interconnect the spokes with virtual network peering and avoid transiting through the hub. The solution is to perform a careful architecture and security review to ensure that bypassing the hub doesn't bypass important security or auditing points that might exist only in the hub.

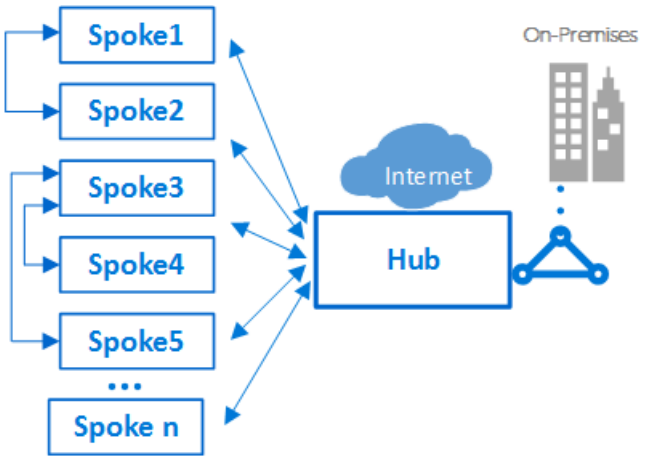


Figure 3: An example of spokes connecting to each other and a hub.

Spokes can also be interconnected to a spoke that acts as a hub. This approach creates a two-level hierarchy: the spoke in the higher level (level 0) becomes the hub of lower spokes (level 1) of the hierarchy. The spokes of a hub-and-spoke implementation are required to forward the traffic to the central hub so that the traffic can transit to its destination in either the on-premises network or the public internet. An architecture with two levels of hubs introduces complex routing that removes the benefits of a simple hub-and-spoke relationship.

Organize and set up Azure Machine Learning environments

7/14/2021 • 10 minutes to read • Edit Online

When planning an Azure Machine Learning deployment for an enterprise environment, there are some common decision points that affect how you create the workspace:

- **Team structure:** The way your Data Science teams are organized and collaborate on projects given use case and data segregation, or cost management requirements.
- **Environments:** The environments used as part of your development and release workflow to segregate development from production.
- **Region:** The location of your data and the audience you need to serve your Machine Learning solution to.

Team structure and workspace setup

The workspace is the top-level resource in Azure Machine Learning. It stores the artifacts produced when working with Machine Learning and the managed compute and pointers to attached and associated resources. From a manageability standpoint, the workspace as an Azure Resource Manager resource allows for Azure role-based access control (Azure RBAC), management by Policy, and can be used as a unit for cost reporting.

Organizations typically choose one or a combination of the following solution patterns to follow manageability requirements.

Workspace per team: Choose to use one workspace for each team when all members of a team require the same level of access to data and experimentation assets. For example, an organization with three machine learning teams might create three workspaces, one for each team.

The benefit of using one workspace per team is that all Machine Learning artifacts for the team's projects are stored in one place. Productivity increases can be realized because team members can easily access, explore, and reuse experimentation results. Organizing your workspaces by team reduces your Azure footprint and simplifies cost management by team. Because the number of experimentation assets can grow quickly, you can keep your artifacts organized by following naming and tagging conventions. For recommendations about how to name resources, see Develop your naming and tagging strategy for Azure resources.

A consideration for this approach is each team member must have similar data access level permissions. Granular RBAC and access control lists (ACL) for data sources and experimentation assets are limited within a workspace. You can't have use case data segregation requirements.

Workspace per project: Choose to use one workspace for each project if you require segregation of data and experimentation assets by project, or have cost reporting and budgeting requirements at a project level. For example, an organization with four machine learning teams that each runs three projects, might create 12 workspace instances.

The benefit of using one workspace per project is that costs can be managed at the project level. Teams typically create a dedicated resource group for Azure Machine Learning and associated resources for similar reasons. When you work with external contributors, for example, a project-centered workspace simplifies collaboration on a project because external users only need to be granted access to the project resources, not the team resources.

A consideration with this approach is the isolation of experimentation results and assets. The discovery and reuse of the assets might be more difficult because of assets being spread across multiple workspace instances.

Single Workspace: Choose to use one workspace for non-team or non-project related work, or when costs can't be directly associated to a specific unit of billing, for example with R&D.

The benefit of this setup is the cost of individual, non-project related work can be decoupled from project-related costs. When you set up a single workspace for all users to do their individual work, you reduce your Azure footprint.

A consideration for this approach is the workspace might become cluttered quickly when many Machine Learning practitioners share the same instance. Users might require UI-based filtering of assets to effectively find their resources. You can create shared Machine Learning workspaces for each business division to mitigate scale concerns or to segment budgets.

Environments and workspace setup

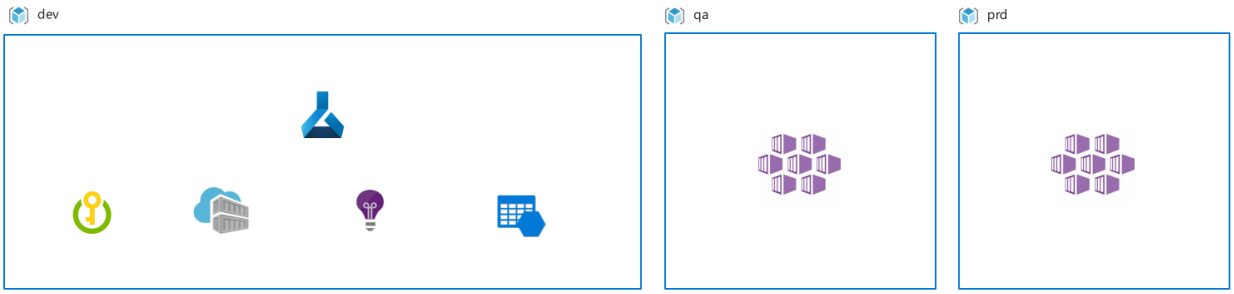
An environment is a collection of resources that deployments target based on their stage in the application lifecycle. Common examples of environment names are Dev, Test, QA, Staging, and Production.

The development process in your organization affects requirements for environment usage. Your environment affects the setup of Azure Machine Learning and associated resources, for example attached compute. For example, data availability might put constraints on the manageability of having a Machine Learning instance available for each environment. The following solution patterns are common:

Single environment workspace deployment: When you choose a single environment workspace deployment, Azure Machine Learning is deployed to one environment. This setup is common for research-centered scenarios, where there is no need to release Machine Learning artifacts based on their lifecycle stage, across environments. Another scenario where this setup makes sense is when only inferencing services, and not Machine Learning pipelines, are deployed across environments.

The benefit of a research-centered setup is a smaller Azure footprint and minimal management overhead. This way of working implies no need to have an Azure Machine Learning workspace deployed in each environment.

A consideration for this approach is a single environment deployment is subject to data availability. Caution is required with the Datastore set up. If you set up extensive access, for example, writer access on production data sources, you might unintentionally harm data quality. If you bring work to production in the same environment where development is done, the same RBAC restrictions apply for both the development work and the production work. This setup might make both environments too rigid or too flexible.



Multiple environment workspace deployment: When you choose a multiple environment workspace deployment, a workspace instance is deployed for each environment. A common scenario for this setup is a regulated workplace with a clear separation of duties between environments, and for users who have resource access to those environments.

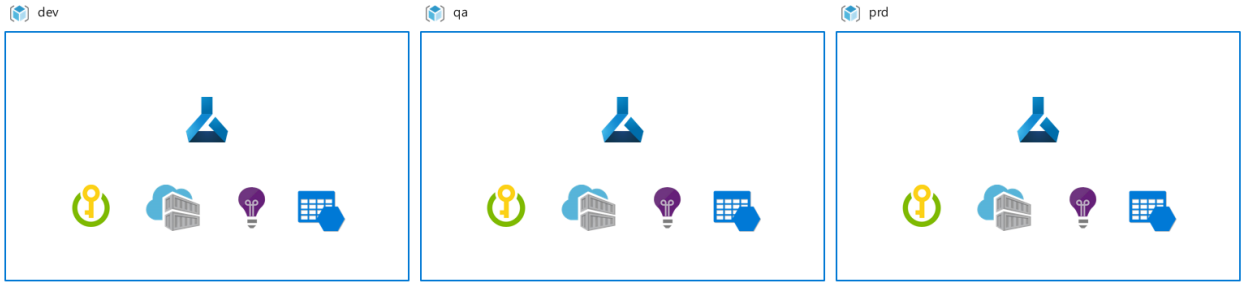
The benefits of this setup are:

- Staged rollout of Machine Learning workflows and artifacts. For example, models across environments,

with the potential of enhancing agility and reducing time-to-deployment.

- Enhanced security and control of resources because you have the ability to assign more access restrictions in downstream environments.
- Training scenarios on production data in non-development environments because you can give a select group of users access.

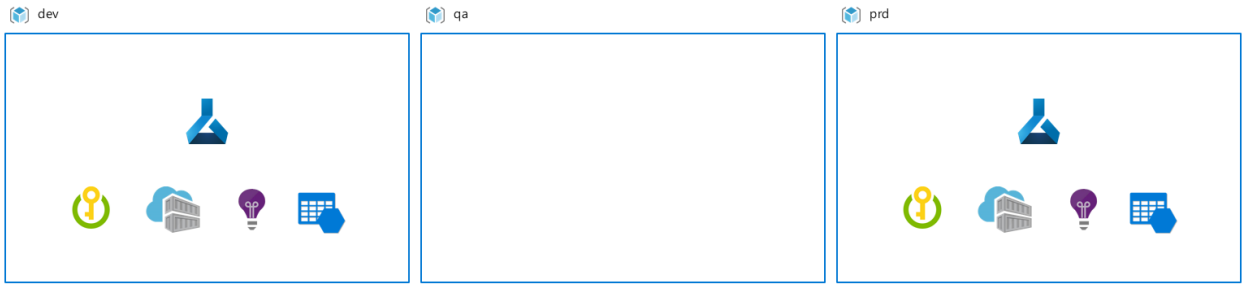
A consideration for this approach is you are at risk for more management and process overhead since this setup requires a fine-grained development and rollout process for Machine Learning artifacts across workspace instances. Additionally, data management and engineering effort might be required to make production data available for training in the development environment. Access management is required for you to give a team access to resolve and investigate incidents in production. And finally, Azure DevOps and Machine Learning engineering expertise is needed on your team to implement automation workflows.



One environment with limited data access, one with production data access: When you choose this setup, Azure Machine Learning is deployed to two environments: one with limited data access and one with production data access. This setup is common if you need to segregate development and production environments. For example, you might be working under organizational constraints to make production data available in any environment, or you might want to segregate development work from production work without duplicating data more than required due to the high cost of maintenance.

The benefit of this setup is the clear separation of duties and access between development and production environments. Another benefit is lower resource management overhead when compared to a multi-environment deployment scenario.

A consideration for this approach a defined development and rollout process for Machine Learning artifacts across workspaces is required. Another consideration is data management and engineering effort might be required to make production data available for training in a development environment. However, it might require relatively less effort than a multi-environment workspace deployment.



Regions and resource setup

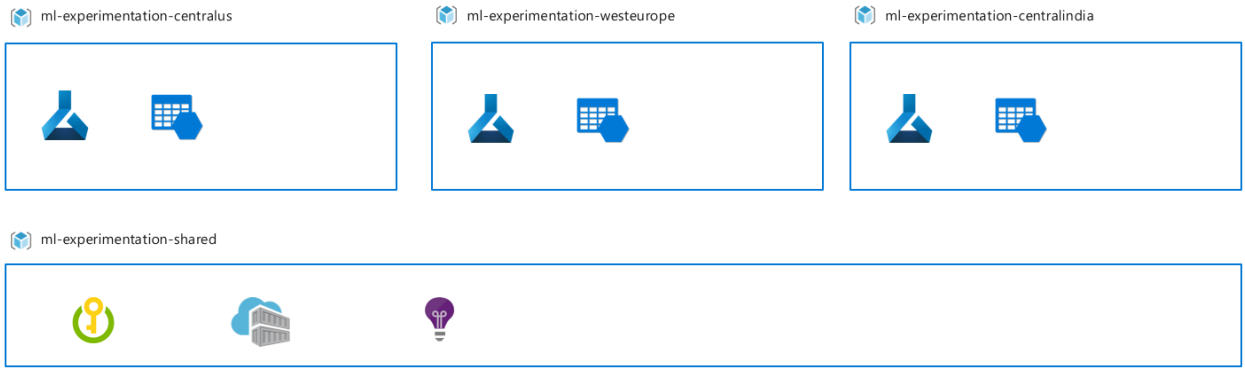
The location of your resources, data, or users might require you to create Azure Machine Learning workspace instances and associated resources in multiple Azure regions. For example, one project might span its resources across the West Europe and East US Azure regions for performance, cost, and compliance reasons. The following scenarios are common:

Regional training: The machine learning training jobs run in the same Azure region as where the data is located. In this setup, a Machine Learning workspace is deployed to each Azure region where data is located. It's

a common scenario when you are acting under compliance, or when you have data movement constraints across regions.

The benefit of this setup is experimentation can be done in the data center where the data is located with the least network latency. A consideration for this approach is when a Machine Learning pipeline is run across multiple workspace instances, it adds more management complexity. It becomes challenging to compare experimentation results across instances and adds overhead to quota and compute management.

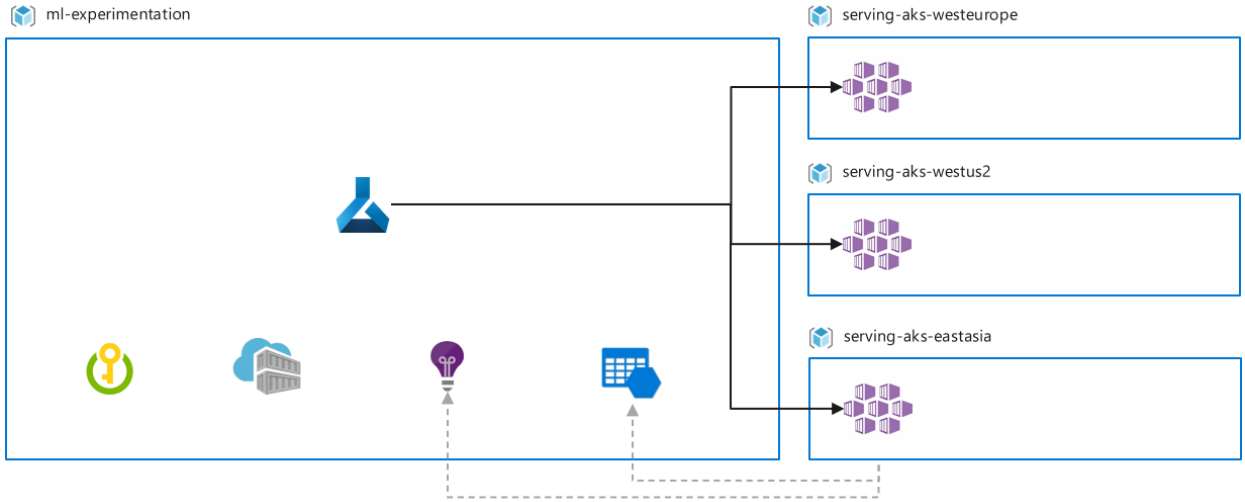
If you want to attach storage across regions, but use compute from one region, Azure Machine Learning supports the scenario of attaching storage accounts in a region rather than the workspace. Metadata, for example metrics, will be stored in the workspace region.



Regional serving: Machine Learning services are deployed close to where the target audience lives. For example, if target users are in Australia and the main storage and experimentation region is West Europe, deploy the Machine Learning workspace for experimentation in West Europe, and deploy an AKS cluster for inference endpoint deployment in Australia.

The benefits of this setup are the opportunity for inferencing in the data center where new data is ingested, minimizing latency and data movement, and compliance with local regulations.

A consideration for this approach is a multi-region setup provides several advantages, it also adds more overhead on quota and compute management. When there is a requirement for batch inferencing, regional serving might require a multi-workspace deployment. Data collected through inferencing endpoints might require to be transferred across regions for retraining scenarios.

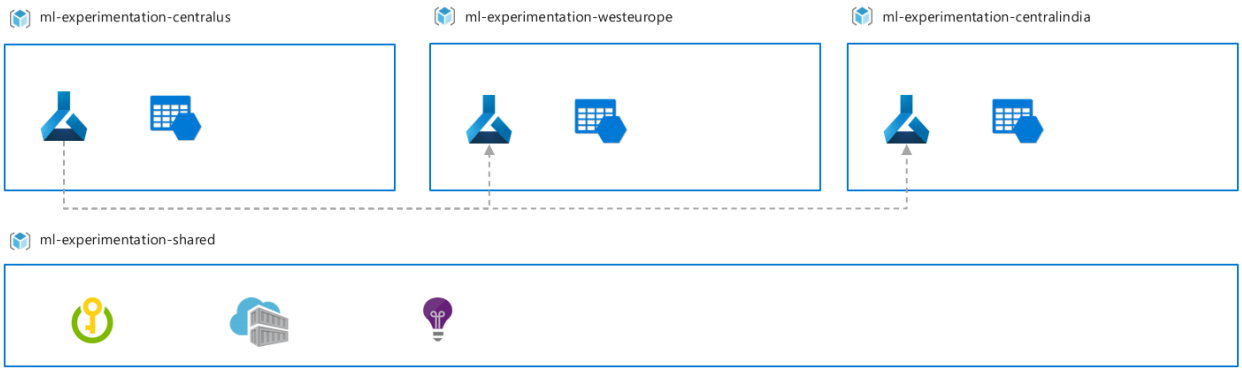


Regional fine-tuning: A base model is trained on an initial dataset, for example, public data or data from all regions, and is later fine-tuned with a regional dataset. The regional dataset might only exist in a particular region because of compliance or data movement constraints. For example, base model training might be done in a workspace in region A, while fine tuning might be done in a workspace in region B.

The benefit of this setup is experimentation is available in compliance with the data center where the data

resides, and still takes advantage of base model training on a larger dataset in an earlier pipeline stage.

A consideration is this approach provides the ability for complex experimentation pipelines, however it might create more challenges. For example, comparing experiment results across regions and more adding more overhead to quota and compute management.



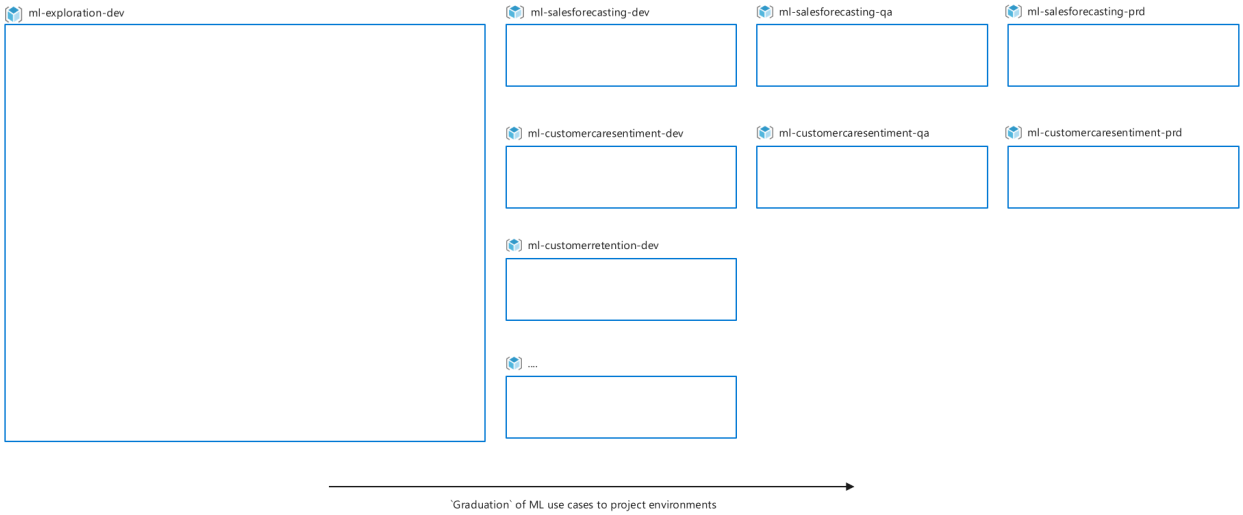
Reference implementation

To illustrate the deployment of Azure Machine Learning in a larger setting, this section outlines how the organization 'Contoso' has set up Azure Machine Learning given their organizational constraints, reporting, and budgeting requirements:

- Contoso creates resource groups on a solution basis for cost management and reporting reasons.
- IT administrators only create resource groups and resources for funded solutions to meet budget requirements.
- Because of the exploratory and uncertain nature of Data Science, users need a place to experiment and work for use case and data exploration. Often, exploratory work can't be directly associated to a particular use case, and can be associated only to an R&D budget. Contoso wants to fund some Machine Learning resources centrally that anyone can use for exploration purposes.
- Once a Machine Learning use case proves to be successful in the exploratory environment, teams can request resource groups. For example, Dev, QA, and Prod for iterative experimentation project work, and access to production data sources can be set up.
- Data segregation and compliance requirements don't allow live production data to exist in development environments.
- Different RBAC requirements exist for various user groups by IT policy per environment, for example access is more restrictive in production.
- All data, experimentation, and inferencing is done in a single Azure region.

To adhere to the above requirements, Contoso has set up their resources in the following way:

- Azure Machine Learning workspaces and resource groups are scoped per project to follow budgeting and use case segregation requirements.
- A multiple-environment setup for Azure Machine Learning and associated resources to address cost management, RBAC, and data access requirements.
- A single resource group and Machine Learning workspace that is dedicated for exploration.
- Azure Active Directory groups that are different per user role and environment, for example operations that a data scientist can do in a production environment are different than in the development environment, and access levels might differ per solution.
- All resources are created in a single Azure region.



Next steps

To learn about best practices on Machine Learning DevOps with Azure Machine Learning, see Machine learning DevOps guide.

Machine learning DevOps guide

To learn about considerations when managing budgets, quota and cost with Azure Machine Learning, see Manage budgets, costs, and quota for Azure Machine Learning at organizational scale.

Manage budgets, costs, and quota for Azure Machine Learning at organizational scale

Manage budgets, costs, and quota for Azure Machine Learning at organizational scale

7/15/2021 • 16 minutes to read • Edit Online

When you manage compute costs incurred from Azure Machine Learning, at an organization scale with many workloads, many teams, and users, there are numerous management and optimization challenges to work through.

In this article, we present best practices to optimize costs, manage budgets, and share quota with Azure Machine Learning. It reflects the experience and lessons learned from running machine learning teams internally at Microsoft and while partnering with our customers. You'll learn how to:

- Optimize compute resources to meet workload requirements.
- Drive the best use of a team's budget.
- Plan, manage and share budgets, cost, and quota at enterprise-scale.

Optimize compute to meet workload requirements

When you start a new machine learning project, exploratory work might be needed to get a good picture of compute requirements. This section provides recommendations on how you can determine the right virtual machine (VM) SKU choice for training, for inferencing, or as a workstation to work from.

Determine the compute size for training

Hardware requirements for your training workload might vary from project to project. To meet these requirements, Azure Machine Learning compute offers various types of VMs:

- **General purpose:** Balanced CPU to memory ratio.
- **Memory optimized:** High memory to CPU ratio.
- **Compute optimized:** High CPU to memory ratio.
- **High performance compute:** Deliver leadership-class performance, scalability, and cost efficiency for various real-world HPC workloads.
- **Instances with GPUs:** Specialized virtual machines targeted for heavy graphic rendering and video editing, as well as model training and inferencing (ND) with deep learning.

You might not know yet what your compute requirements are. In this scenario, we recommend starting with either of the following cost effective default options. These options are for lightweight testing and for training workloads.

TYPE	VIRTUAL MACHINE SIZE	SPECS
CPU	Standard_DS3_v2	4 cores, 14 gigabytes (GB) RAM, 28-GB storage
GPU	Standard_NC6	6 cores, 56 gigabytes (GB) RAM, 380-GB storage, NVIDIA Tesla K80 GPU

To get the best VM size for your scenario, it might consist of trial and error. Here are several aspects to consider.

- If you need a CPU:
  - Use a memory optimized VM if you're training on large datasets.

    - Use a compute optimized VM if you're doing real-time inferencing or other latency sensitive tasks.
    - Use a VM with more cores and RAM in order to speed up training times.
- If you need a GPU:
    - Use a VM with a faster GPU in order to speed up training times. Here's the list of GPU-enabled VM families in order of speed:
        - NC-series (Nvidia Tesla K80)
        - NV-series (Nvidia Tesla M80)
        - ND-series (Nvidia Tesla P40)
        - NCv2-series (Nvidia Tesla P100)
        - NCv3-series (Nvidia Tesla V100)
        - NDv2-series (Nvidia Tesla V100 with NVLink connections)
        - NCasT4_v3-series (Nvidia Tesla T4)
    - If you're doing distributed training, use VM sizes that have multiple GPUs.
    - If you're doing distributed training on multiple nodes, use GPUs that have NVLink connections.

While you select the VM type and SKU that best fits your workload, evaluate comparable VM SKUs as a trade-off between CPU and GPU performance and pricing. From a cost management perspective, a job might run reasonably well on several SKUs.

Certain GPUs such as the NC family, particularly NC_Promo SKUs, have similar abilities to other GPUs such as low latency and ability to manage multiple computing workloads in parallel. They're available at discounted prices compared to some of the other GPUs. Considerately selecting VM SKUs to the workload might save cost significantly in the end.

A reminder on the importance for utilization is to sign up for a greater number of GPUs doesn't necessarily execute with faster results. Instead, make sure the GPUs are fully utilized. For example, double check the need for NVIDIA CUDA. While it might be required for high-performance GPU execution, your job might not take a dependency on it.

Determine the compute size for inference

Compute requirements for inference scenarios differ from training scenarios. Available options differ based on whether your scenario demands offline inference in batch or requires online inference in real time.

For real-time inference scenarios consider the following suggestions:

- Use profiling capabilities on your model with Azure Machine Learning to determine how much CPU and memory you need to allocate for the model when deploying it as a web service.
- If you're doing real-time inference but don't need high availability, deploy to Azure Container Instances (no SKU selection).
- If you're doing real-time inference but need high availability, deploy to Azure Kubernetes Service.
    - If you're using traditional machine learning models and receive < 10 queries/second, start with a CPU SKU. F-series SKUs often work well.
    - If you're using deep learning models and receive > 10 queries/second, try a NVIDIA GPU SKU (NCasT4_v3 often works well) with Triton.

For batch inference scenarios consider the following suggestions:

- When you use Azure Machine Learning pipelines for batch inferencing, follow the guidance in Determine the compute size for training to choose your initial VM size.
- Optimize cost and performance by scaling horizontally. One of the key methods of optimizing cost and performance is by parallelizing the workload with the help of parallel run step in Azure Machine Learning. This pipeline step allows you to use many smaller nodes to execute the task in parallel, which allows you to scale horizontally. There's an overhead for parallelization though. Depending on the workload and the degree

of parallelism that can be achieved, a parallel run step may or may not be an option.

Determine the size for compute instance

For interactive development, Azure Machine Learning's compute instance is recommended. The compute instance (CI) offering brings single node compute that's bound to a single user and can be used as a cloud workstation.

Some organizations disallow the use of production data on local workstations, have enforced restrictions to the workstation environment, or restrict the installation of packages and dependencies in the corporate IT environment. A compute instance can be used as a workstation to overcome the limitation. It offers a secure environment with production data access, and runs on images that come with popular packages and tools for data science pre-installed.

When compute instance is running, user is billed for VM compute, Standard Load Balancer (included lb/outbound rules, and data processed), OS disk (Premium SSD managed P10 disk), temp disk (the temp disk type depends on the VM size chosen), and public IP address. To save costs, we recommend users consider:

- Start and stop the compute instance when it's not in use.
- Work with a sample of your data on a compute instance and scale out to compute clusters to work with your full set of data
- Submit experimentation jobs in *local* compute target mode on the compute instance while developing or testing, or when you switch to shared compute capacity when you submit jobs at full scale. For example, many epochs, full set of data, and hyperparameter search.

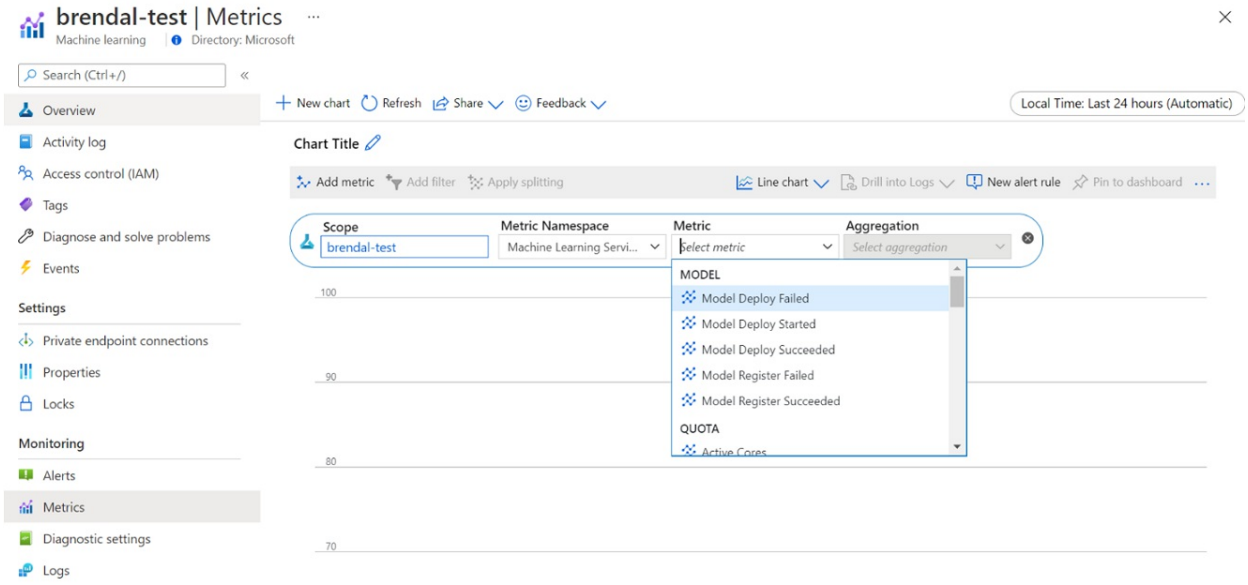
If you stop the compute instance, it stops billing for VM compute hours, temp disk, and Standard Load Balancer data processed costs. Note user still pays for OS disk and Standard Load Balancer included lb/outbound rules even when compute instance is stopped. Any data saved on OS disk is persisted through stop and restarts.

Tune the chosen VM size by monitoring compute utilization

You can view information on your Azure Machine Learning compute usage and utilization via Azure Monitor. You can view details on model deployment and registration, quota details such as active and idle nodes, run details such as canceled and completed runs, and compute utilization for GPU and CPU utilization.

Based on the insights from the monitoring details, you can better plan or adjust your resource usage across the team. For example, if you notice many idle nodes over the past week, you can work with the corresponding workspace owners to update the compute cluster configuration to prevent this extra cost. Benefits of analyzing the utilization patterns can help with forecasting costs and budget improvements.

You can access these metrics directly from the Azure portal. Go to your Azure Machine Learning workspace, and select *Metrics* under the monitoring section on the left panel. Then, you can select details on what you would like to view, such as metrics, aggregation, and time period. For more information, see Monitor Azure Machine Learning documentation page.



Switch between local, single-node, and multi-node cloud compute while you develop

There are varying compute and tooling requirements throughout the machine learning lifecycle. Azure Machine Learning can be interfaced with through an SDK and CLI interface from practically any preferred workstation configuration to meet these requirements.

To save costs and work productively, it's recommended to:

- Clone your experimentation code base locally by using Git and submit jobs to cloud compute using the Azure Machine Learning SDK or CLI.
- If your dataset is large, consider managing a sample of your data on your local workstation, while keeping the full dataset on cloud storage.
- Parameterize your experimentation code base so that you can configure your jobs to run with a varying number of epochs or on datasets of different sizes.
- Don't hard code the folder path of your dataset. You can then easily reuse the same code base with different datasets, and under local and cloud execution context.
- Bootstrap your experimentation jobs in *local* compute target mode while you develop or test, or when you switch to a shared compute cluster capacity when you submit jobs at full scale.
- If your dataset is large, work with a sample of data on your local or compute instance workstation, while scaling to cloud compute in Azure Machine Learning to work with your full set of data.
- When your jobs take a long time to execute, consider optimizing your code base for distributed training to allow for scaling out horizontally.
- Design your distributed training workloads for node elasticity, to allow flexible use of single-node and multi-node compute, and ease usage of compute that can be preempted.

Combine compute types using Azure Machine Learning pipelines

When you orchestrate your machine learning workflows, you can define a pipeline with multiple steps. Each step in the pipeline can run on its own compute type. This allows you to optimize performance and cost to meet varying compute requirements across the machine learning lifecycle.

Drive the best use of a team's budget

While budget allocation decisions might be out of the span of control of an individual team, a team is typically empowered to use their allocated budget to their best needs. By trading off job priority versus performance and cost wisely, a team can achieve higher cluster utilization, lower overall cost, and use a larger number of compute hours from the same budget. This can result in enhanced team productivity.

Optimize the costs of shared compute resources

The key to optimize costs of shared compute resources is to ensure that they're being used to their full capacity. Here are some tips to optimize your shared resource costs:

- When you use compute instances, only turn them on when you have code to execute. Shut them down when they aren't being used.
- When you use compute clusters, set the minimum node count to 0 and the maximum node count to a number that is evaluated based on your budget constraints. Use the Azure pricing calculator to calculate the cost of full utilization of one VM node of your chosen VM SKU. Autoscaling will scale down all the compute nodes when there's no one using it. It will only scale up to the number of nodes you have budget for. You can configure autoscaling to scale down all the compute nodes.
- Monitor your resource utilizations such as CPU utilization and GPU utilization when training models. If the resources aren't being fully used, modify your code to better use resources or scale down to smaller or cheaper VM sizes.
- Evaluate whether you can create shared compute resources for your team to avoid computing inefficiencies caused by cluster scaling operations.
- Optimize compute cluster autoscaling timeout policies based on usage metrics.
- Use workspace quotas to control the amount of compute resources that individual workspaces have access to.

Introduce scheduling priority by creating clusters for multiple VM SKUs

Acting under quota and budget constraints, a team must trade off timely execution of jobs versus cost, to ensure important jobs run timely and a budget is used in the best way possible.

To support best compute utilization, teams are recommended to create clusters of various sizes and with *low priority* and *dedicated* VM priorities. Low-priority computes make use of surplus capacity in Azure and hence come with discounted rates. On the downside, these machines can be preempted anytime a higher priority ask comes in.

Using the clusters of varying size and priority, a notion of scheduling priority can be introduced. For example, when experimental and production jobs compete for the same *NC* GPU-quota, a production job might have preference to run over the experimental job. In that case, run the production job on the dedicated compute cluster, and the experimental job on the low priority compute cluster. When quota falls short, the experimental job will be preempted in favor of the production job.

Next to VM priority, consider running jobs on various VM SKUs. It might be that a job takes longer to execute on a VM instance with a P40 GPU than on a V100 GPU. However, since V100 VM instances might be occupied or quota fully used, the time to completion on the P40 might still be faster from a job throughput perspective. You might also consider running jobs with lower priority on less performant and cheaper VM instances from a cost management perspective.

Early-terminate a run when training doesn't converge

When you continuously experiment to improve a model against its baseline, you might be executing various experiment runs, each with slightly different configurations. For one run, you might tweak the input datasets. For another run, you might make a hyperparameter change. Not all changes might be as effective as the other. You detect early that a change didn't have the intended affect on the quality of your model training. To detect if training does not converge, monitor training progress during a run. For example, by logging performance metrics after each training epoch. Consider early terminating the job to free up resources and budget for another trial.

Plan, manage and share budgets, cost, and quota

As an organization grows its number of machine learning use cases and teams, it requires an increased operating maturity from IT and finance as well as coordination between individual machine learning teams to ensure efficient operations. Company-scale capacity and quota management become important to address

scarceness of compute resources and overcome management overhead.

This section discusses best practices for planning, managing, and sharing budgets, cost, and quota at enterprise-scale. It's based on learnings from managing many GPU training resources for machine learning internally at Microsoft.

Understanding resource spend with Azure Machine Learning

One of the biggest challenges as an administrator for planning compute needs is starting new with no historical information as a baseline estimate. On a practical sense, most projects will start from a small budget as a first step.

To understand where the budget is going, it's critical to know where Azure Machine Learning costs come from:

- Azure Machine Learning only charges for compute infrastructure used and doesn't add a surcharge on compute costs.
- When an Azure Machine Learning workspace is created, there are also a few other resources created to enable Azure Machine Learning: Key Vault, Application Insights, Azure Storage, and Azure Container Registry. These resources are used in Azure Machine Learning and you'll pay for these resources.
- There are costs associated with managed compute such as training clusters, compute instances, and managed inferencing endpoints. With these managed compute resources, there are the following infrastructure costs to account for: virtual machines, virtual network, load balancer, bandwidth, and storage.

Govern and restrict compute usage by policy

When you manage an Azure environment with many workloads, it can be a challenge to keep the overview on resource spend. Azure Policy can help control and govern resource spend, by restricting particular usage patterns across the Azure environment.

In specific for Azure Machine Learning, we recommend setting up policies to allow only for usage of specific VM SKUs. Policies can help prevent and control selection of expensive VMs. Policies can also be used to enforce usage of low-priority VM SKUs.

Allocate and manage quota based on business priority

Azure allows you to set limits for quota allocation on a subscription and Azure Machine Learning workspace level. Restricting who can manage quota through Azure role-based access control (RBAC) can help ensure resource utilization and cost predictability.

Availability of GPU quota can be scarce across your subscriptions. To ensure high quota utilization across workloads, we recommend monitoring whether quota is best used and assigned across workloads.

At Microsoft, it's determined periodically whether GPU quotas are best used and allocated across machine learning teams by evaluating capacity needs against business priority.

Commit capacity ahead of time

If you have a good estimate of how much compute will be used in the next year or next few years, you can purchase Azure Reserved VM Instances at a discounted cost. There are one-year or three-year purchase terms. Because Azure Reserved VM Instances are discounted, there can be significant cost savings compared to pay-as-you go prices.

Azure Machine Learning supports reserved compute instances. Discounts are automatically applied against Azure Machine Learning managed compute.

Manage data retention

Every time a machine learning pipeline is executed, intermediate datasets can be generated at each pipeline step for data caching and reuse. The growth of data as an output of these machine learning pipelines can become a pain point for an organization that is running many machine learning experiments.

Data scientists typically don't spend their time to clean up the intermediate datasets that are generated. Over time, the amount of data that is generated will add up. Azure Storage comes with a capability to enhance the management of the data lifecycle. Using Azure Blob Storage lifecycle management, you can set up general policies to move data that is unused into colder storage tiers and save costs.

Infrastructure cost optimization considerations

Networking

Azure networking cost is incurred from outbound bandwidth from Azure datacenter. All inbound data to an Azure datacenter is free. The key to reduce network cost is to deploy all your resources in the same datacenter region whenever possible. If you can deploy Azure Machine Learning workspace and compute in the same region that has your data, you can enjoy lower cost and higher performance.

You might want to have private connection between your on-premises network and your Azure network to have a hybrid cloud environment. ExpressRoute enables you to do that but considering the high cost of ExpressRoute, it might be more cost effective to move away from a hybrid cloud setup and move all resources to Azure cloud.

Azure Container Registry

For Azure Container Registry, the determining factors for cost optimization include:

- Required throughput for Docker image downloads from the container registry to Azure Machine Learning
- Requirements for enterprise security features, such as Azure Private Link

For production scenarios where high throughput or enterprise security is required, the Premium SKU of Azure Container Registry is recommended.

For dev/test scenarios where throughput and security are less critical, we recommend either Standard SKU or Premium SKU.

The Basic SKU of Azure Container Registry isn't recommended for Azure Machine Learning. It's not recommended because of its low throughput and low included storage, which can be quickly exceeded by Azure Machine Learning's relatively large sized (1+ GB) Docker images.

Consider computing type availability when choosing Azure regions

When you pick a region for your compute, keep the compute quota availability in mind. Popular and larger regions such as East US, West US, and West Europe tend to have higher default quota values and greater availability of most CPUs and GPUs, compared to some other regions with stricter capacity restrictions in place.

Learn more

Track costs across business units, environments, or projects by using the Cloud Adoption Framework

Next steps

To learn more about how to organize and set up Azure Machine Learning environments, see Organize and set up Azure Machine Learning environments.

Organize and set up Azure Machine Learning environments

To learn about best practices on Machine Learning DevOps with Azure Machine Learning, see Machine learning DevOps guide.

Machine learning DevOps guide

Machine learning DevOps guide

7/15/2021 • 18 minutes to read • Edit Online

Machine learning DevOps (MLOps) is an organizational change that relies on a combination of people, process, and technology to deliver machine learning solutions in a robust, scalable, reliable, and automated way. This guide provides a balanced view across the three areas of people, process, and technology. It summarizes best practices and learnings from adopting machine learning DevOps in the enterprise with Azure Machine Learning.

This article is subdivided into three sections:

- **An introduction to machine learning DevOps (MLOps):** A conceptual introduction that provides a balanced view across the three areas of people, process, and technology to anyone new to machine learning DevOps.
- **MLOps best practices with Azure Machine Learning:** Key considerations and recommendations to apply MLOps practices scoped to a single machine learning project.
- **MLOps at organizational scale: AI factories:** Key considerations and recommendations to drive organizational change with MLOps in an enterprise environment.

An introduction to machine learning DevOps (MLOps)

Why MLOps?

Modern machine learning algorithms and frameworks make it increasingly easy to develop models that can make accurate predictions.

You might have built a machine learning model that exceeds all of your accuracy expectations and impresses your business sponsors. Now it's time to deploy the model into production and it might not be as easy as you had expected. There are likely many things to put in place before your model can be used.

Over time, you or one of your colleagues might develop a new model that can do better than the old model. The question you must ask is, can you implement it without disrupting your business? It might be necessary for regulatory purposes to re-create the model and explain the model's predictions if unusual or biased predictions are made. Data inputted to your training and model can change over time. Because of these changes, it might be necessary to retrain the model periodically to maintain the accuracy of its predictions. Who is responsible to feed the data, monitor the performance, retrain the model, and fix it should it fail?

If you experience these challenges, you might want to consider implementing a machine learning DevOps (MLOps) strategy for your project. At a high level, MLOps refers to the application of DevOps principles to AI-infused applications. Consider a common use case: Suppose you have an application that serves a model's predictions via a REST API. Even a simple use case such as this can face many issues in production. Some MLOps tasks fit well in the general DevOps framework. For example, to set up unit tests and integration tests, or track changes through version control. Other tasks are more unique to MLOps that include how to:

- Enable continuous experimentation and comparison against a baseline model
- Monitor the incoming data to detect data drift
- Trigger model retraining and set-up a rollback just in case
- Create reusable data pipelines that can be applied for both training and scoring

Ultimately, the goal of MLOps is to close the gap between development and production and deliver value to customers faster. To achieve this goal, you must rethink how things are done in development and in production. The extent to which data scientists are involved in MLOps is an organizational choice. The role of Data Scientist is defined differently across different organizations. We recommend you review the MLOps maturity model to see

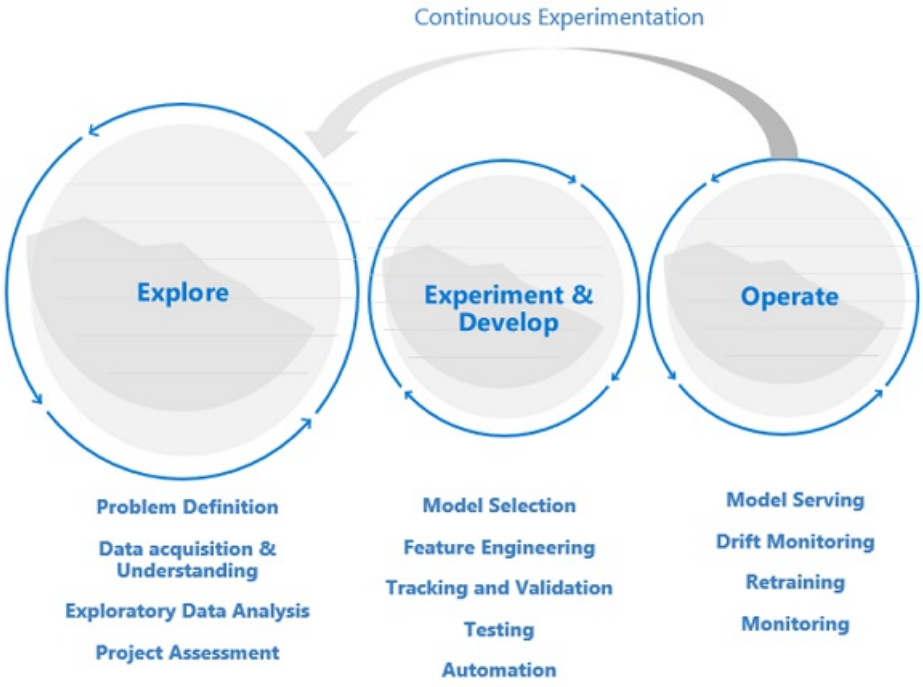
where your organization is and where your organization wants to be on the maturity scale.

How machine learning DevOps is different than DevOps

Exploration precedes development and operations

Data science projects are different from application development or data engineering projects. Data science projects may or may not make it to production. After an initial analysis, it might become clear that the business outcome can't be achieved with the available datasets. Because of this reason, an exploration phase is usually the first step in a data science project. The goal in this phase is to define and refine the problem and run exploratory data analysis, in which statistics and visualizations are used to confirm or falsify the problem hypotheses. There needs to be a common understanding that the project might not extend beyond this phase. It's important to make this phase as seamless as possible to have a quick turnaround. Unless there's an element of security, which enforces processes and procedures, they should be avoided and the Data Scientist should be allowed to work with the tool and data of their choice. Real data is needed for data exploration work.

The experimentation and development stage usually begins when there is enough confidence that the data science project is feasible and can provide real business value. This stage is when development practices become increasingly important. It's a good practice to capture metrics for all of the experiments that are done at this stage. It's also important to incorporate source control so that it's possible to compare models and toggle between different versions of the code if needed. Development activities include the refactoring, testing, and automation of exploration code into repeatable experimentation pipelines, and the creation of model serving applications and pipelines. Refactoring code into more modular components and libraries helps increase reusability and testability, and it allows for performance optimization. Finally, what is deployed into staging and production environments is the model serving application or batch inference pipelines. Next to monitoring of infrastructure reliability and performance, similar to what's done for a regular application with traditional DevOps, the quality of the data, the data profile, and the model must be continuously monitored at the risk of degradation or drift. Machine learning models require retraining over time to stay relevant in a changing environment.



Data science lifecycle requires an adaptive way of working

If you apply a typical DevOps way of working to a data science project, you might not find success because of the uncertain nature of data quality and its correlatively. Exploration and experimentation are recurring activities and needs throughout a machine learning project. The teams at Microsoft follow a project lifecycle and working process that was developed to reflect data science-specific activities. The Team Data Science Process and The Data Science Lifecycle Process are examples of reference implementations.

Data quality requirements and data availability constrain the work environment

For a machine learning team to effectively develop machine learning-infused applications, production data access is desirable across work environments. If production data access isn't possible because of compliance requirements or technical constraints, consider implementing Azure role-based access control (RBAC) with Azure Machine Learning, Just-in-Time access, or data movement pipelines to create production data replicas and enable user productivity.

Machine learning requires a greater operational effort

Unlike traditional software, a machine learning solution is constantly at risk of degradation because of its dependency on data quality. To maintain a qualitative solution once in production, continuous monitoring and re-evaluation of data and model quality is critical. It's expected that a production model requires timely retraining, redeployment, and tuning. These tasks come on top of day-to-day security, infrastructure monitoring, or compliance requirements and require special expertise.

Machine learning teams requires specialists and domain experts

While data science projects share roles with regular IT projects, the success of a machine learning team highly depends on a group of machine learning technology specialists and domain subject matter experts. Where the technology specialist has the right background to do end-to-end machine learning experimentation, the domain expert can support the specialist to analyze and synthesize the data, or qualify the data for use.

Common technical roles that are unique to data science projects are Domain Expert, Data Engineer, Data Scientist, AI Engineer, Model Validator, and Machine Learning Engineer. To learn more about roles and tasks within a typical data science team, also refer to the Team Data Science Process.

Seven principles for machine learning DevOps

When you plan to adopt MLOps for your next machine learning project, consider applying the following core principles as the foundation to any project.

- **Version control code, data, and experimentation outputs:** Unlike traditional software, data has a direct influence on the quality of machine learning models. Along with versioning your experimentation code base, version your datasets to ensure you can reproduce experiments or inference results. Versioning experimentation outputs like models can save effort and the computational cost of recreation.
- **Use multiple environments:** To segregate development and testing from production work, replicate your infrastructure in at least two environments. Access control for users might differ in each environment.
- **Manage infrastructure and configurations-as-code:** When you create and update infrastructure components in your work environments, use infrastructure as code to prevent inconsistencies between environments. Manage machine learning experiment job specifications as code, so that you can easily rerun and reuse a version of your experiment across environments.
- **Track and manage machine learning experiments:** Track the performance KPIs and other artifacts of your machine learning experiments. When you keep a history of job performance, it allows for a quantitative analysis of experimentation success, and enables greater team collaboration and agility.
- **Test code, validate data integrity, model quality:** Test your experimentation code base that includes correctness of data preparation functions, feature extraction functions, checks on data integrity, and obtained model performance.
- **Machine learning continuous integration and delivery:** Use continuous integration to automate test execution in your team. Include model training as part of continuous training pipelines, and include A/B testing as part of your release, to ensure that only a qualitative model might land in production.
- **Monitor services, models, and data:** When you serve machine learning models in an operationalized environment, it's critical to monitor these services for their infrastructure uptime and compliance, and for model quality. Set up monitoring to identify data and model drift, to understand whether retraining is required, or to set up triggers for automatic retraining.

Machine learning DevOps (MLOps) best practices with Azure Machine Learning

Azure Machine Learning offers several asset management, orchestration, and automation services to help you manage the lifecycle of your model training and deployment workflows. This section discusses best practices and recommendations to apply MLOps across the areas of people, process, and technology supported by Azure Machine Learning.

People

- Work in project teams to best utilize specialist and domain knowledge in your organization. Organize and set up Azure Machine Learning Workspaces on a project basis to comply with use case segregation requirements.
- Define a set of responsibilities and tasks in your organization as a role, where one team member on an MLOps project team could fulfill multiple roles. Use Custom Roles in Azure to define a set of granular Azure RBAC operations for Azure Machine Learning that each role can perform.
- Standardize on a project lifecycle and agile methodology. The Team Data Science Process provides a reference lifecycle implementation.
- Balanced teams can execute all MLOps stages from exploration to development to operations.

Process

- Standardize on a code template to allow for code reuse and increase ramp up time at project start or when a new team member joins the project. Azure Machine Learning pipelines and job submissions script, and CI/CD pipelines lend themselves well for the creation of templates.
- Use version control. Jobs that are submitted from a Git-backed folder automatically track repo metadata with the job in Azure Machine Learning for reproducibility.
- Version experiment inputs and outputs to enable reproducibility. Use Azure Machine Learning datasets, model management, and environment management capabilities to facilitate.
- Build up a run history of experiment runs to allow for comparison, planning, and collaboration. Use an experiment tracking framework like MLflow for metric collection.
- Continuously measure and control the quality of your team's work through continuous integration on the full experimentation code base.
- Early-terminate training when a model doesn't converge. Use an experiment tracking framework in combination with the run history in Azure Machine Learning to monitor job execution.
- Define an experiment and model management strategy. Consider using a name like *Champion* to refer to the current baseline model, or refer to *Challenger* models for candidate models, which could outperform the *Champion* model in production. Apply tags in Azure Machine Learning to mark experiment and models as appropriate. In some scenarios, such as sales forecasting, it can take months to determine whether the model's predictions are accurate.
- Elevate continuous integration to continuous training by including model training as part of the build. For example, start model training on the full dataset with each pull request.
- Shorten the time-to-feedback on the quality of machine learning pipeline by running an automated build on a sample of the data. Use Azure Machine Learning pipeline parameters to parameterize input datasets.
- Use continuous deployment for machine learning models to automate the deployment and testing of real time scoring services across your Azure environments (development, test, production).
- In some regulated industries, model validation steps might be required before a machine learning model can be used in a production environment. By automating validation steps, to an extent, you might accelerate time to delivery. When manual review or validation steps are still the bottleneck, consider whether it's possible to certify the automated model validation pipeline. Use resource tags in Azure Machine Learning to indicate asset compliance, candidates for review, or as triggers for deployment.
- Don't retrain in production and directly replace the production model without any integration testing. Even though model performance and functional requirements are good, among other potential issues, a model

might have grown its environment footprint, breaking the serving environment.

- When production data access is only available in production, use Azure RBAC and custom roles to give a select number of machine learning practitioners the read access they require, for example for data exploration. Alternatively, make a data copy available in the non-production environments.
- Agree on naming conventions and tags for Azure Machine Learning experiments to differentiate retraining baseline machine learning pipelines from experimental work.

Technology

- When you submit jobs via the studio user interface (UI) or CLI interface, instead of submitting jobs via the SDK, use the CLI or Azure DevOps Machine Learning tasks to configure automation pipeline steps. This process might reduce the code footprint by reusing the same job submissions directly from automation pipelines.
- Use event-based programming. For example, trigger an offline model testing pipeline using an Azure Function once a new model gets registered. Or send a notification to an `Ops` email alias when a critical pipeline fails to run. Azure Machine Learning produces events to Event Grid that you can subscribe to.
- When you use Azure DevOps for automation, use Azure DevOps Tasks for Machine Learning to use machine learning models as pipeline triggers.
- When you develop Python packages for your machine learning application, you can host them in an Azure DevOps repository as artifacts and publish them as a feed. This approach allows you to integrate the DevOps workflow for building packages with your Azure Machine Learning workspace.
- Consider the use of a staging environment to system integration test machine learning pipelines with upstream or downstream application components.
- Create unit and integration tests for your inference endpoints for enhanced debugging and accelerated time to deployment.
- To trigger retraining, use dataset monitors and use event-driven workflows to subscribe to data drift events and automate the trigger of machine learning pipelines for retraining.

MLOps at organizational scale: AI factories

A data science team might decide they can manage a handful of machine learning use cases internally. The adoption of machine learning DevOps (MLOps) helps set up project teams for better quality, reliability, and maintainability of solutions through balanced teams, supported processes, and technology automation. This adoption allows the team to scale and focus on the development of new use cases.

As the number of use cases grows in an organization, the management burden of supporting these use cases grows linearly, or even more. The challenge becomes how to use organizational scale to accelerate time-to-market, quicker assessment of use case feasibility, enable repeatability, and how to best utilize the available resources and skillsets across the full range of projects.

An AI factory is the development of repeatable business process, and a collection of standardized artifacts to accelerate the development and deployment of a large set of machine learning use cases by optimizing team set-up, recommended practices, MLOps strategy, architectural patterns, and reusable templates tailored to business requirements.

Standardize on repeatable architectural patterns

Repeatability is a key part of developing a factory process. Data science teams can accelerate project development and improve consistency across projects by developing a few repeatable architectural patterns that cover most of the machine learning use cases for their organization. Once these patterns are in place, most projects can use these patterns and reap the following benefits:

- Accelerated design phase
- Accelerated approvals from IT and security teams when they reuse tools across projects
- Accelerated development due to reusable infrastructure as code templates and project templates, which are

covered in more detail in the next section

The architectural patterns can include but aren't limited to the following topics:

- Preferred services for each stage of the project
- Data connectivity and governance
- A machine learning DevOps (MLOps) strategy tailored to the requirements of the industry, business, or data classification
- Experiment management process *Champion* or *Challenger* models

Facilitate cross-team collaboration and sharing

Shared code repositories and utilities can accelerate the development of machine learning solutions. These repositories can be developed in a modular way during project development so they are generic enough to be used by other projects. They can be made available in a central repository that all data science teams can access.

Share and reuse of intellectual property

At the beginning of a project, the following intellectual property should be reviewed to maximize code reuse:

- Internal code, such as packages and modules, which have been designed for reuse within the organization.
- Datasets, which have been created in other machine learning projects or that are available in the Azure ecosystem.
- Existing data science projects with similar architecture and business problem.
- GitHub or open-source repos that can accelerate the project.

Project retrospectives should include an action item to review if there are elements of the project can be shared and generalized for broader reuse, so that the list of assets listed above organically grows with time.

To help sharing and discovery, many companies have introduced shared repositories for the organization of code snippets and machine learning artifacts. Artifacts in Azure Machine Learning can be defined-as-code, including datasets, models, environments, and pipelines, that allows you to share efficiently across projects and workspaces.

Project templates

To accelerate the migration of existing solutions, and maximize code reuse when starting a new project, many companies have standardized on a project template to kick start a new project. Examples of project templates recommended for use with Azure Machine Learning are Azure Machine Learning examples, Data science lifecycle process, and Team data science process.

Central data management

The process to get access to data for exploration or production usage can be time-consuming. Many companies centralize their data management to bring data producers and data consumers together and to help with easier data access for machine learning experimentation.

Shared utilities

Enterprise-wide centralized dashboards can be implemented to consolidate logging and monitoring information, such as error logging, service availability and telemetry, and model performance monitoring.

Use Azure Monitor metrics for Azure Machine Learning and associated services like Azure Storage to build a dashboard to keep track of experimentation progress, compute infrastructure health or GPU quota utilization.

Create a specialist machine learning engineering team

Many companies have implemented the role of the machine learning engineer who specializes in the creation and operation of robust machine learning pipelines, drift monitoring and retraining workflows, monitoring dashboards, and drives overall responsibility for industrializing the machine learning solution from development to production. They work closely with data engineering, architects, and security and operations to ensure all the necessary controls are in place.

While data science requires deep domain expertise, machine learning engineering as a discipline is more technical in focus. This difference makes the machine learning engineer more flexible to work across various projects and business departments. Large data science practices can benefit from a specialist machine learning engineering team that drives repeatability and reuse of automation workflows across various use cases and business departments.

Enablement and documentation

It's important to provide clear guidance on the AI factory process to new and existing teams and users. This guidance will ensure consistency and reduce the amount of effort required of the machine learning engineering team when they industrialize a project. Consider designing content specifically for the various roles in your organization.

Everyone has a unique learning style, so a mixture of the following types of documents can help accelerate the adoption of the AI factory framework.

- Central hub with links to all artifacts. For example, this hub can be a Teams team or a SharePoint site
- Training and enablement plan designed for each role
- High-level summary presentation of the approach along with a companion video
- Detailed document or playbook
- How-to videos
- Readiness assessments

Azure MLOps Video Series

In this section, these Azure MLOps video series show how to enable MLOps for your machine learning solution from your initial developemnt to production.

Ethics

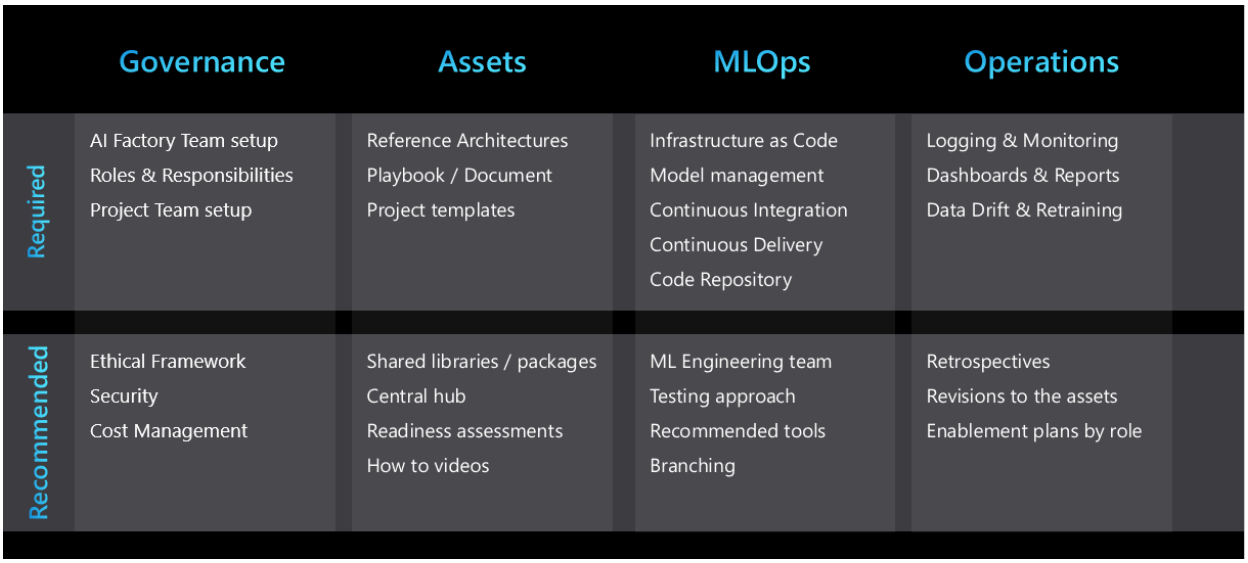
Ethics play an instrumental role in the design of an AI solution. If ethical principles aren't implemented, trained models can exhibit the same bias present in the data they were trained on. This issue can result in the project being discontinued and more importantly, it can risk the organization's reputation.

To ensure that the key ethical principles that the company stands for are implemented across projects, a list of these principles, along with ways of validating them from a technical perspective during the testing phase, should be provided. Use the machine learning features in Azure Machine Learning to learn what responsible machine learning is, and to learn ways you can put it into practice.

Summary

An AI factory relies on building repeatable processes and reusable assets that can help an organization scale from tens of use cases to thousands of use cases in a scalable way.

The following figure summarizes key elements of an AI factory:



Not every organization's machine learning DevOps (MLOps) requirements are the same. The MLOps architecture for a large multinational enterprise is unlikely to fit a small startup. Organizations start small and build up as their maturity, model catalog, and experience grows.

Next steps

To learn more about how to organize and set up Azure Machine Learning environments, see Organize and set up Azure Machine Learning environments. Also, you can watch hands-on Azure MLOps video series.

Organize and set up Azure Machine Learning environments

To learn more about how to manage budgets, quota and cost with Azure Machine Learning, see Manage budgets, costs, and quota for Azure Machine Learning at organizational scale.

Manage budgets, costs, and quota for Azure Machine Learning at organizational scale

Track costs across business units, environments, or projects

3/6/2021 • 8 minutes to read • Edit Online

Building a cost-conscious organization requires visibility and properly defined access (or scope) to cost-related data. This best-practice article outlines decisions and implementation approaches to creating tracking mechanisms.

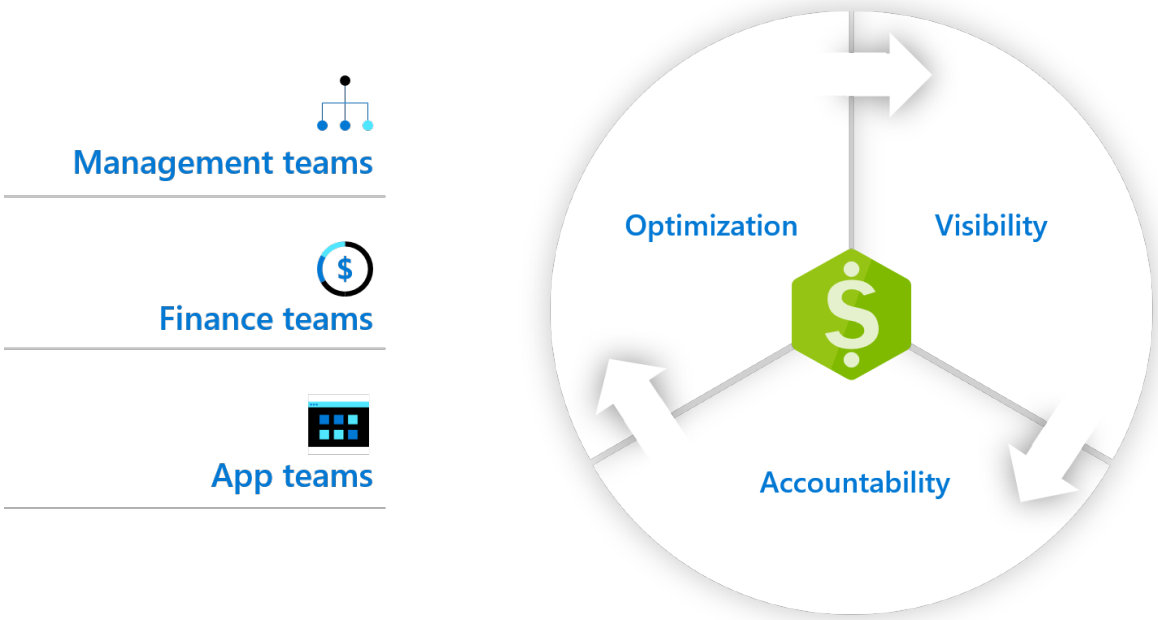


Figure 1: Outline of a cost-conscious process.

Establish a well-managed environment hierarchy

Cost control, much like governance and other management constructs, depends on a well-managed environment. Establishing such an environment (especially a complex one) requires consistent processes in the classification and organization of all assets.

Assets (also known as resources) include all virtual machines, data sources, and applications deployed to the cloud. Azure provides several mechanisms for classifying and organizing assets. Organize and manage your Azure subscriptions details options for organizing resources based on multiple criteria to establish a well-managed environment. This article focuses on the application of Azure fundamental concepts to provide cloud cost visibility.

Classification

Tagging is an easy way to classify assets. Tagging associates metadata to an asset. That metadata can be used to classify the asset based on various data points. When tags are used to classify assets as part of a cost management effort, companies often need the following tags: business unit, department, billing code, geography, environment, project, and workload or application categorization. Azure Cost Management + Billing can use these tags to create different views of cost data.

Tagging is a primary way to understand the data in any cost reporting. It's a fundamental part of any well-managed environment. It's also the first step in establishing proper governance of any environment.

The first step in accurately tracking cost information across business units, environments, and projects is to

define a tagging standard. The second step is to ensure that the tagging standard is consistently applied. The following articles can help you accomplish: developing naming and tagging standards and establishing a governance MVP to enforce tagging standards

Resource organization

There are several approaches to organizing assets. This section outlines a best practice based on the needs of a large enterprise with cost structures spread across business units, geographies, and IT organizations. A similar best practice for a smaller, less complex organization is available in the standard enterprise governance guide.

For a large enterprise, the following model for management groups, subscriptions, and resource groups will create a hierarchy that allows each team to have the right level of visibility to perform their duties. When the enterprise needs cost controls to prevent budget overrun, it can apply governance tooling like Azure Blueprints or Azure Policy to the subscriptions within this structure to quickly block future cost errors.

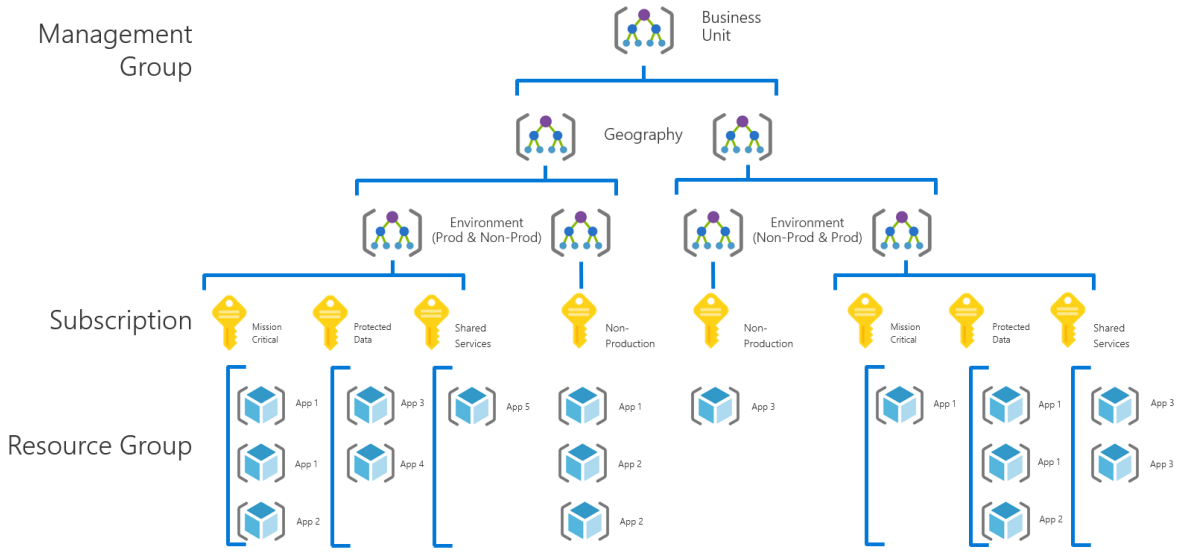


Figure 2: Resource organization for a large enterprise.

In the preceding diagram, the root of the management group hierarchy contains a node for each business unit. In this example, the multinational company needs visibility into the regional business units, so it creates a node for geography under each business unit in the hierarchy.

Within each geography, there's a separate node for production and nonproduction environments to isolate cost, access, and governance controls. To allow for more efficient operations and wiser operations investments, the company uses subscriptions to further isolate production environments with varying degrees of operational performance commitments. Finally, the company uses resource groups to capture deployable units of a function, called applications.

The diagram shows best practices but doesn't include these options:

- Many companies limit operations to a single geopolitical region. That approach reduces the need to diversify governance disciplines or cost data based on local data-sovereignty requirements. In those cases, a geography node is unnecessary.
- Some companies prefer to further segregate development, testing, and quality control environments into separate subscriptions.
- When a company integrates a cloud center of excellence (CCoE) team, shared services subscriptions in each geography node can reduce duplicated assets.
- Smaller adoption efforts might have a much smaller management hierarchy. It's common to see a single root node for corporate IT, with a single level of subordinate nodes in the hierarchy for various environments. This isn't a violation of best practices for a well-managed environment. But it does make it more difficult to provide a least-rights access model for cost control and other important functions.

The rest of this article assumes the use of the best-practice approach in the preceding diagram. But the following articles can help you apply the approach to a resource organization that best fits your company:

- Scale your Azure environment with multiple subscriptions
- Organize and manage your Azure subscriptions
- Deploy a governance MVP to govern well-managed environment standards

Provide the right level of cost access

Managing cost is a team activity. The organization readiness section of the Cloud Adoption Framework defines a small number of core teams and outlines how those teams support cloud adoption efforts. This article expands on the team definitions to define the scope and roles to assign to members of each team for the proper level of visibility into cost management data.

Roles define what a user can do to various assets. The **Scope** defines which assets (user, group, service principal, or managed identity) a user can do those things to.

As a general best practice, we suggest a least-privilege model in assigning people to various roles and scopes.

Roles

Azure Cost Management + Billing supports the following built-in roles for each scope:

- Owner: Can view costs and manage everything, including cost configuration.
- Contributor: Can view costs and manage everything, including cost configuration, but excluding access control.
- Reader: Can view everything, including cost data and configuration, but can't make changes.
- Cost Management Contributor: Can view costs and manage cost configuration.
- Cost Management Reader: Can view cost data and configuration.

As a general best practice, members of all teams should be assigned the role of Cost Management Contributor. This role grants access to create and manage budgets and exports to more effectively monitor and report on costs. But members of the cloud strategy team should be set to Cost Management Reader only. That's because they're not involved in setting budgets within the Azure Cost Management + Billing tool.

Scope

The following scope and role settings will create the required visibility into cost management. This best practice might require minor changes to align to asset organization decisions.

- Cloud adoption team. Responsibilities for ongoing optimization changes require Cost Management Contributor access at the resource group level.
    - **Working environment.** At a minimum, the cloud adoption team should already have Contributor access to all affected resource groups, or at least those groups related to dev/test or ongoing deployment activities. No additional scope setting is required.
    - **Production environments.** When proper separation of responsibility has been established, the cloud adoption team probably won't continue to have access to the resource groups related to its projects. The resource groups that support the production instances of their workloads will need additional scope to give this team visibility into the production cost impact of its decisions. Setting the Cost Management Contributor scope for production resource groups for this team will allow the team to monitor costs and set budgets based on usage and ongoing investment in the supported workloads.
- Cloud strategy team. Responsibilities for tracking costs across multiple projects and business units require Cost Management Reader access at the root level of the management group hierarchy.
    - Assign Cost Management Reader access to this team at the management group. This will ensure

ongoing visibility into all deployments associated with the subscriptions governed by that management group hierarchy.

- Cloud governance team. Responsibilities for managing cost, budget alignment, and reporting across all adoption efforts requires Cost Management Contributor access at the root level of the management group hierarchy.
  - In a well-managed environment, the cloud governance team likely has a higher degree of access already, making additional scope assignment for Cost Management Contributor unnecessary.
- Cloud center of excellence. Responsibility for managing costs related to shared services requires Cost Management Contributor access at the subscription level. Additionally, this team might require Cost Management Contributor access to resource groups or subscriptions that contain assets deployed by CCoE automations to understand how those automations affect costs.
  - **Shared services.** When a cloud center of excellence is engaged, best practice suggests that assets managed by the CCoE are supported from a centralized shared service subscription within a hub and spoke model. In this scenario, the CCoE likely has Contributor or Owner access to that subscription, making additional scope assignment for Cost Management Contributor unnecessary.
  - **CCoE automation/controls.** The CCoE commonly provides controls and automated deployment scripts to cloud adoption teams. The CCoE has a responsibility to understand how these accelerators affect costs. To gain that visibility, the team needs Cost Management Contributor access to any resource groups or subscriptions running those accelerators.
- **Cloud operations team.** Responsibility for managing ongoing costs of production environments requires Cost Management Contributor access to all production subscriptions.
  - The general recommendation puts production and nonproduction assets in separate subscriptions that are governed by nodes of the management group hierarchy associated with production environments. In a well-managed environment, members of the operations team likely have Owner or Contributor access to production subscriptions already, making the Cost Management Contributor role unnecessary.

Additional cost management resources

Azure Cost Management + Billing is a well-documented tool for setting budgets and gaining visibility into cloud costs for Azure or AWS. After you establish access to a well-managed environment hierarchy, the following articles can help you use that tool to monitor and control costs.

Get started with Azure Cost Management + Billing

To begin using Azure Cost Management + Billing, see How to optimize your cloud investment with Azure Cost Management + Billing.

Use Azure Cost Management + Billing

- Create and manage budgets
- Export cost data
- Optimize costs based on recommendations
- Use cost alerts to monitor usage and spending

Use Azure Cost Management + Billing to govern AWS costs

- Set up AWS Cost and Usage report integration
- Manage AWS costs

Establish access, roles, and scope

- Understanding cost management scope
- Setting scope for a resource group

Skills readiness path during the readiness phase of a migration journey

5/24/2021 • 3 minutes to read • Edit Online

During the readiness phase of a migration journey, the objective is to prepare for the journey ahead. This phase is accomplished in two primary areas: organizational readiness and environmental (technical) readiness. Each area might require new skills for both technical and nontechnical contributors. The following sections describe a few options to help build the necessary skills.

Organizational readiness learning paths

Depending on the motivations and business outcomes associated with a cloud adoption effort, leaders might be required to establish new organizational structures or virtual teams to facilitate various functions. The following articles help to develop the skills that are necessary to structure those teams in accordance with desired outcomes:

- Initial organization alignment: Overview of organizational alignment and various team structures to facilitate specific goals.
- Break down silos and fiefdoms: Understand two common organizational antipatterns and ways to guide the team to productive collaboration.

Environmental (technical) readiness learning paths

During the readiness phase, technical staff are called upon to create a migration landing zone that's capable of hosting, operating, and governing workloads that were migrated to the cloud. Developing the necessary skills can be accelerated with the following learning paths:

- Create an Azure account: The first step to using Azure is to create an account. Your account holds the Azure services you provision and handles your personal settings like identity, billing, and preferences.
- Azure portal: Tour the Azure portal features and services, and customize the portal.
- Introduction to Azure: Get started with Azure by creating and configuring your first virtual machine in the cloud.
- Introduction to security in Azure: Discuss the basic concepts for protecting your infrastructure and data when you work in the cloud. Understand what responsibilities are yours and what Azure takes care of for you.
- Manage resources in Azure: Learn how to work with the Azure command line and web portal to create, manage, and control cloud-based resources.
- Create a VM: Create a virtual machine by using the Azure portal.
- Azure networking: Learn some of the Azure networking basics and how Azure networking helps improve resiliency and reduce latency.
- Azure compute options: Review the Azure compute services.
- Secure resources with Azure role-based access control (Azure RBAC): Use Azure RBAC to secure resources.
- Data storage options: Benefits of Azure data storage.

During the readiness phase, architects are called upon to architect solutions that span all Azure environments. The following skill-building resources can prepare architects for these tasks:

- Foundations for Cloud Architecture: A Pluralsight course to help architect the right foundational solutions.
- Microsoft Azure architecture: A Pluralsight course to ground architects in Azure architecture.

- Designing migrations for Microsoft Azure: A Pluralsight course to help architects design a migration solution.

Deeper skills exploration

Various learning options beyond these initial options are available for developing skills.

Typical mappings of cloud IT roles

Microsoft and partners offer various options for all audiences to develop skills with Azure services.

- Map roles and skills: A resource for mapping your cloud career path. Learn about your cloud role and suggested skills. Follow a learning curriculum at your own pace to build the skills that you need most to stay relevant.
- Explore Azure certification training and exams to gain official recognition for your Azure knowledge.

Microsoft Learn

Microsoft Learn is a new approach to learning. Readiness for the new skills and responsibilities that come with cloud adoption doesn't come easily. Microsoft Learn provides a more rewarding approach to hands-on learning that helps you achieve your goals faster. Earn points and levels and achieve more.

The following examples are a few tailored learning paths on Microsoft Learn, which align to the Ready methodology of the Cloud Adoption Framework:

Azure fundamentals: Learn cloud concepts such as high availability, scalability, elasticity, agility, fault tolerance, and disaster recovery. Understand the benefits of cloud computing in Azure and how it can save you time and money. Compare and contrast basic strategies for transitioning to the Azure cloud. Explore the breadth of services available in Azure including compute, network, storage, and security.

Manage resources in Azure: Learn how to work with the Azure command line and web portal to create, manage, and control cloud-based resources.

Administer infrastructure resources in Azure: Learn how to create, manage, secure, and scale virtual machine resources.

Store data in Azure: Azure provides a variety of ways to store data: unstructured, archival, relational, and more. Learn the basics of storage management in Azure, how to create a storage account, and how to choose the right model for the data you want to store in the cloud.

Architect great solutions in Azure: Learn how to design and build secure, scalable, and high-performing solutions in Azure by examining the core principles found in sound architecture.

Learn more

For additional learning paths, browse the Microsoft Learn catalog. Use the **Roles** filter to align learning paths with your role.

Cloud readiness antipatterns

8/4/2021 • 4 minutes to read • Edit Online

Customers often experience antipatterns during the readiness phase of cloud adoption. These antipatterns can lead to unexpected downtime, disaster recovery problems, and availability issues.

Antipattern: Assume released services are ready for production

Because cloud computing is evolving rapidly, companies often release preview versions of new services. Customers tend to assume that they can use any available cloud service in a production environment. But, problems can result, for these reasons:

- Preview services usually don't provide uptime service-level agreements (SLAs).
- New services often aren't as mature as cloud services that are already available.

Example: Use a preview service in production

A research institute uses a preview cloud service in production. The service seems to be a good fit for its use case. But, the institute doesn't perform due diligence on the service. The institute also doesn't follow its reference architecture's requirements and guidelines.

Problems come up with the preview service that lead to unexpected downtime. The institute begins to think that cloud services in general aren't as mature or resilient as promised.

Preferred outcome: Use pre-approved cloud services in production

When evaluating new services that are in preview, only use these services in proof of concept (POC) scenarios. Don't use these services in production environments, because they don't have SLAs. Find the right balance between functionality and maturity when approving cloud services. See Cloud services due diligence checklist for an established framework that you can use to quickly evaluate cloud services.

Antipattern: Assume increased resiliency and availability

Cloud computing often offers advantages over on-premises computing. Examples include:

- **Increased resiliency:** Recovering after failure.
- **Availability:** Running in a healthy state without significant downtime.

Because most cloud services offer these advantages, many companies assume that all cloud services offer resiliency and high availability by default. In reality, these features are often only available at extra cost and with additional technical effort.

Example: Assume high availability

A start-up implements a mission-critical application on infrastructure as a service (IaaS) services. Developers at the start-up have looked into a virtual machine (VM) with an uptime SLA of 99.9%. Since they'd like to cut costs, they use a single VM and premium storage.

When the VM fails, their application can't recover. Unexpected downtime results. They'd assumed that the cloud offers high availability by default. They weren't aware that performance guarantees can differ between:

- Service models like platform as a service (PaaS) and software as a service (SaaS).
- Technical architectures like load-balanced availability sets and Availability Zones.

Preferred outcome: Reduce failures while balancing resiliency and costs

See trusted, mature resources for information on architectural best practices that can reduce the scope of failures:

- Reference architectures
- Microsoft Azure Well-Architected Framework

Identify the right balance between costs and features like high resiliency and availability. Increased resiliency and availability typically lead to increased costs. For instance:

- A single VM might have an SLA with a guaranteed uptime of 99.9%.
- Two VMs running the same workload would provide an SLA with an uptime between 99.95 and 99.99 percent.

Engage in the essential process of *requirements engineering* when designing a cloud-based solution. Use an SLA estimator to help calculate your application's end-to-end SLA.

Antipattern: Become a cloud provider

Some companies try to make their internal IT department a cloud provider. IT then becomes responsible for reference architectures. IT also needs to provide IaaS and PaaS to business units. Since this type of work isn't usually part of IT's core business, the resulting service offerings can lack usability, resiliency, efficiency, and security.

Example: Provide monolithic managed cloud services

A corporation's IT department establishes a cloud center of excellence (CCoE) that serves as a broker between IT and business units. To ensure the corporation is cloud-compliant, the managing board assigns the CCoE the task of providing monolithic end-to-end services. The CCoE sets up an internal cloud procurement portal that business units can use to order a fully managed cloud VM as a service. But, IT controls who can access and use the entire platform. As a result, IT actively prevents business units from taking advantage of the full range of services that Azure provides. Business units can't access the cloud portal. They only get access through Secure Shell (SSH) and Remote Desktop Protocol (RDP) to the server that they order.

For several reasons, the CCoE then has trouble providing a monolithic managed service to wrap each service that's available in the cloud:

- The cloud offers a large number of services across multiple solution areas. Compared with developing IaaS solutions, designing and engineering Internet of Things (IoT) and AI solutions requires different expertise and skill sets.
- Cloud services change frequently.
- Trying to provide monolithic services increases the time to market substantially, with IT managing the process, not the business units.

Preferred outcome: Provide guardrails

When adopting cloud technologies, have the IT department gain firsthand experience with the cloud by starting with IT workloads. Use the Microsoft Cloud Adoption Framework for Azure to identify your first adoption project.

Use a mature cloud operating model such as centralized operations that makes IT responsible for defining platform guardrails like governance. Then business units can adopt cloud projects in a secure and consistent manner, within the guardrails that IT defines.

Consider adopting only one major public cloud provider at the start, because all major platforms differ significantly in setup, management, and usage.

Use SaaS solutions as much as possible for IT tooling, such as:

- Code repositories.
- Continuous integration and continuous delivery (CI/CD).
- Collaboration systems.

For cloud workloads, advise IT to use familiar procedures that operate safely and securely at scale.

Next steps

- Overview of the reliability pillar
- First adoption project

Cloud migration in the Cloud Adoption Framework

6/22/2021 • 4 minutes to read • Edit Online

Any enterprise-scale cloud adoption plan, will include workloads that do not warrant significant investments in the creation of new business logic. Those workloads could be moved to the cloud through any number of approaches: lift and shift; lift and optimize; or modernize. Each of these approaches is considered a migration.

Watch the following video to get a quick overview of the lift and shift approach.

The following exercises will help establish the iterative processes to assess, migrate, optimize, secure, and manage those workloads.

To prepare you for this phase of the cloud adoption lifecycle, we recommend the following:

	Migrate your first workload: Use the Azure migration guide to become familiar with the Azure native tools and approach to migration.
	Migration scenarios: Use additional migration tools and approaches to act on other migration scenarios.
	Best practices: Address common migration needs through the application of consistent best practices.
	Process improvements: Migration is a process heavy activity. As migration efforts scale, use these process improvements to evaluate and mature various aspects of migration.

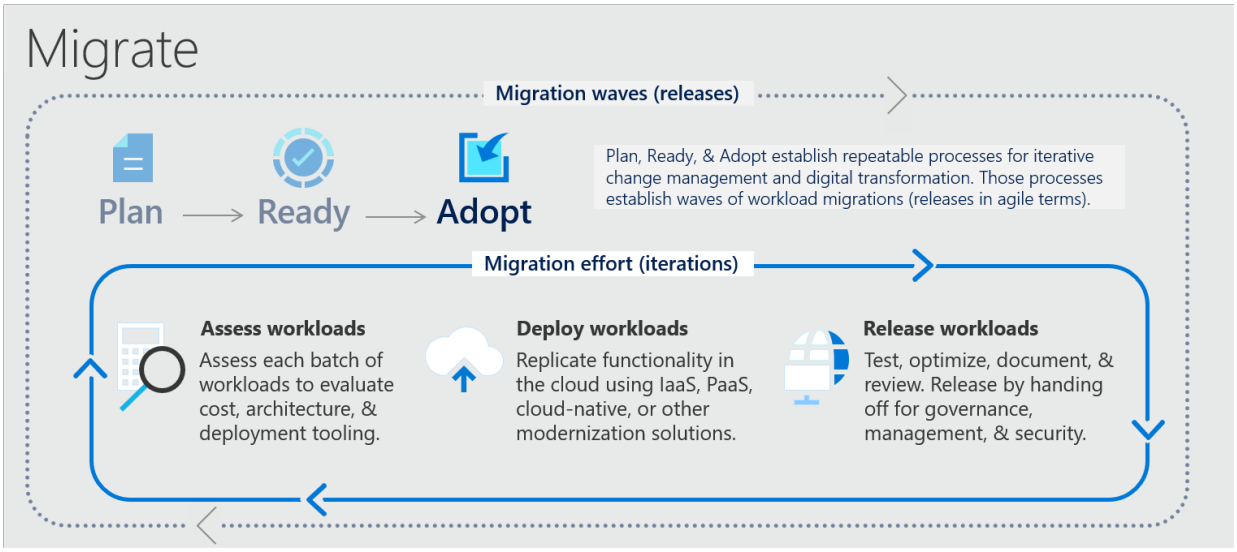
The Migrate methodology and the steps above build on the following assumptions:

- The methodology governing migration sprints fits within migration waves or releases, which are defined using the Plan, Ready, and Adopt methodologies. Within each migration sprint, a batch of workloads is migrated to the cloud.
- Before migrating workloads, at least one landing zone has been identified, configured, and deployed to meet the needs of the near-term cloud adoption plan.
- Migration is commonly associated with the terms *lift and shift* or *rehost*. This methodology and the above steps are built on the belief that no datacenter and few workloads should be migrated using a pure rehost approach. While many workloads can be rehosted, customers more often choose to modernize specific assets within each workload. During this iterative process, the balance between speed and modernization is a common discussion point.

Migration effort

The actions required to migrate workloads generally falls into three efforts (or phases) for each workload: assess workloads, deploy workloads, and release workloads. This section of the Cloud Adoption Framework teaches readers how to maximize the return from each phase required to migrate a workload to production.

In a standard two-week long iteration, an experienced migration team can complete this process for 2-5 workloads of low-medium complexity. More complex workloads, such as SAP, may take several two-week iterations to complete all three phases of migration effort for a single workload. Experience and complexity both have a significant impact on timelines and migration velocity.



The following bullets provide an overview of the phases of this process (pictured above):

- **Assess workloads:** Assess workloads to evaluate cost, modernization, and deployment tooling. This process focuses on validating or challenging the assumptions made during earlier discovery and assessments by looking more closely at rationalization options. This is also when user patterns and dependencies are studied more closely to ensure workloads will achieve technical success after migration.

  Watch this video to get a quick overview about completing a comprehensive assessment.

- **Deploy workloads:** After workloads are assessed, the existing functionality of those workloads is replicated (or improved) in the cloud. This could involve a *lift and shift* or *rehost* to the cloud. But more commonly during this phase, many of the assets supporting these workloads will be modernized to capitalize on the benefits of the cloud.
- **Release workloads:** Once functionality is replicated to the cloud, workloads can be tested, optimized, documented, and released for ongoing operations. Critical during this process, is the effort to review the migrated workloads and hand them off to governance, operations management, and security teams for ongoing support of those workloads.

> **NOTE**
> In some early iterations of migration effort, it is common to limit scope to a single workload. This approach maximizes skills retention and provides the team with more time to experiment and learn.

> **NOTE**
>
> When building a migration factory, some teams may choose to disperse each of the above phases across multiple teams and multiple sprints. This approach can improve repeatability and accelerate migration efforts.

Migration waves and iterative change management

Migration iterations deliver technical value by migrating assets and workloads. A migration wave is the smallest collection of workloads that deliver tangible and measurable business value. Each iteration should result in a report outlining the technical efforts completed. However, business change and strategic planning typically happen at a slightly higher level. As the cloud adoption team delivers on the migration effort, the cloud strategy team focuses on planning the next 1-2 migration waves. The cloud strategy team also tracks technical progress as a learning metric to better understand the timelines for realizing business value. In that regard, migration waves are the iterative change management approach to tracking business outcomes, people, and timelines.

As outlined in the graphic in the prior section, processes within the Plan methodology, the Ready methodology, and to some extent the Strategy methodology of the Cloud Adoption Framework provide guidance on planning and managing the migration waves. The management of those waves will prioritize and define the migration effort to be delivered by the technical teams.

Next steps

The steps outlined above, and subsequent guidance in the Migrate methodology, can help you develop skills to improve processes within each migration sprint. The Azure migration guide is a brief series of articles that outlines the most common tools and approaches needed during your first migration wave.

Azure migration guide

Azure migration guide overview

8/25/2021 • 2 minutes to read • Edit Online

The Cloud Adoption Framework's Migrate methodology guides readers through an iterative process of migrating one workload, or a small collection of workloads per release. In each iteration, the process of assess, migration, and optimize and promote is followed to ensure that workloads are ready to meet production demands. That cloud-agnostic process can guide migration to any cloud provider.

This guide demonstrates a simplified version of that process when migrating from your on-premises environment to **Azure**.

> **TIP**
> For an interactive experience, view this guide in the Azure portal. Go to the Azure Quickstart Center in the Azure portal, select **Azure migration guide**, and then follow the step-by-step instructions.

- Migration tools
- When to use this guide

This guide is the suggested path for your first migration to Azure, as it will expose you to the methodology and the cloud-native tools most commonly used during migration to Azure. Those tools are presented across the following pages:

- **Assess each workload's technical fit.** Validate the technical readiness and suitability for migration.
- **Migrate your services.** Perform the actual migration, by replicating on-premises resources to Azure.
- **Manage costs and billing.** Understand the tools required to control costs in Azure.
- **Optimize and promote.** Optimize for cost and performance balance before promoting your workload to production.
- **Get assistance.** Get help and support during your migration or post-migration activities.

It is assumed that a landing zone has already been deployed, in alignment with the best practices in the Cloud Adoption Framework's Ready methodology.

Assess workloads and refine plans

8/25/2021 • 6 minutes to read • Edit Online

The resources in this guide help you assess each workload, challenge assumptions about each workload's suitability for migration, and finalize architectural decisions about migration options.

- Tools
- Challenge assumptions
- Scenarios and stakeholders
- Timelines
- Cost management

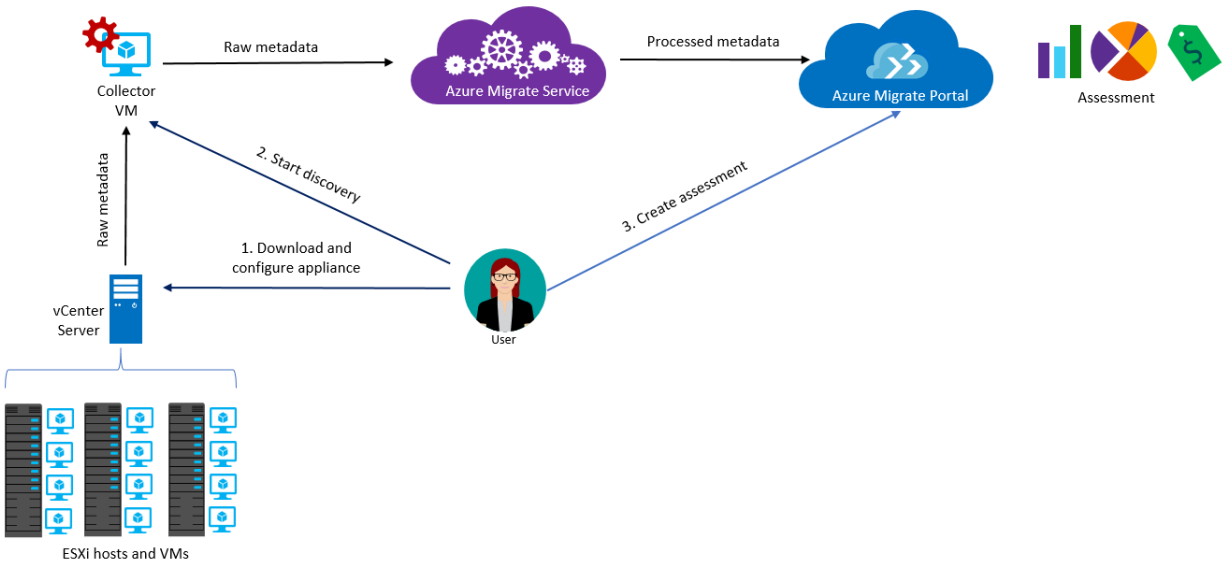
If you didn't follow the guidance in the links above, you will likely need data and an assessment tool to make informed migration decisions. Azure Migrate is the native tool for assessing **and** migrating to Azure. If you haven't already, use these steps to create a new server migration project and collect the necessary data.

Azure Migrate

Azure Migrate provides a centralized hub to assess and migrate to Azure on-premises servers, infrastructure, applications, and data. This service:

- Assesses whether on-premises servers are ready for migration to Azure.
- Estimates the size of Azure VMs/Azure SQL configuration/number of Azure VMware Solution nodes after migration.
- Estimates costs for running on-premises servers in Azure.
- Identifies cross-server dependencies and optimization strategies for moving interdependent servers to Azure.

If you're considering a lift-and-shift approach or are in the early assessment stages of migration, this service is for you. After completing the assessment, use Azure Migrate to execute the migration.



Create a new project

Begin server discovery, assessment, and migration using Azure Migrate via these steps:

1. Select **Azure Migrate**.
2. In **Overview**, select **Discover, assess and migrate**.

3. Select **Add tools**.
4. In **Project**, select your Azure subscription, then create a resource group if you don't have one.
5. In **Project Details**, specify the project name and geography where you want to create the project, then select **Create**.
6. After creating the project, tools are visible in project and user can start with Discovery.

DISCOVER, ASSESS AND MIGRATE

Learn more

- Azure Migrate overview
- Migrate physical or virtualized servers to Azure
- Azure Migrate in the Azure portal

Dependency analysis

Dependency analysis identifies dependencies between discovered on-premises servers. It provides these advantages:

- Gather servers into groups for assessment, more accurately, and with greater confidence.
- Identify servers that should be migrated together. This step is especially useful if you're unsure which servers are part of an app deployment that you want to migrate to Azure.
- Identify whether servers are in use and which servers can be decommissioned instead of migrated.
- Analyze dependencies to ensure that nothing is left behind and avoid surprise outages after migration.

Learn more

- Azure Migrate dependency analysis

Deploy workloads and assets (infrastructure, apps, and data)

7/21/2021 • 11 minutes to read • Edit Online

In this phase of the journey, you use the output of the assessment phase to initiate the migration of the environment. This guide helps identify the appropriate tools to reach a completed state. You'll explore native tools, third-party tools, and project management tools.

- Native migration tools
- Third-party migration tools
- Project management tools
- Cost management

The following sections describe the native Azure tools available to perform or assist with migration. For information on choosing the right tools to support your migration efforts, see the Cloud Adoption Framework's migration tools decision guide.

Azure Migrate

Azure Migrate delivers a unified and extensible migration experience. Azure Migrate provides a one-stop, dedicated experience to track your migration journey across the phases of assessment and migration to Azure. It provides you the option to use the tools of your choice and track the progress of migration across these tools.

Azure Migrate is a centralized hub to assess and migrate on-premises servers, infrastructure, applications, and data to Azure. It provides the following functionality:

- Unified platform with assessment, migration, and progress tracking.
- Enhanced assessment and migration capabilities:
  - Assess on-premises servers including SQL Server instances and migrate them to Azure virtual machines or Azure VMware Solution (AVS) (Preview).
  - Agentless migration of VMware virtual machines to Azure.
  - Assess on-premises databases and migrate them to Azure SQL Database or to SQL Managed Instance.
  - Assess on-premises web applications and migrate them to Azure App Service by using the Azure App Service Migration Assistant.
  - Assess your on-premises virtual desktop infrastructure (VDI) and migrate it to Azure Virtual Desktop.
  - Migrate large amounts of data to Azure quickly and cost-effectively using Azure Data Box products.
- Extensible approach with ISV integration (such as Cloudamize).

To perform a migration using Azure Migrate, follow these steps:

1. Search for Azure Migrate under **All services**. Select **Azure Migrate** to continue.
2. a. In **Overview**, select **Discover, assess and migrate**.
3. Select **Add tools**.
4. In **Project**, select your Azure subscription, then create a resource group if you don't have one.
5. In **Project Details**, specify the project name and geography where you want to create the project, then select **Create**.
6. After creating the project, tools are visible in project and user can start with Discovery.

> **NOTE**
>
> For guidance specific to your scenario refer to the tutorials and Azure Migrate documentation.

Learn more

- About Azure Migrate
- Azure Migrate tutorial: Migrate physical or virtualized servers to Azure

Azure Database Migration Service

Azure Database Migration Service is a fully managed service that enables seamless migrations from multiple database sources to Azure data platforms, with minimal downtime (online migrations). Database Migration Service performs all of the required steps. You can initiate your migration projects assured that the process takes advantage of best practices recommended by Microsoft.

Create an Azure Database Migration Service instance

If this is the first time using Azure Database Migration Service, you need to register the resource provider for your Azure subscription:

1. Select **All services** > **Subscriptions**, and choose the target subscription.
2. Select **Resource providers**.
3. Search for `migration`, and then to the right of **Microsoft.DataMigration**, select **Register**.

GO TO SUBSCRIPTIONS

After you register the resource provider, you can create an instance of Azure Database Migration Service.

1. Select **+ Create a resource** and search the marketplace for **Azure Database Migration Service**.
2. Complete the Create Migration Service wizard, then select **Create**.

The service is now ready to migrate the supported source databases to target platforms such as SQL Server, MySQL, PostgreSQL, or MongoDB.

CREATE AN AZURE DATABASE MIGRATION SERVICE INSTANCE

For more information, see:

- Azure Database Migration Service overview
- Create an instance of Azure Database Migration Service
- Azure Migrate in the Azure portal
- Azure portal: create a migration project

Azure App Service Migration Assistant

The Azure App Service Migration Assistant is part of a larger suite of applications that help organizations with their transition to the cloud. The Migration Assistant provides a guided, wizard-like user experience that performs two tasks:

1. It performs an assessment of a specific web app installed on Windows Server by running pre-migration compatibility checks on the web app to determine whether a migration to Azure App Service is possible without modification to the web app.
2. If the assessment proves that the web app can be migrated, the Migration Assistant performs the migration. You'll need to give the Migration Assistant access to your Azure account, select which resource group you want to use, and select a name for the web app, among other details.

Alternatively, the Migration Assistant generates an Azure Resource Manager template that you can use to

migrate the web application in a more automated and repeatable way.

Migrate a web app to Azure App Service

The Migration Assistant begins the migration process by collecting key details from you about your Azure account and then performs the migration.

First, you'll sign in to your Azure account and associate your Migration Assistant session with your account by using a unique code. Next, you'll choose the subscription, the resource group, and the website's domain name. You can choose to create a new Azure App Service plan to host the web app or select an existing plan. The choice affects the geographical region from which your web app is hosted. You'll also have a chance to associate this migration effort with an existing Azure Migrate project. Finally, you can either choose to skip database setup or choose to set up a hybrid connection to enable a database connection.

After the Migration Assistant collects and verifies your selections, it creates the needed Azure App Service resources in the selected region and resource group. It zips up the web app's source files and uses the Azure App Service deployment API to deploy them. Finally, it performs optional migration steps, like helping you set up a hybrid connection.

After a successful migration, you'll need to perform any post-migration tasks. These might include:

- Manually moving application settings and connection strings in your web.config file to Azure App Service.
- Migrating data from an on-premises SQL Server instance to an Azure SQL Database instance.
- Setting up an SSL certificate.
- Setting up custom domain names.
- Setting up permissions in Azure Active Directory.

You might also decide to change the Azure App Service hosting plan and other settings like autoscaling and deployment slots.

For more information, see:

Migrate ASP.NET applications to Azure

Data Migration Assistant

Data Migration Assistant (DMA) helps you upgrade to a modern data platform by detecting compatibility issues that can affect database functionality in your new version of SQL Server or Azure SQL Database. DMA recommends performance and reliability improvements for your target environment and allows you to move your schema, data, and uncontained objects from your source server to your target server.

Data Migration Assistant is integrated with Azure Migrate, allowing you to track all assessment progress in the Azure Migrate dashboard. Launch DMA from Azure Migrate by adding the Azure Migrate: database assessment tool, and add your database assessment to Azure Migrate by selecting the **Upload to Azure Migrate** button in DMA.

> **NOTE**
> For large migrations (in terms of number and size of databases), we recommend that you use Azure Database Migration Service, which can migrate databases at scale.

Start using Data Migration Assistant with these steps:

1. Download and install Data Migration Assistant from the Microsoft download center.
2. Create an assessment by selecting the **New (+)** icon, then select the **Assessment** project type.
3. Set the source and target server type, then select **Create**.
4. Configure the assessment options as required (recommend all defaults).
5. Add the databases to assess.

6. Select **Next** to start the assessment.
7. View results in Data Migration Assistant.

For an enterprise, we recommend following the approach outlined in Assess an enterprise and consolidate assessment reports with DMA to assess multiple servers, combine the reports, and then use provided Power BI reports to analyze the results.

For more information, including detailed usage steps, see:

- Data Migration Assistant overview
- Assess an enterprise and consolidate assessment reports with DMA
- Analyze consolidated assessment reports created by Data Migration Assistant with Power BI

SQL Server Migration Assistant

Microsoft SQL Server Migration Assistant (SSMA) is a tool designed to automate database migration to SQL Server from Microsoft access, DB2, MySQL, Oracle, and SAP ASE. The general concept is to collect, assess, and then review with these tools, however, due to the variances in the process for each of the source systems we recommend reviewing the detailed SQL Server Migration Assistant documentation.

For more information, see:

- SQL Server Migration Assistant overview

Database Experimentation Assistant

Database Experimentation Assistant (DEA) is a new A/B testing solution for SQL Server upgrades. It will assist in evaluating a targeted version of SQL for a given workload. Customers who are upgrading from previous SQL Server versions (SQL Server 2005 and above) to any new version of the SQL Server can use these analysis metrics.

The Database Experimentation Assistant contains the following workflow activities:

- **Capture:** The first step of SQL Server a/B testing is to capture a trace on your source server. The source server usually is the production server.
- **Replay:** The second step of SQL Server a/B testing is to replay the trace file that was captured to your target servers. Then, collect extensive traces from the replays for analysis.
- **Analysis:** The final step is to generate an analysis report by using the replay traces. The analysis report can help you gain insight about the performance implications of the proposed change.

For more information, see:

- Overview of Database Experimentation Assistant

Azure Cosmos DB data migration tool

Azure Cosmos DB data migration tool can import data from various sources into Azure Cosmos DB collections and tables. You can import from JSON files, CSV files, SQL, MongoDB, Azure Table Storage, Amazon DynamoDB, and even Azure Cosmos DB SQL API collections. The data migration tool can also be used when migrating from a single partition collection to a multipartition collection for the SQL API.

For more information, see:

- Azure Cosmos DB data migration tool

Release workloads (test, optimize, and handoff)

3/6/2021 • 3 minutes to read • Edit Online

Now that you have migrated your services to Azure, the next phase includes reviewing the solution for possible areas of optimization. This effort could include reviewing the design of the solution, right-sizing the services, and analyzing costs.

This phase is also an opportunity to optimize your environment and perform possible transformations of the environment. For example, you may have performed a "rehost" migration, and now that your services are running on Azure you can revisit the solutions configuration or consumed services, and possibly perform some "refactoring" to modernize and increase the functionality of your solution.

The remainder of this article focuses on tools for optimizing the migrated workload. When the proper balance between performance and cost has been reached, a workload is ready to be promoted to production. For guidance on promotion options, see the process improvement articles on optimize and promote.

- Right-size assets
- Cost management

All Azure services that provide a consumption-based cost model can be resized through the Azure portal, CLI, or PowerShell. The first step in correctly sizing a service is to review its usage metrics. The Azure Monitor service provides access to these metrics. You may need to configure the collection of the metrics for the service you're analyzing, and allow an appropriate time to collect meaningful data based on your workload patterns.

1. Go to **Monitor**.
2. Select **Metrics** and configure the chart to show the metrics for the service to analyze.

GO TO MONITOR

The following are some common services that you can resize.

Resize a virtual machine

Azure Migrate performs a right-sizing analysis as part of its pre-migration assessment phase, and virtual machines migrated with this tool will likely already be sized based on your pre-migration requirements.

However, for virtual machines created or migrated using other methods, or in cases where your post-migration virtual machine requirements need adjustment, you may want to further refine your virtual machine sizing.

1. Go to **Virtual machines**.
2. Select the desired virtual machine from the list.
3. Select **Size** and the desired new size from the list. You may need to adjust the filters to find the size you need.
4. Select **Resize**.

Resizing production virtual machines can cause service disruptions. Try to apply the correct sizing for your VMs before you promote them to production.

GO TO VIRTUAL MACHINES

- Manage reservations for Azure resources
- Resize a Windows VM
- Resize a Linux virtual machine using Azure CLI

Partners can use the Partner Center to review the usage.

- Azure VM sizing for maximum reservation usage

Resize a storage account

1. Go to **Storage accounts**.
2. Select the desired storage account.
3. Select **Configure** and adjust the properties of the storage account to match your requirements.
4. Select **Save**.

GO TO STORAGE ACCOUNTS

Resize a SQL database

1. Go to either **SQL databases**, or **SQL servers**, then select the server.
2. Select the desired database.
3. Select **Configure** and the desired new service tier size.
4. Select **Apply**.

GO TO SQL DATABASES

Migration-focused cost control mechanisms

3/6/2021 • 7 minutes to read • Edit Online

The cloud introduces a few shifts in how we work, regardless of our role on the technology team. Cost is a great example of this shift. In the past, only finance and IT leadership were concerned with the cost of IT assets (infrastructure, apps, and data). The cloud empowers every member of IT to make and act on decisions that better support the end user. However, with that power comes the responsibility to be cost conscious when making those decisions.

This article introduces the tools that can help make wise cost decisions before, during, and after migrating workloads to Azure.

The tools in this article include:

- Azure Migrate
- Azure pricing calculator
- Azure TCO calculator
- Azure Cost Management + Billing
- Azure Advisor

The processes described in this article may also require a partnership with IT managers, finance, or line-of-business application owners.

- Estimate VM costs prior to migration
- Estimate and optimize VM costs during and after migration
- Tips and tricks to optimize costs

Prior to migration of any asset (infrastructure, app, or data), there is an opportunity to estimate costs and refine sizing based on observed performance criteria for those assets. Estimating costs serves two purposes: it allows for cost control, and it provides a checkpoint to ensure that current budgets account for necessary performance requirements.

Cost calculators

For manual cost calculations, there are two handy calculators that can provide a quick cost estimate based on the architecture of the workload to be migrated.

- The Azure pricing calculator provides cost estimates for the Azure products you select.
- Sometimes decisions require a comparison of the future cloud costs and the current on-premises costs. The total cost of ownership (TCO) calculator can provide such a comparison.

These manual cost calculators can be used on their own to forecast potential spend and savings. They can also be used in conjunction with the cost forecasting tools of Azure Migrate to adjust the cost expectations to fit alternative architectures or performance constraints.

Azure Migrate calculations

Prerequisites: The remainder of this tab assumes the reader has already populated Azure Migrate with a collection of assets (infrastructure, apps, and data) to be migrated. The prior article on assessments provides instructions on collecting the initial data. Once the data is populated, follow the next few steps to estimate monthly costs based on the data collected.

Azure Migrate calculates monthly cost estimates based on data captured by the collector and Service Map. The

following steps will load the cost estimates:

1. Navigate to **Azure Migrate assessment** in the portal.
2. In the project **Overview** page, select **+ Create assessment**.
3. Select **View all** to review the assessment properties.
4. Create the group, and specify a group name.
5. Select the machines that you want to add to the group.
6. Select **Create assessment**, to create the group and the assessment.
7. After the assessment is created, view it in **Overview** > **Dashboard**.
8. In the **Assessment details** section of the portal navigation, select **Cost details**.

The resulting estimate, pictured below, identifies the monthly costs of compute and storage, which often represent the largest portion of cloud costs.

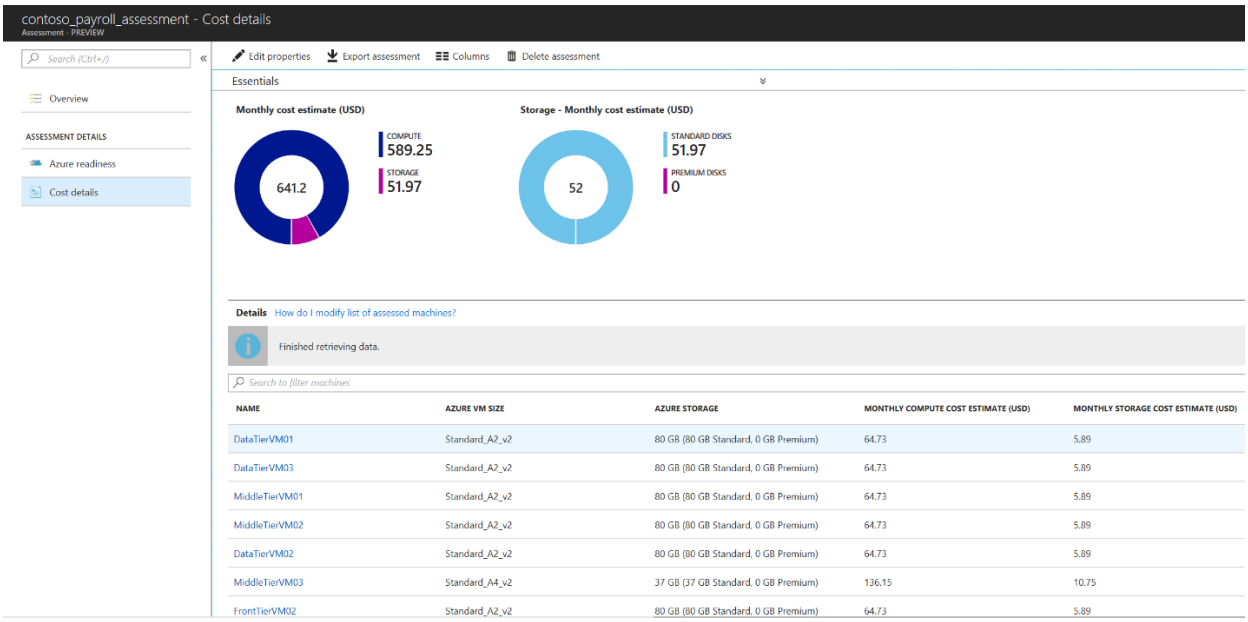


Figure 1: Diagram of the cost details view of an assessment in Azure Migrate.

Additional resources

- Set up and review an assessment with Azure Migrate
- For a more comprehensive plan on cost management across larger numbers of assets (infrastructure, apps, and data), see the Cloud Adoption Framework governance model. In particular, see Cost Management discipline guidance and Cost Management discipline improvement.

Get assistance

3/6/2021 • 2 minutes to read • Edit Online

We know that getting the right support at the right time will accelerate your migration efforts. Review the assistance avenues below to meet your needs.

- Support plans
- Partners

Microsoft Support

Microsoft offers a basic support plan to all Azure customers. You have 24x7 access to billing and subscription support, online self-help, documentation, white papers, and support forums.

If you need help from Microsoft Support while using Azure, follow these steps to create a support request:

1. Select **Help + support** in the Azure portal.
2. Select **New support request** to enter details about your issue and contact support.

1. Select **Help + support**.
2. Select **New support request** to enter details about your issue and contact support.

CREATE A SUPPORT REQUEST

To view your support requests, follow these steps:

1. Select **Help + support** in the Azure portal.
2. Select **All support requests** to view your support requests.

1. Select **Help + support**.
2. Select **All support requests** to view your support requests.

VIEW YOUR SUPPORT REQUESTS

Need support engineer assistance for deep technical guidance?

1. Select **Help + support** in the Azure portal.
2. Select **Support Plans** to review the plans available to you.

1. Select **Help + support**.
2. Select **Support Plans** to review the plans available to you.

REVIEW SUPPORT PLANS

Online communities

The following online communities provide community-based support:

- MSDN forums
- Stack Overflow

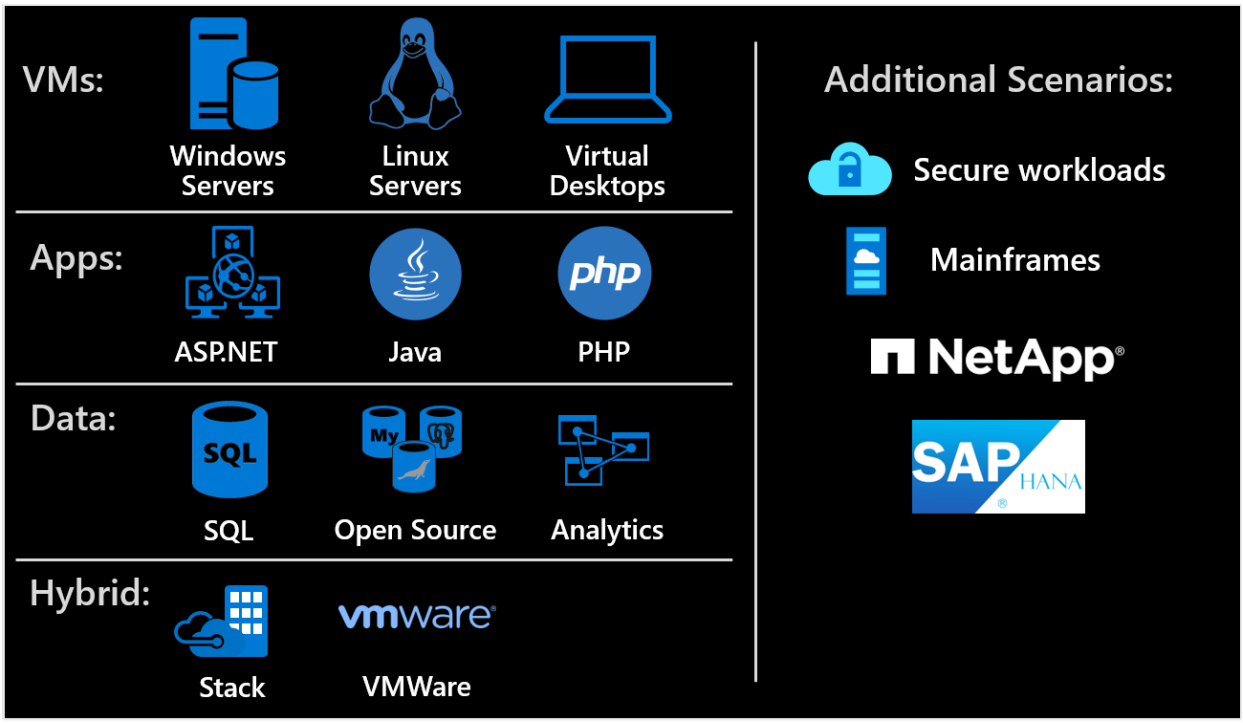
The One Migrate approach to migrating the IT portfolio

6/22/2021 • 2 minutes to read • Edit Online

Azure and Azure Migrate are both well known for hosting Microsoft technologies. But you might not be aware of Azure's ability to support migrations beyond Windows and SQL Server. The *One Migrate* scenarios captured in the Migrate methodology demonstrate the same set of consistent guidelines and processes for migrating both Microsoft and third-party technologies.

Migration scenarios

The following diagram and table outline a number of scenarios that follow the same iterative Migrate methodology for migration and modernization.

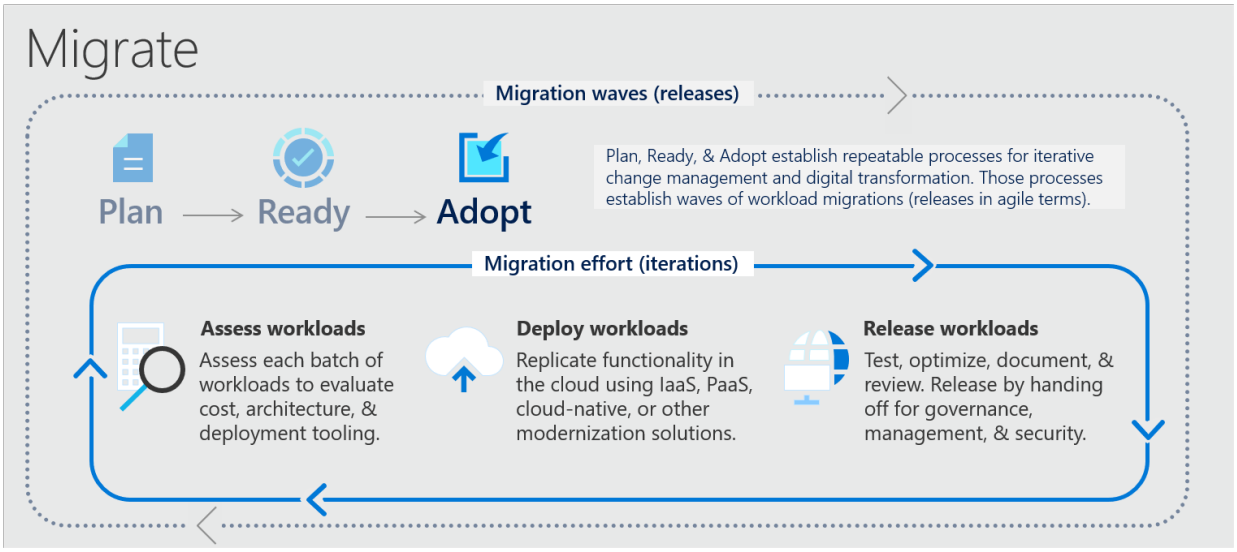


Virtual machines	Virtual machines	Linux servers	Virtual desktops
Applications	ASP.NET	Java	PHP
Data	SQL Server	Open-source databases	Analytics
Hybrid	Azure Stack	VMware	
Technology platforms	SAP (Classic & HANA)	Kubernetes	Mainframes
Additional scenarios	Secure workloads	Multitenant environments	NetApp

Watch this short video for more information about using the right tools to move, manage and optimize your cloud environment.

Migrate methodology

In each of the preceding migration scenarios, the same basic process will guide your efforts as you move your existing workloads to the cloud, as shown here:



In each scenario, you'll structure migration waves to guide the releases of multiple workloads. Establishing a cloud adoption plan and Azure landing zones through the plan and Ready methodologies helps to add structure to your migration waves.

During each iteration, follow the Migrate methodology to assess, deploy, and release workloads. To modify those processes to fit your organization's specific scenario, select any of the migration scenarios listed in the table.

Next steps

If you aren't migrating a specific scenario, start by following the four-step Cloud Adoption Framework migration process.

Overview of application migration examples for Azure

7/21/2021 • 9 minutes to read • Edit Online

This section of the Cloud Adoption Framework for Azure provides examples of several common migration scenarios and demonstrates how you can migrate on-premises infrastructure to Microsoft Azure.

Introduction

Azure provides access to a comprehensive set of cloud services. As developers and IT professionals, you can use these services to build, deploy, and manage applications on a range of tools and frameworks through a global network of datacenters. As your business faces challenges associated with the digital shift, the Azure platform helps you to figure out how to:

- Optimize resources and operations.
- Engage with your customers and employees.
- Transform your products.

The cloud provides advantages for speed and flexibility, minimized costs, performance, and reliability. But many organizations will need to continue to run on-premises datacenters. In response to cloud adoption barriers, Azure provides a hybrid cloud strategy that builds bridges between your on-premises datacenters and the Azure public cloud. An example is using Azure cloud resources like Azure Backup to protect on-premises resources or Azure analytics to gain insights into on-premises workloads.

As part of the hybrid cloud strategy, Azure provides growing solutions for migrating on-premises applications and workloads to the cloud. With simple steps, you can comprehensively assess your on-premises resources to figure out how they'll run in the Azure platform. Then, with a deep assessment in hand, you can confidently migrate resources to Azure. When resources are up and running in Azure, you can optimize them to retain and improve access, flexibility, security, and reliability.

Migration patterns

Strategies for migration to the cloud fall into four broad patterns: rehost, refactor, rearchitect, or rebuild. The strategy you adopt depends on your business drivers and migration goals. You might adopt multiple patterns. For example, you could choose to rehost noncritical applications while rearchitecting applications that are more complex and business-critical. Let's look at these patterns.

PATTERN	DEFINITION	WHEN TO USE

PATTERN	DEFINITION	WHEN TO USE
Rehost	Often referred to as a lift-and-shift migration, this option doesn't require code changes. You can use it to migrate your existing applications to Azure quickly. Each application is migrated as is to reap the benefits of the cloud without the risk and cost associated with code changes.	When you need to move applications quickly to the cloud. When you want to move an application without modifying it. When your applications are designed so that they can take advantage of Azure infrastructure as a service (IaaS) scalability after migration. When applications are important to your business, but you don't need to immediately change application capabilities.
Refactor	Often referred to as "repackaging," refactoring requires minimal changes to applications so that they can connect to Azure platform as a service (PaaS) and use cloud offerings. For example, you could migrate existing applications to Azure App Service or Azure Kubernetes Service (AKS). Or, you could refactor relational and nonrelational databases into options such as Azure SQL Managed Instance, Azure Database for MySQL, Azure Database for PostgreSQL, and Azure Cosmos DB.	If your application can easily be repackaged to work in Azure. If you want to apply innovative DevOps practices provided by Azure, or if you're thinking about DevOps using a container strategy for workloads. For refactoring, you need to think about the portability of your existing code base and available development skills.
Rearchitect	Rearchitecting for migration focuses on modifying and extending application functionality and the code base to optimize the application architecture for cloud scalability. For example, you could break down a monolithic application into a group of microservices that work together and scale easily. You could also rearchitect relational and nonrelational databases to a fully managed database solution, such as SQL Managed Instance, Azure Database for MySQL, Azure Database for PostgreSQL, and Azure Cosmos DB.	When your applications need major revisions to incorporate new capabilities or to work effectively on a cloud platform. When you want to use existing application investments, meet scalability requirements, apply innovative DevOps practices, and minimize use of virtual machines.

PATTERN	DEFINITION	WHEN TO USE
Rebuild	Rebuild takes things a step further by rebuilding an application from scratch using Azure cloud technologies. For example, you could build greenfield applications with cloud-native technologies like Azure Functions, AI, SQL Managed Instance, and Azure Cosmos DB.	When you want rapid development, and existing applications have limited functionality and lifespan. When you're ready to expedite business innovation (including DevOps practices provided by Azure), build new applications using cloud-native technologies, and take advantage of advancements in AI, blockchain, and IoT.

Migration example articles

This section provides examples of several common migration scenarios. Each example includes background information and detailed deployment scenarios that illustrate how to set up a migration infrastructure and assess the suitability of on-premises resources for migration. More articles will be added to this section over time.

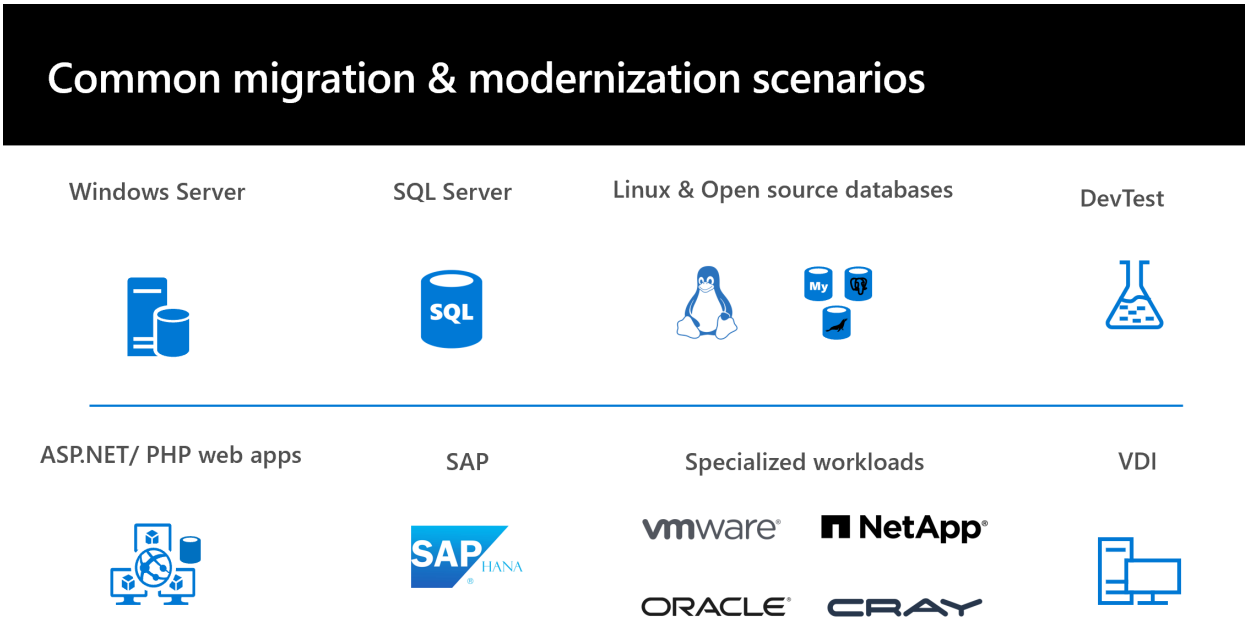


Figure 1: Common migration and modernization project categories.

This series focuses on each migration scenario, driven by slightly different business goals that determine the migration strategy. For each deployment scenario, we provide information about:

- Business drivers and goals.
- A proposed architecture.
- Steps to perform the migration.
- Recommendations for cleanup and next steps after migration is finished.

Assessment

ARTICLE	DETAILS

ARTICLE	DETAILS
Assess on-premises resources for migration to Azure	This best practice article in the Plan methodology discusses how to run an assessment of an on-premises application running on VMware. In the article, an example organization assesses application VMs by using Azure Migrate and the application SQL Server database by using Data Migration Assistant.

Infrastructure

ARTICLE	DETAILS
Deploy Azure infrastructure	This article shows how an organization can prepare its on-premises infrastructure and its Azure infrastructure for migration. The infrastructure example established in this article is referenced in the other samples provided in this section.

Windows Server workloads

ARTICLE	DETAILS
Rehost an application on Azure VMs	This article provides an example of migrating on-premises application VMs to Azure VMs using Azure Migrate.

SQL Server workloads

ARTICLE	DETAILS
Migrate SQL Server databases to Azure	This article demonstrates how the fictional company Contoso assessed, planned, and migrated its various on-premises SQL Server databases to Azure.
Rehost an application on an Azure VM and SQL Managed Instance	This article provides an example of a lift-and-shift migration to Azure for an on-premises application. This process involves migrating the application front-end VM by using Azure Migrate and the application database to SQL Managed Instance by using Azure Database Migration Service.
Rehost an application on Azure VMs using SQL Server Always On availability groups	This example shows how to migrate an application and data by using Azure-hosted SQL Server VMs. It uses Azure Migrate to migrate the application VMs and Database Migration Service to migrate the application database to a SQL Server cluster that's protected by an Always On availability group.

Linux and open-source databases

ARTICLE	DETAILS
Migrate open-source databases to Azure	This article demonstrates how the fictional company Contoso assessed, planned, and migrated its various on-premises open-source databases to Azure.

ARTICLE	DETAILS
Migrate MySQL to Azure	This article demonstrates how the fictional company Contoso planned and migrated its on-premises MySQL open-source database platform to Azure.
Migrate PostgreSQL to Azure	This article demonstrates how the fictional company Contoso planned and migrated its on-premises PostgreSQL open-source database platform to Azure.
Migrate MariaDB to Azure	This article demonstrates how the fictional company Contoso planned and migrated its on-premises MariaDB open-source database platform to Azure.
Rehost a Linux application on Azure VMs and Azure Database for MySQL	This article provides an example of migrating a Linux-hosted application to Azure VMs by using Azure Migrate. The application database is migrated to Azure Database for MySQL by using Database Migration Service.
Rehost a Linux application on Azure VMs	This example shows how to complete a lift-and-shift migration of a Linux-based application to Azure VMs by using Azure Migrate.

Dev/test workloads

ARTICLE	DETAILS
Migrate dev/test environments to Azure IaaS	This article demonstrates how Contoso rehosts its dev/test environment for two applications running on VMware VMs by migrating to Azure VMs.
Migrate to Azure DevTest Labs	This article discusses how Contoso moves its dev/test workloads to Azure by using DevTest Labs.

ASP.NET and PHP web apps

ARTICLE	DETAILS
Refactor a Windows application using App Service and SQL Database	This example shows how to migrate an on-premises Windows-based application to an Azure web app and migrate the application database to an Azure SQL Database server instance by using Database Migration Service.
Refactor a Windows application using App Service and SQL Managed Instance	This example shows how to migrate an on-premises Windows-based application to an Azure web app and migrate the application database to SQL Managed Instance by using Database Migration Service.
Refactor a Linux application to multiple regions using App Service, Azure Traffic Manager, and Azure Database for MySQL	This example shows how to migrate an on-premises Linux-based application to an Azure web app on multiple Azure regions by using Traffic Manager to integrate with GitHub for continuous delivery. The application database is migrated to an Azure Database for MySQL instance.

ARTICLE	DETAILS
Rebuild an application in Azure	This article provides an example of rebuilding an on-premises application by using a range of Azure capabilities and managed services. These capabilities and services include App Service, AKS, Azure Functions, Azure Cognitive Services, and Azure Cosmos DB.
Refactor Team Foundation Server to Azure DevOps Services	This article shows an example migration of an on-premises Team Foundation Server deployment to Azure DevOps Services in Azure.

SAP

ARTICLE	DETAILS
SAP migration guide	Get practical guidance to move your on-premises SAP workloads to the cloud.
Migrate SAP applications to Azure	White paper and roadmap for your SAP journey to the cloud.
Migration methodologies for SAP on Azure	Overview of various migration options to move SAP applications to Azure.

Specialized workloads

ARTICLE	DETAILS
Move on-premises VMware infrastructure to Azure	This article provides an example of moving on-premises VMware VMs to Azure by using Azure VMware Solution.
Azure NetApp Files	Enterprise file storage powered by NetApp. Run Linux and Windows file workloads in Azure.
Oracle on Azure	Run your Oracle databases and enterprise applications in Azure and Oracle Cloud Infrastructure.
Cray in Azure	High-performance computing with Cray in Azure. A dedicated supercomputer on your virtual network.

VDI

ARTICLE	DETAILS
Move on-premises Remote Desktop Services to Azure Virtual Desktop	This article shows how to migrate on-premises Remote Desktop Services to Azure Virtual Desktop.

Migration scaling

ARTICLE	DETAILS
Scale a migration to Azure	This article shows how an example organization prepares to scale to a full migration to Azure.

Demo applications

The example articles provided in this section use two demo applications: SmartHotel360 and osTicket.

SmartHotel360: This test application was developed by Microsoft to use when you work with Azure. It's provided under an open-source license, and you can download it from GitHub. It's an ASP.NET application connected to a SQL Server database. In the scenarios discussed in these articles, the current version of this application is deployed to two VMware VMs running Windows Server 2008 R2 and SQL Server 2008 R2. These application VMs are hosted on-premises and managed by vCenter Server.

osTicket: This open-source service desk ticketing application runs on Linux. You can download it from GitHub. In the scenarios discussed in these articles, the current version of this application is deployed on-premises to two VMware VMs running Ubuntu 16.04 LTS using Apache 2, PHP 7.0, and MySQL 5.7.

Assessment and migration planning with Azure Migrate for servers

6/3/2021 • 16 minutes to read • Edit Online

Migration of workloads to Azure can be a challenging project. Preparing well for a migration project helps drive success and reduces the complexity of the project.

This article shows how the fictional company Tailwind Traders uses Azure Migrate to plan for migration. Tailwind does discovery and assessment of their existing environment, which is targeted for migration to Azure.

When using this example to help plan your own infrastructure migration efforts, keep in mind that the provided sample plan and deployment is specific to Tailwind. Review your organization's business needs, structure, and technical requirements when making important infrastructure migration decisions.

Whether you need all the elements described in this article depends on your migration strategy and needs. For example, you might find it's better to discover workload dependencies using existing network monitoring tools or get recommended VM sizes using Movere or other third-party tools.

Overview

In order for Tailwind to prepare for migration to Azure, it's critical to plan for the right set of discovery and assessment tools and activities. Deploying and running discovery tools enables Tailwind to gather critical information on the environment targeted for migration to Azure.

The information gathered will enable Tailwind to:

- Build an up-to-date inventory (servers and workloads) of their datacenter environments.
- Determine server readiness for migration to Azure.
- Perform an assessment to get recommended sizing and storage for Azure VMs.
- Get cost estimation for Azure VMs (compute and storage).
- Analyze server and workload dependencies.
- Assess application compatibility with Azure.
- Assess migration options for existing SQL Server instances.
- Assess options for migrating applications to Azure App Service.
- Plan for migration waves.

For successful migration planning, Tailwind needs to think about following areas:

- **Step 1: Set up tooling for discovery and assessment.** How many discovery appliances or agents will you need? What are best practices to run discovery of your environments? Do I need dependency mapping and application discovery?
- **Step 2: Perform assessment.** How do I select parameters for assessment? How do I know whether my servers are ready for migration to Azure? How do I estimate costs of migrated servers?
- **Step 3: Dependency analysis and migration planning.** Which servers are used for a specific application workload? What other components does the workload depend on? What are the dependencies between servers and its workloads?
- **Step 4: Plan for migration waves.** Are my servers ready for migration to Azure? How do I prioritize and group servers and associated workloads across migration waves?

Before you start

Before diving deep into infrastructure discovery and assessment, consider reading some background information relevant to Azure Migrate discovery and assessment:

- Review the Azure Migrate: Discovery and assessment tool overview.
- Review the deployment scenarios and requirements for the Azure Migrate appliance that performs discovery, Azure Migrate appliance.
- Review the Azure Migrate assessment overview.
- Optionally, complete the Microsoft Learn training, Migrate virtual machines and applications using Azure Migrate.

Step 1: Set up tooling for discovery and assessment

Tailwind first needs to determine the Azure Migrate appliances needed for discovery and assessment.

Discovery tooling: Planning and implementation

To start gathering data required for assessment, Tailwind needs to prepare to deploy the appliances and agents needed for discovery of their on-premises server infrastructure.

Using the following workflow, Tailwind can define the server discovery tools required to gather information from on-premises servers and their dependencies.

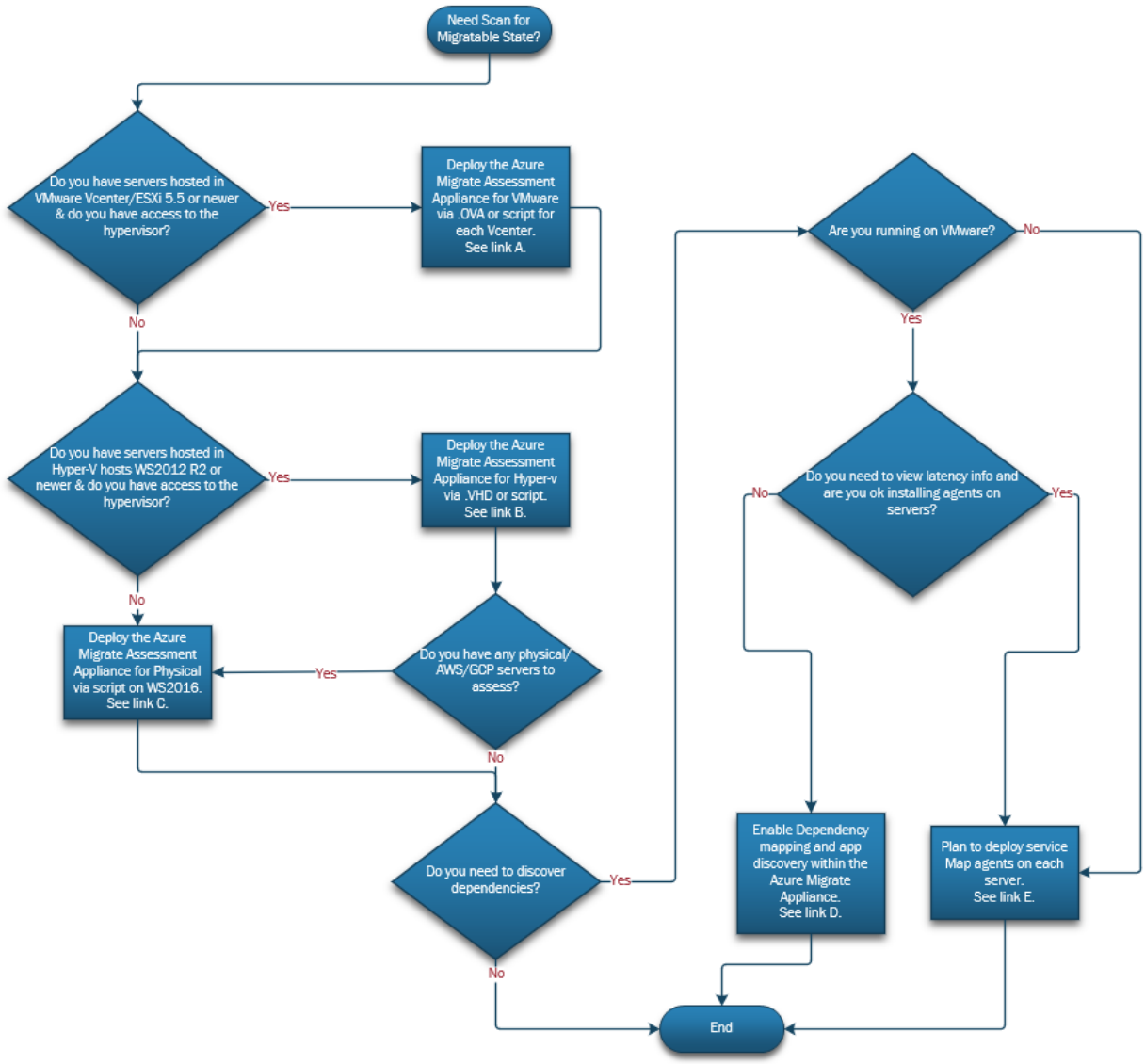


Figure 1: Discovery appliances and agents workflow.

For more information on Azure Migrate, see:

- Discover VMware VMs
- Discover Hyper-V VMs
- Other VMs:
  - Discover physical or other hypervisors
  - Discover AWS VMs
  - Discover GCP VMs
- Discovery:
  - Dependency analysis via Azure Migrate (agentless)
  - Application discovery
  - SQL Server discovery
- Dependency analysis via Service Map (agent-based)

> **NOTE**
> You can discover up to 10,000 servers in a VMware environment, up to 5,000 servers in a Hyper-V environment, and up to 1,000 physical servers by using a single appliance. If your environment exceeds those limits, read about scaling a Hyper-V assessment, scaling a VMware assessment, or scaling a physical server assessment.

Based on the preceding workflow and following relevant Azure Migrate documentation, Tailwind deploys the required appliances or agents for discovery. During deployment, Tailwind must provide server credentials to perform software inventory (installed applications, roles, and features), agentless dependency analysis, agent-based dependency analysis, and discovery of SQL Server instances and databases. Tailwind's migration team will collaborate with the server infrastructure and SQL Server database administrator teams to prepare required credentials for discovery. After deployment and configuration, Tailwind regularly verifies that data is being collected by the appliance and visible in the Azure Migrate project.

As a best practice, Tailwind gathers data over a prolonged period of time, such as five weeks. The goal is to gather server performance data and dependencies over a period of time, which ensures collection of metrics about monthly infrastructure consumption spikes, such as month end transactions (payroll application), weekly batch processes, and data import or export.

Step 2: Perform assessment

After discovery of the on-premises servers has been running for at least a day, Tailwind can start creating assessments in the Azure Migrate project through the Azure portal. Their plan will be to create an assessment per environment (production and nonproduction).

Tailwind can choose to run various assessments such as a Azure VM assessment, Azure SQL Database assessment, or AVS assessment. Tailwind starts with the VM assessment by following the assessment best practices and executes an Azure SQL Database assessment later. As a discovery source, Tailwind chooses servers discovered through the Azure Migrate appliance.

Tailwind understands that the Azure Migrate appliance will continue to collect performance data of their on-premises infrastructure and send it over to Azure Migrate, therefore Tailwind has the opportunity to recalculate Azure Migrate assessment at any future point in time to ensure the latest data sets are being included as migration execution approximates.

> **NOTE**
> For server infrastructure that cannot be discovered via the Azure Migrate appliance, Tailwind will create a CSV file with the required data and upload it to Azure Migrate. For more information, see Assess servers using an imported CSV file.

Assessment properties

Tailwind plans to tune their assessments and plans to specify custom parameters. For both production and nonproduction assessments, it specifies its primary Azure region as the target location, as well as same sizing criteria, comfort factor, pricing, and licensing. However, in terms of compute and storage, Tailwind plans to have VM SKUs and storage types specified differently in the assessments.

For production, Tailwind clears the VM SKU options for A-series and sets the storage type to Standard SSD and Premium SSD-managed disks. For nonproduction, Tailwind ensures all VM-series is select and storage type is set to automatic.

Tailwind plans to migrate to Azure and observe behavior for at least a few months before making purchase arrangements for reserved instances. If Tailwind wants to purchase reserved instances in advance, then for the assessments, after discussing with business on selecting right reserved capacity (compute) option, Tailwind will select the VM-series or specific VM SKUs, which are within the reserved instances purchased scope.

Tailwind selects performance-based sizing criteria and sets performance history to one month. This configuration will enable Tailwind to find recommended VM sizes and managed disks based on actual server requirements that were gathered during the discovery phase. Thus, resulting in optimization of infrastructure as a result of the migration. Tailwind understands that Azure Migrate assessment will recommend VM SKUs based on CPU and memory utilization. For disk types, Azure Migrate will base the recommendations on IOPS and throughput of the existing disks in the on-premises servers assessed.

Further, as resource usage for existing workloads is expected to increase slightly over time, Tailwind sets the comfort factor to **1.1x**. This configuration will add 10 percent more resources to VM requirements based on the gathered performance data, and recommended VM sizes will include this 10 percent buffer.

Based on the Azure offer associated with Tailwind's subscription, the appropriate offer or licensing program and currency is selected. Tailwind is eligible for Azure Hybrid Benefit, so it sets **Already have a Windows Server license** to **yes**. Tailwind understands that not configuring Azure Hybrid Benefit might result in extra billing of Windows Server licenses that were already purchased. Tailwind also understands that Azure Hybrid Benefit primarily applies to Windows operating systems, so for any servers running Linux or other non-Windows operating systems, Tailwind should consult on license portability separately.

After saving assessment properties, Tailwind needs to select servers to assess. Tailwind creates a new server group that includes all discovered servers from all discovery appliances and creates assessments for each specific environment. Although Tailwind considered creating a server group and assessment for each specific application, while the ideal scenario, it was decided to create server groups at a later stage, closer to wave migration planning, as more time is needed to analyze dependencies between servers.

> **NOTE**
> Tailwind can create multiple assessments. Each assessment can have different properties and a different group of servers. Multiple assessments can be run against the same set of servers or groups. All assessments are stored in the Azure Migrate project. This allows Tailwind to compare different assessments and decide what migration options are best for their needs and business case (such as VM sizing, storage type, and reserved capacity) and estimate the monthly cost.

Azure readiness

Azure Migrate can calculate assessment results in a few minutes after creating the assessment. Once the assessment results are available, Tailwind reviews Azure readiness for all servers that were assessed. Azure

Migrate provides visualization that enables Tailwind to drill down on servers readiness details for each server. Assessment details include recommended VM SKU, storage type, networking configuration, and most importantly, Azure readiness.

Tailwind spends some time reviewing the reasons why some servers are conditionally ready for migration to Azure and why some servers aren't. Tailwind refers to the list of assessment readiness issues, which provides insights on the steps needed to mitigate Azure readiness issues. After a thorough review, Tailwind prepares a plan of action to fix readiness issues where possible (upgrading the OS, changing the boot type, removing unused disks, and so on). For servers with issues that can't be easily mitigated, Tailwind considers other migration approaches, such as whether to replatform, modernize, replace, or retire.

> **NOTE**
>
> Although the server is marked as **Ready** and Azure infrastructure will support the necessary configuration, the replication appliance or agents may not always support the replication of the server (such as Windows Server Failover Clustering with shared disks). It's also important to review the Azure Migrate appliance (VMware agentless), replication appliance (VMware agent-based, physical and other clouds), and replication agents (Hyper-V) supported VM configurations. If an unsupported VM configuration is identified and confirmed, consider a different migration approach such as redeployment of the application or database in a new Azure VM.

Database assessment

Tailwind is running SQL Server in their on-premises environment. As part of the migration project to Azure, Tailwind would like to understand extra options and modernization approaches for migrating SQL Server workloads to Azure SQL Database or SQL Managed Instance. Although Tailwind's primary migration approach is rehosting to Azure VMs via server replication using Azure Migrate, Tailwind would like to explore what modernization is possible given their current server configurations. This stage is exploratory for Tailwind, in order to keep focus on the current migration strategy.

Azure Migrate supports creating assessments for migration to Azure SQL Database and SQL Managed Instance, which Tailwind finds as insightful assessments to create. For the target location, Tailwind chooses their primary Azure region and for target deployment type Tailwind selects **Recommended** to determine whether Azure SQL Database or SQL Managed Instance is a better fit for their databases. Tailwind set the comfort factor to **1.1x** to account for planned future workload growth. In the pricing section Tailwind chooses options that are aligned with their Azure offer and licensing programs. Tailwind reuses the prior groups created and runs assessment for this group.

Azure cost details

Azure Migrate assessment results include cost estimation for assessed servers. Tailwind can estimate total compute and storage costs for migration, or drill down for compute and storage cost estimation per individual server. Tailwind understands that after migration to Azure, the server might incur other costs related to networking, monitoring, backup, and so on. Those extra costs are not part of the assessment cost details. For these cost details scenarios, Tailwind resorts to using the Azure calculator.

Step 3: Dependency analysis and migration planning

To perform a successful migration to Azure, it's critical for Tailwind to understand workload details in the context of their dependencies during migration planning. As a best practice, Tailwind migration team needs to identify application owners of the workloads running on each of the server assessed for the migration. Tailwind uses their CMDB data to link discovered servers to various business applications and identify respective applications owners, their development, and support teams. After application owners are identified, Tailwind migration team schedules review meetings (interviews) with them.

The migration team is now better prepared to group servers for workload-specific assessment based on the application tiers and initial dependencies given the application owner interviews and analysis.

Dependency analysis

To further validate server groups and plan for migration waves, Tailwind needs to understand dependencies between workloads and its servers. Tailwind uses the results of the dependency analysis, which ran during the assessment phase. Tailwind exports dependency results and uses Power BI to visualize network connections.

To perform effective dependency analysis, Tailwind takes the following steps:

1. Clean up the dependency mapping scan to show unique dependencies based on a one-month period of time by using the PowerShell script as described in Analyze server dependencies (agentless). This script works as-is for agentless dependency mapping. For agent-based dependency mapping, KQL queries must be created in order to perform the same task. With agent-based dependency mapping, the longest period of time available is one hour.
2. Filter out the noise from the dependency mapping scan by removing well-known dependencies such as:
   - Known system processes
   - Known internet-based endpoints (non-RFC 1918)
   - Known internal endpoints (RFC 1918) such as domain controllers, file servers, network appliances such as load balancers, and management infrastructure (such as SCCM, System Center Operations Manager, backup, and antimalware).
3. Integrate dependency scan into visualization tooling like Power BI. Helpful visualizations include:
   - Sankey diagram
   - Network Navigator chart
   - Tassels Parallel Sets Slicer

Review data collected with application owners

As preparation continues for the migration, Tailwind realizes that extra workload details must be gathered from application owners. Based on those details, the migration team reviews complexity, criticality, and dependencies of each server group or workload, and then finalizes the prioritization of server groups for their migration wave schedules. As Tailwind schedules extra interviews with application owners, it ensures collection of the following data points:

- Business criticality
- Workload architecture (all servers and other components such as load balancers that are required for workload to run)
- Number and type of users (internal, external)
- Data classification
- SLA requirements
- Downtime window
- Backup mechanism
- Monitor mechanism
- Patching mechanism
- Licensing mechanism (OS and software, dependency on MAC/IP address)
- Firewall rules (host and network): Ports and protocols
- Effect of changing the server's IP address

As a best practice, Tailwind reviews server assessment results with application owners. During the review, Tailwind will:

- Validate assessment sizing recommendations (such as VM size and storage configuration).
- Identify any components that need optimization or replacement in Azure (such as load balancers or clusters).
- Verify all dependencies are discovered and validated with application owners.

- Confirm SLA requirements are met.
- Review SQL assessment results and review options for rehosting your SQL Server databases on Azure SQL Database or SQL Managed Instance.
- Review cost estimates.

Document the migration plan

Based on the final assessment results and application owner interviews, the Tailwind migration team finalizes the migration approach. They document the migration steps including details from assessment, dependency analysis, and inputs from the application owner interviews. This captures extra configurations like load balancers, DNS updates, database connection strings, licenses, and any extra steps the application team or vendor might need to execute as part of pre-migration and post-migration steps.

Step 4: Plan for migration waves

Tailwind now needs to finalize prioritization of the migration waves based on workload criticality, dependencies, business requirements, and so on.

Server grouping

Based on the interviews and dependency mapping data, Tailwind continues to refine their server groups based on the application architecture (multitier) and server dependencies. Tailwind's initial server groups are as follows:

- Server group 1 (LOB application 1): Five servers
- Server group 2 (LOB application 2): 30 servers
- Server group 3 (LOB application 3): 50 servers

Migration wave planning

Tailwind plans to start with smaller migration waves, which include standalone servers and workloads that aren't business critical. Starting with single-instance servers as part of the initial migration wave, it will be easier to resolve any issues, verify migration process end-to-end, and build confidence in the migration strategy. During subsequent migration waves, Tailwind plans to include multiple nonproduction workloads. Once finalized with nonproduction, Tailwind's migration waves will include production workloads. The most complex production workloads will be left until the end within the last migration wave schedules.

A migration wave can be a single large business application running on multiple web, application, and database servers or a few smaller, independent workloads. In tightly coupled environments, server groups might need to be split into different migration waves, such as a server group of 100+ servers. Since it contains shared servers across multiple applications (such as shared databases, shared web servers, or shared file servers) the number of interdependent servers is large.

In the scenario where a server group (interdependent servers) has to be split and not all servers are migrated in a single wave, Tailwind will:

- Identify which connections are the most latency-sensitive connections to dependencies and prioritize these dependencies for the migration wave.
- Reschedule the left-out servers to be migrated on the next immediate migration wave or as part of an interim migration wave.

Tailwind reviews the guidance in Build a migration plan with Azure Migrate to prioritize server groups and workloads when assigning them to migration waves. These best practices include:

- Start with server groups with the least number of servers.
- Start with server groups based on lowest business criticality. For example:
  - Prioritize environments with lower business impact (nonproduction vs. production).

- Prioritize workloads with lower SLA and highest RTO/RPO.
- Prioritize workloads with less restrictive data classification.
- Prioritize workloads with newer operating systems, since older operating systems tend to have more issues booting in Azure or in supporting Azure management extensions.

Tailwind understands some migration waves will change migration execution progresses, so the migration team reviews and modifies waves periodically. The migration team incorporates learnings from executed waves to improve assessment and migration planning, such as capturing any missing network requirements, performing pre-migration testing, implementing post-migration tweaks, and reviewing permissions.

Conclusion

In this article, Tailwind sets up Azure Migrate tools for discovery and assessment and executes steps required to build a migration plan, server groups, and prioritization of server groups in migration waves.

Not every step taken here is required for building a migration plan. However, as good migration planning is often key to successful migration project, Tailwind invests time in developing a proper migration plan by executing all steps as described above. It's also important to review and continuously improve the process based on previous migration wave and feedback from key stakeholders teams.

Next steps

As a next step, Tailwind will begin deploying the infrastructure required to execute migration. To review best practices for migration execution, see Plan and deploy Azure Migrate for servers.

Migration execution with Azure Migrate for servers

6/15/2021 • 11 minutes to read • Edit Online

This article shows how the fictional company Tailwind Traders prepares its deployment plan for on-premises infrastructure migration using the server replication features of Azure Migrate.

When you use this example to help plan your own infrastructure migration efforts, keep in mind that the provided sample plan and deployment is specific to Tailwind. Review your organization's business needs, structure, and technical requirements when making important infrastructure migration decisions.

Whether you need all the elements described in this article depends on your migration strategy and needs. For example, you might prefer to redeploy a VM in Azure and redeploy the application instead of replicating the on-premises server to Azure.

Overview

In order for Tailwind to migrate to Azure, it's critical to plan for the migration toolset and activities. Generally, Tailwind needs to think about 3 areas:

- **Step 1: Tooling and replication.** How many replication appliances will you need? What are best practices to enable replication?
- **Step 2: Migration cutover preparation.** What activities will resources need to execute before and after the migration? What value is provided, and when should I plan for a test migration? What are the constraints when testing in an isolated VNet?
- **Step 3: Migration cutover execution.** How does cutover look like? What happens after cutover?

Before you start

Before we start diving deep in infrastructure migration planning and deployment, consider reading some background information relevant to Azure Migrate server replication:

- Review the Azure Migrate: Server Migration tool overview.
- Review the differences between the Azure Migrate appliance, replication appliance, and Hyper-V migration architecture.
- Optionally, complete the Microsoft Learn training, Migrate virtual machines and applications using Azure Migrate training.

Step 1: Tooling and replication

Tailwind needs to determine considerations on the number of appliances needed for replication and replication traffic impact.

Capacity planning for core quotas

In order to proactively ensure that the target migration Azure subscriptions will be able to host the virtual machines created during test migration or production migration, subscriptions core quotas need to be available for the target VM SKUs. Ensure subscription quotas for specific VM SKUs have been increased for the specific target region.

Server migration tooling planning and implementation

After increasing the subscription core quotas, Tailwind needs to prepare to deploy the appliances or agents

needed for replication of their on-premises server infrastructure.

Using the following workflow, Tailwind can identify the server migration tools required to enable replication of on-premises servers. This will aid Tailwind in proactively identifying necessary infrastructure requests to successfully enable replication of their migratable estate.

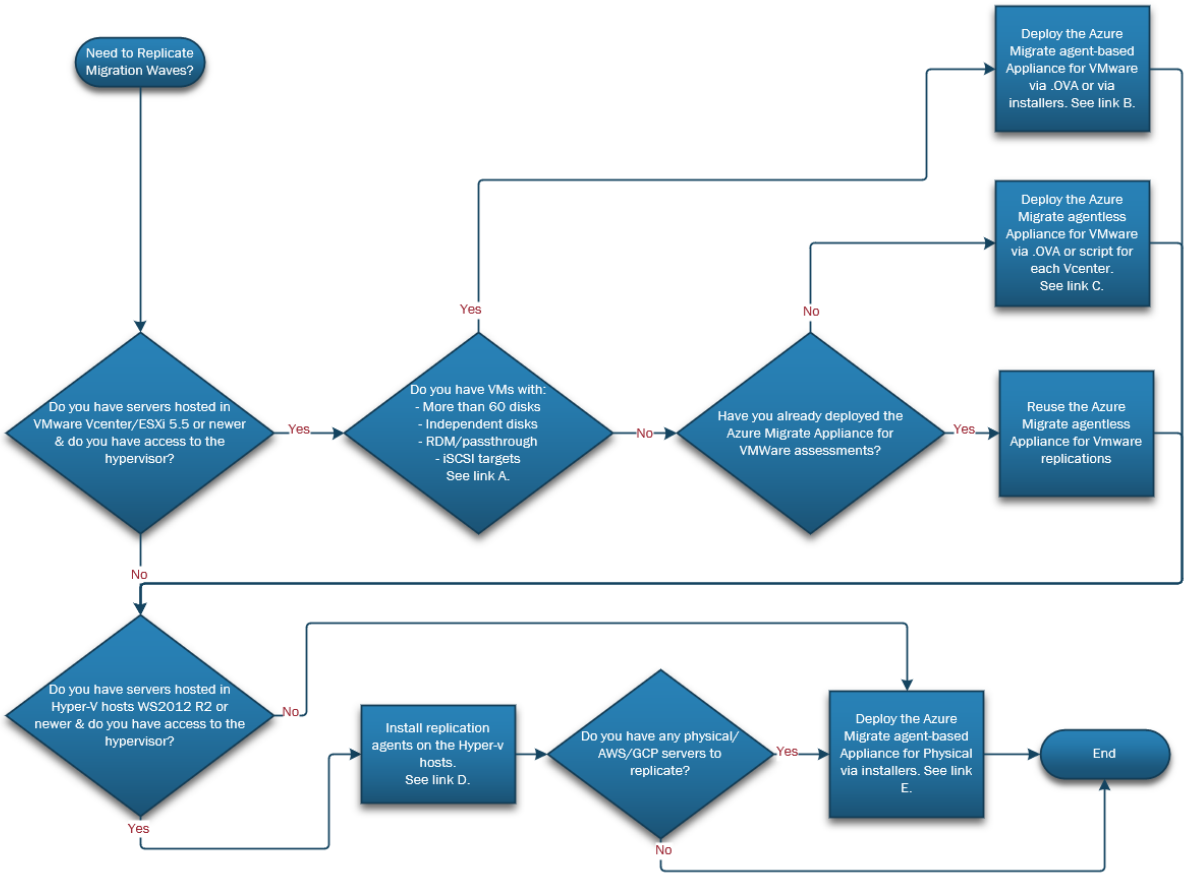


Figure 1: Replication appliances and agents workflow.

For more information, see these Azure Migrate topics:

- Support matrix for VMware migration.
- Migrate VMware VMs to Azure (agent-based).
- Migrate VMware VMs to Azure (agentless) via PowerShell.
- Migrate Hyper-V VMs to Azure.
- Other hypervisors:
  - Migrate machines as physical servers to Azure
  - Discover, assess, and migrate Amazon Web Services VMs to Azure
  - Discover, assess, and migrate Google Cloud Platform VMs to Azure

As a best practice, Tailwind works closely with their virtualization administrators to ensure careful monitoring of key performance counters for CPU, memory and, disk space of the deployed appliances and hypervisor hosts. This ensures that virtualization infrastructure has enough resources to handle additional load from replication appliances and agents.

Replication

Enable and monitor replication

With replication appliances and agents configured, Tailwind can start planning the replication of their on-premises servers. A collection of applications and their dependencies which must be migrated during the same time window is commonly referred to as a migration wave or migration group. Tailwind will use the term *migration wave* to maintain consistency across planning activities.

As a best practice, Tailwind plans to enable initial replication for only a subset of their migration waves, cautious about their available bandwidth. Tailwind understands the initial replication is a full copy of the servers and consumes more bandwidth versus ongoing (delta) replications. Given Tailwind's bandwidth constraint considerations, Tailwind will enable replication for just migration waves that are only a few weeks from test migration and cutover dates.

Tailwind will also monitor initial and ongoing replication closely to ensure healthy and stable replication before enabling replication for additional servers. If errors or warnings arise, Tailwind can proactively detect and act on it before test migration or cutover dates.

Replication tuning

Based on the observed initial and ongoing replications bandwidth patterns, Tailwind will tune their replication strategy based on the following queries:

- How much bandwidth is needed and available for replications?
- How many VMs, on average, can be initially replicated at the same time?
- How many VMs, on average, can be left replicating (delta replication) at the same time?
- Is there an need to throttle replication within the replication appliances or agents?

Step 2: Migration cutover preparation

Given the success in replication toolset deployment and planning for a subset of their migration waves, Tailwind decides to start planning their testing needs and pre-migration and post-migration activities.

Tailwind understands that the migrations are an orchestration of both business and technical groups. Therefore, the following activities are defined as pre-migration and post-migration activities.

Business

In order to prepare the business and its stakeholders for the migration activities, Tailwind defines the following items:

- A maintenance window for each of the applications to migrate.
- Communications on application downtime and impact to business.
- Points of contact (POCs) that can provide support for the following key areas during migration testing and cutover:
  - Network administrators
  - Backup administrators
  - Server administrators
  - Identity administrators
  - Application owners (frontend and backend)
  - Microsoft Support
  - Partner (if available)
- A soak period after the cutover of the application to Azure. During the soak period, if any issues arise then the rollback plan must be executed. After the soak period has expired, rollback of the application cannot be performed.

Technical: Pre-migration

In order to plan for best practice pre-migration activities, Tailwind defines the following activities to be executed prior to a migration failover:

- A rollback plan.
- Latest backup of the servers.
- Opening of firewall prefixes, ports and protocols for necessary traffic between on-premises to Azure and within Azure VNets and subnets.

- Obtain local administrator credentials or keys for server sign-in purposes.
- Review manual changes needed for Windows and Linux
  - For legacy Linux distributions, instructions to install Hyper-V drivers are available in Supported Linux and FreeBSD virtual machines for Hyper-V on Windows.
  - For legacy Windows versions such as Windows Server 2003 or Windows Server 2008, instructions to install Hyper-V drivers are available in Prepare Windows Server 2003 machines for migration.
- Prepare isolated virtual networks for test migrations.
  - Plan for secure management access via RDP or SSH into this environment, using services like Azure Bastion.
  - Plan for an isolated virtual network in each subscription containing the migrated VMs. The test migration functionality in Azure Migrate must use a virtual network in the same subscription where the migrated VM will reside.

Technical: Post-migration

Further, Tailwind defines the following activities to be executed after the migration failover:

- Review the documented Azure Migrate post-migration activities based on the source environments:
  - VMware agentless
  - VMware agent-based
  - Hyper-V
  - Physical
  - AWS
  - GCP
- In addition, Tailwind adds the following post-migration activities as best practices:
  - Validate sign-in with local credentials or keys for RDP or SSH.
  - Verify that DNS resolves and DNS servers are configured in TCP/IP network settings for the OS.
  - Verify that an IP address has been assigned to the server via DHCP in TCP/IP network settings for the OS.
  - Verify that access to OS licensing is activated and there is access to cloud-based licensing endpoints (such as Azure endpoints for key management services).
  - Validate sign-in with domain credentials.
  - Verify that the application has access to dependencies (such as target URLs or connection strings).
  - Verify installation or update required Azure agents:
    - Azure VM agents for Windows or Linux.
    - Log Analytics agent for Windows or Linux.
    - Service Map agent for Windows or Linux.
    - SQL Server IaaS Agent extension.
  - Validate VM monitoring via a new or existing service.
  - Validate VM patching via a new or existing service.
  - Validate VM backup via a new or existing service.
  - Validate VM antivirus and endpoint protection via a new or existing service.
  - Tag Azure resources.
  - Update any existing configuration management database (CMDB).
  - Conduct a post-mortem and document learnings.

Test migration and actual migration

Moving forward, Tailwind now looks to understand the need for a test migration, what test cases make sense, and which virtual networks to use as targets for the test migration and actual migration.

Define a smoke test

As a first step, Tailwind realizes there's a need to perform a smoke test to validate that servers identified to migrate will boot in Azure. You should perform this smoke test in an isolated VNet for all servers to be migrated. Tailwind will follow this recommendation and is especially focused on the smoke test for servers which are legacy, highly customized, or contain hardened operating systems. Additionally, Tailwind is also eager to run a smoke test for servers that have been marked as **Conditionally Ready** by their Azure Migrate assessments.

Tailwind defines a smoke test to be successful when basic server functionality and properties are validated. For example, smoke testing may include:

- The server boots in Azure.
- The administrator can sign in to the server using local credentials.
- TCP/IP settings for DNS, IPv4 and default gateways assignment are updated to the values provided by the Azure VNet via DHCP.
- OS licensing is activated.

Typically this test is led by the server administrators or the migration partner.

Define user acceptance testing

As a second step, Tailwind now looks to perform user acceptance testing (UAT) to ensure that the applications are functional and accessible by expected users. Tailwind is aware that UAT will help find missed configuration changes necessary for a successful migration which may include hardcoded IP addresses.

Tailwind defines UAT to be successful when application functionality and access to dependencies is validated. For example, UAT may include:

- Validate sign-in with domain credentials.
- Verify that the application has access to dependencies (such as target URLs or connection strings).
- Validate application functionality with test users.

Typically, this is usually led by application owners.

Identify testing and migration workflow

Now that test cases have been defined, Tailwind develops the following workflow to encompass the various scenarios it might encounter based on the needs of each application or server.

The majority of Tailwind's scenarios require the second and fifth paths in the following workflow. Tailwind has many legacy servers and other servers marked as **Ready with Conditions**, which might not boot in Azure. Therefore, Tailwind will test each of those servers in an isolated VNet to ensure that each server passes the smoke test. For this, Tailwind will perform a test migration in Azure Migrate which allows for the option for automated clean up of created resources such as VMs and NICs.

Further, Tailwind's environment is tightly coupled which means there's a large amount of servers which are interdependent with one another, thus resulting in large migration waves. Tailwind has decided to split their large migration waves and migrate servers together which have the most strict latency requirements. As a result, some application dependencies must remain on-premises for a given migration wave. Tailwind determines that it should migrate directly into the production VNet, since that network already has connectivity back to their on-premises dependencies. In this path, Tailwind will perform necessary smoke tests in an isolated VNet and perform UAT in the production VNet. If successful, Tailwind will conclude the migration as a final cutover for the servers.

Nonetheless, Tailwind does find value in considering the remainder paths only for scenarios where it's possible to migrate all dependencies to an isolated VNet in order to perform UAT, or where UAT is not enforced.

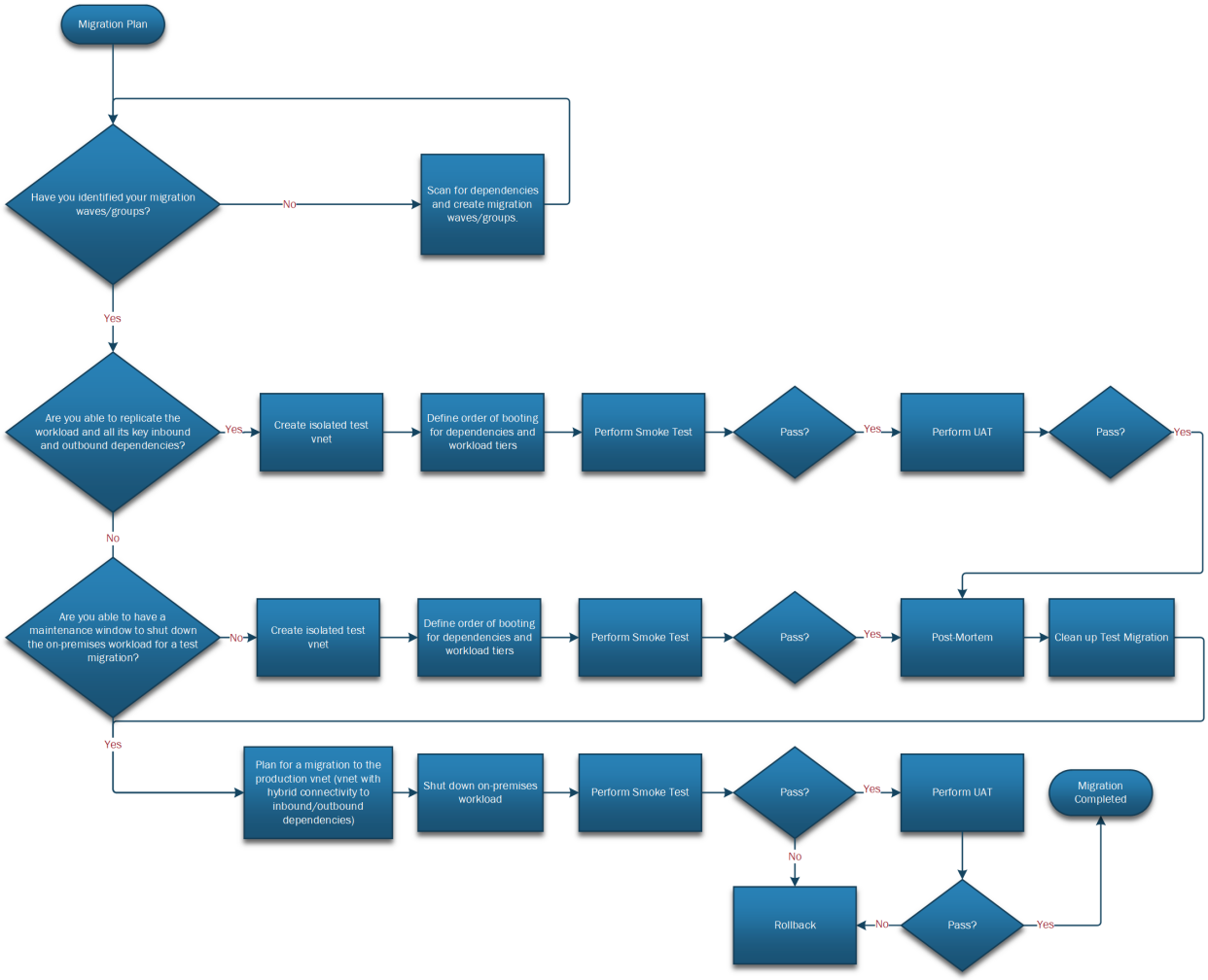


Figure 2: Testing and migration workflow.

Step 3: Final step

As a final step, Tailwind is now ready to confidently perform the production migrations. The envision that during cutover all hands on deck will be required to ensure end-to-end support. Furthermore, after the soak period concludes, Tailwind is looking forward to close-out and call for a successful migration to Azure.

Cutover

With the migration activities and workflow defined, Tailwind irons out the final plans for cutover by:

- Identifying a more specific cutover window, which is planned for a Friday evening or weekend. Each cutover window will last at a minimum for 4 hours.
- Notifying the business and those affected by the migration of the maintenance window. The maintenance window should include a meeting invite that includes the migration plan and a conference bridge to discuss any open items during the migration.
- Contacting the network administrators, backup administrators, server administrators, identity administrators, application owners, Microsoft support resources, and partners to ensure that each party is available during cutover.
- Ensuring a backup of the server has been committed prior to cutover.
- Ensuring the rollback plan is defined and ready for execution if needed.
- Ensure the migration handover to the operations team by setting expectations that day-2 operations must commence for Azure server backup, patching, monitoring, and so on.

Post-go-live

Once the cutover successfully concludes, Tailwind prepares for decommission of the source servers. Tailwind

decides that server decommissioning will be executed after the soak period timeline is concluded.

After each migration wave, Tailwind also conducts a brief retrospective to discuss what went well and what can be improved for future migration waves. Tailwind understands these incremental learnings and improvements will ensure a smoother migration for all subsequent migration waves.

Conclusion

In this article, Tailwind sets up Azure Migrate: Server Migration tools and plans for their infrastructure migration activities and workflow.

Not every step taken here is required for a server migration. In this case, Tailwind planned for a migration workflow, test plans and pre-migration and post-migration activities which can be accomplished by proactive and reliable replication.

Rehost an on-premises application on Azure VMs by using Azure Migrate

5/4/2021 • 13 minutes to read • Edit Online

This article demonstrates how the fictional company Contoso rehosts a two-tier Windows .NET front-end application running on VMware virtual machines (VMs) by migrating application VMs to Azure VMs.

The SmartHotel360 application used in this example is provided as open-source software. If you want to use it for your own testing purposes, you can download it from GitHub.

Business drivers

The IT leadership team has worked closely with business partners to understand what they want to achieve with this migration. They want to:

- **Address business growth.** Contoso is growing, so there's pressure on the company's on-premises systems and infrastructure.
- **Limit risk.** The SmartHotel360 application is critical for the Contoso business. The company wants to move the application to Azure with zero risk.
- **Extend.** Contoso doesn't want to modify the application, but it does want to ensure that the application is stable.

Migration goals

The Contoso cloud team has pinned down goals for this migration. It used these goals to determine the best migration method:

- After migration, the application in Azure should have the same performance capabilities as it does today in VMware. The application will remain as critical in the cloud as it is on-premises.
- Although this application is important to Contoso, the company doesn't want to invest in it at this time. Contoso wants to move the application safely to the cloud in its current form.
- Contoso doesn't want to change the ops model for this application. Contoso does want to interact with it in the cloud in the same way that it does now.
- Contoso doesn't want to change any application functionality. Only the application location will change.

Solution design

After establishing goals and requirements, Contoso designs and reviews a deployment solution. Contoso identifies the migration process, including the Azure services that it will use for the migration.

Current application

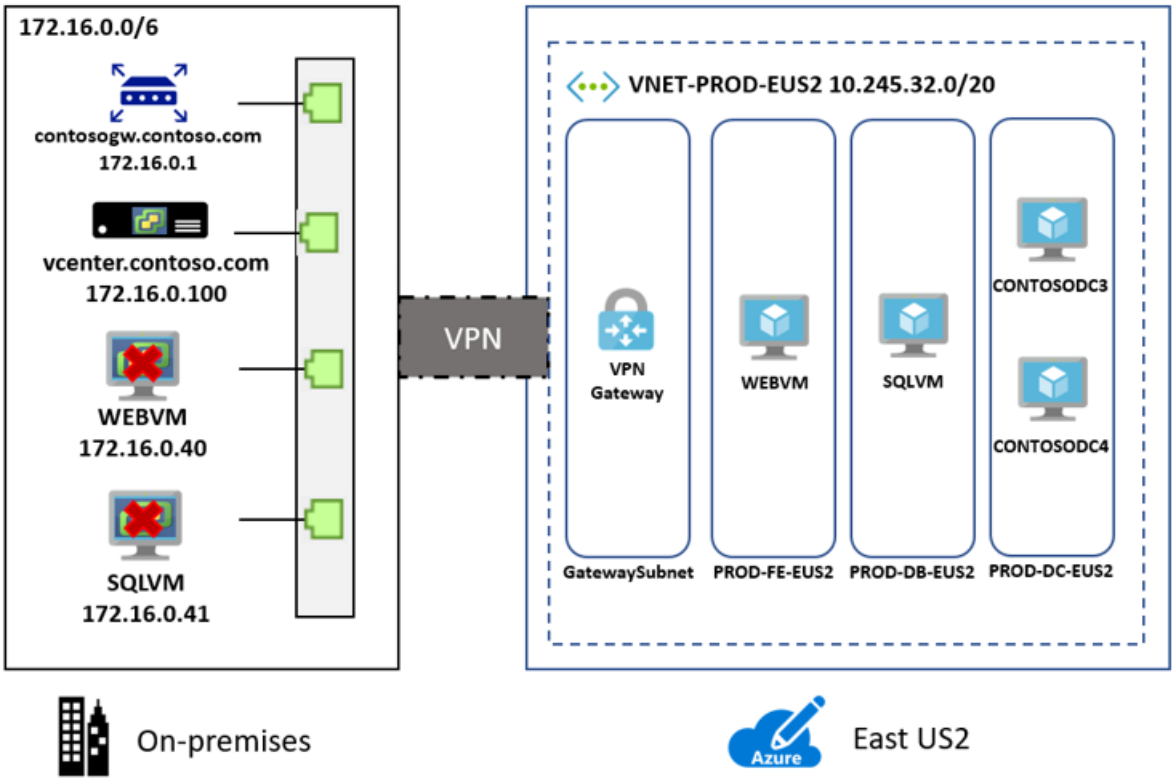
- The application is tiered across two VMs (`WEBVM` and `SQLVM`).
- The VMs are located on VMware ESXi host `contosohost1.contoso.com` (version 6.5).
- The VMware environment is managed by vCenter Server 6.5 (`vcenter.contoso.com`) running on a VM.
- Contoso has an on-premises datacenter (`contoso-datacenter`) with an on-premises domain controller (`contosodc1`).

Proposed architecture

- Because the application is a production workload, the application VMs in Azure will reside in the production

resource group `ContosoRG`.

- The application VMs will be migrated to the primary Azure region (East US 2) and placed in the production network (`VNET-PROD-EUS2`).
- The web front-end VM will reside in the front-end subnet (`PROD-FE-EUS2`) in the production network.
- The database VM will reside in the database subnet (`PROD-DB-EUS2`) in the production network.
- The on-premises VMs in the Contoso datacenter will be decommissioned after the migration is done.



Database considerations

As part of the solution design process, Contoso did a feature comparison between Azure SQL Database and SQL Server. The following considerations helped the company to decide to use SQL Server running on an Azure IaaS VM:

- Using an Azure VM running SQL Server seems to be an optimal solution if Contoso needs to customize the operating system and the database, or co-locate and run partner applications on the same VM.
- With Software Assurance, Contoso can later exchange existing licenses for discounted rates on Azure SQL Managed Instance by using the Azure Hybrid Benefit for SQL Server. This can save up to 30 percent on SQL Managed Instance.

Solution review

Contoso evaluates the proposed design by putting together a list of pros and cons.

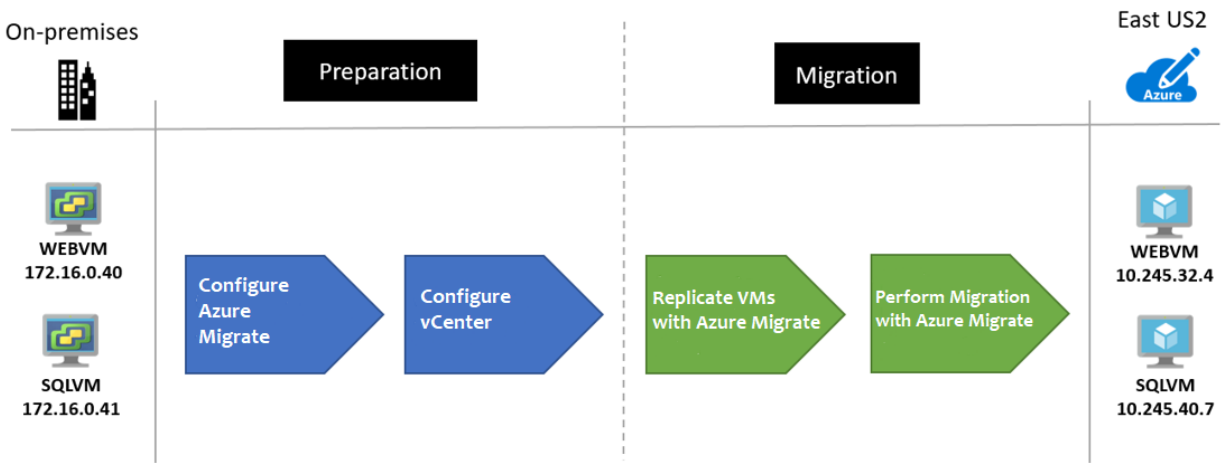
CONSIDERATION	DETAILS

CONSIDERATION	DETAILS
Pros	Both the application VMs will be moved to Azure without changes, making the migration simple. Because Contoso is using a lift-and-shift approach for both application VMs, it doesn't need any special configuration or migration tools for the application database. Contoso can take advantage of its investment in Software Assurance by using the Azure Hybrid Benefit. Contoso will retain full control of the application VMs in Azure.
Cons	`WEBVM` and `SQLVM` are running Windows Server 2008 R2. Azure supports the operating system for specific roles. Learn more. The web and data tiers of the application remain as single points of failure. `SQLVM` is running on SQL Server 2008 R2. SQL Server 2008 R2 is no longer in mainstream support, but it is supported for Azure VMs. Learn more. Contoso must continue supporting the application on Azure VMs rather than moving to a managed service such as Azure App Service or Azure SQL Database.

Migration process

Contoso will migrate the application front-end and database VMs to Azure VMs by using the agentless method in the Azure Migrate: Server Migration tool.

- As a first step, Contoso prepares and sets up Azure components for Azure Migrate: Server Migration, and prepares the on-premises VMware infrastructure.
- The Azure infrastructure is in place, so Contoso just needs to configure the replication of the VMs through the Azure Migrate: Server Migration tool.
- With everything prepared, Contoso can start replicating the VMs.
- After replication is enabled and working, Contoso will migrate the VM by testing the migration and failing it over to Azure, if successful.



Azure services

SERVICE	DESCRIPTION	COST
Azure Migrate: Server Migration	The service orchestrates and manages migration of on-premises applications and workloads as well as Amazon Web Services (AWS) and Google Cloud Platform (GCP) VM instances.	During replication to Azure, Azure Storage charges are incurred. Azure VMs are created, and incur charges, when the migration occurs and the VMs are running in Azure. Learn more about charges and pricing.

Prerequisites

Contoso and other users must meet the following prerequisites for this scenario.

REQUIREMENTS	DETAILS
Azure subscription	Contoso created subscriptions in an earlier article in this series. If you don't have an Azure subscription, create a free account. If you create a free account, you're the administrator of your subscription and can perform all actions. If you use an existing subscription and you're not the administrator, work with the admin to assign you Owner or Contributor permissions. If you need more granular permissions, see Manage Site Recovery access with Azure role-based access control.
Azure infrastructure	Learn how Contoso set up an Azure infrastructure. Learn more about specific prerequisites for Azure Migrate: Server Migration.
On-premises servers	On-premises vCenter servers should be running version 5.5, 6.0, 6.5, or 6.7. ESXi hosts should run version 5.5, 6.0, 6.5, or 6.7. One or more VMware VMs should be running on the ESXi host.

Scenario steps

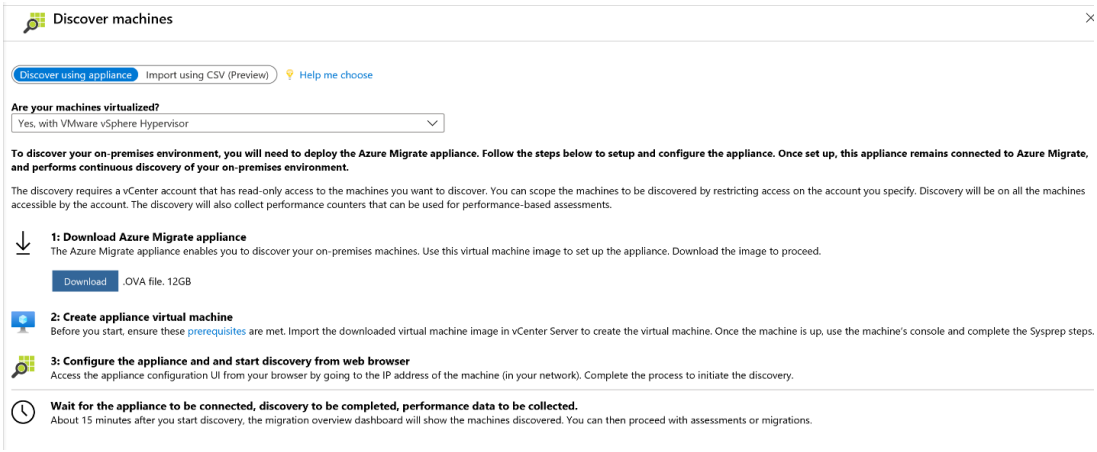
Here's how Contoso admins will run the migration:

- **Step 1: Prepare Azure for Azure Migrate: Server Migration.** They add the server migration tool to their Azure Migrate project.
- **Step 2: Replicate on-premises VMs.** They set up replication and start replicating VMs to Azure Storage.
- **Step 3: Migrate the VMs with Azure Migrate: Server Migration.** They run a test migration to make sure everything's working, and then run a full migration to move the VMs to Azure.

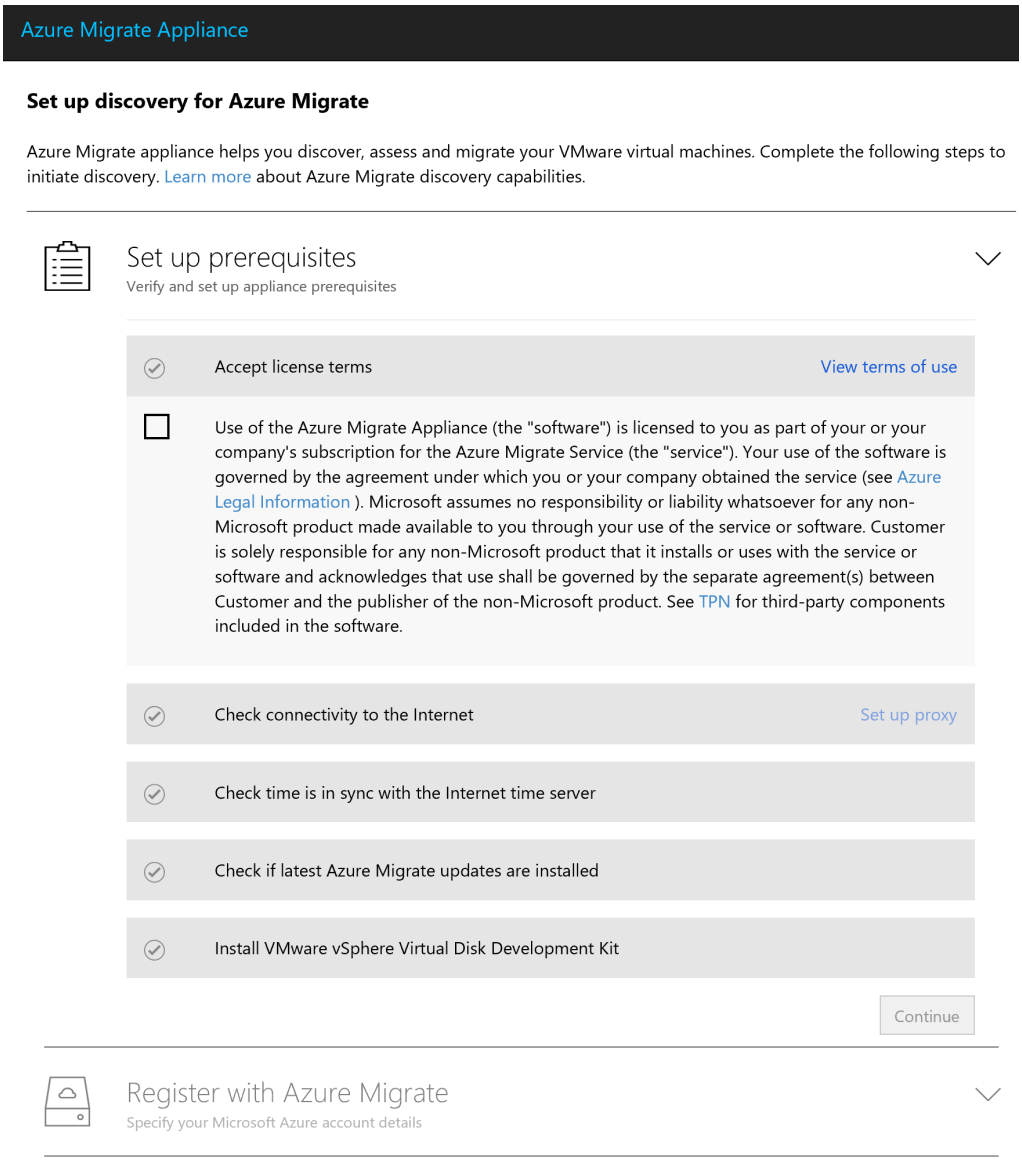
Step 1: Prepare Azure for Azure Migrate: Server Migration

To migrate the VMs to Azure, Contoso needs a virtual network in which Azure VMs will be located when they're created during migration. It also needs the Azure Migrate: Server Migration tool (OVA file) provisioned and configured.

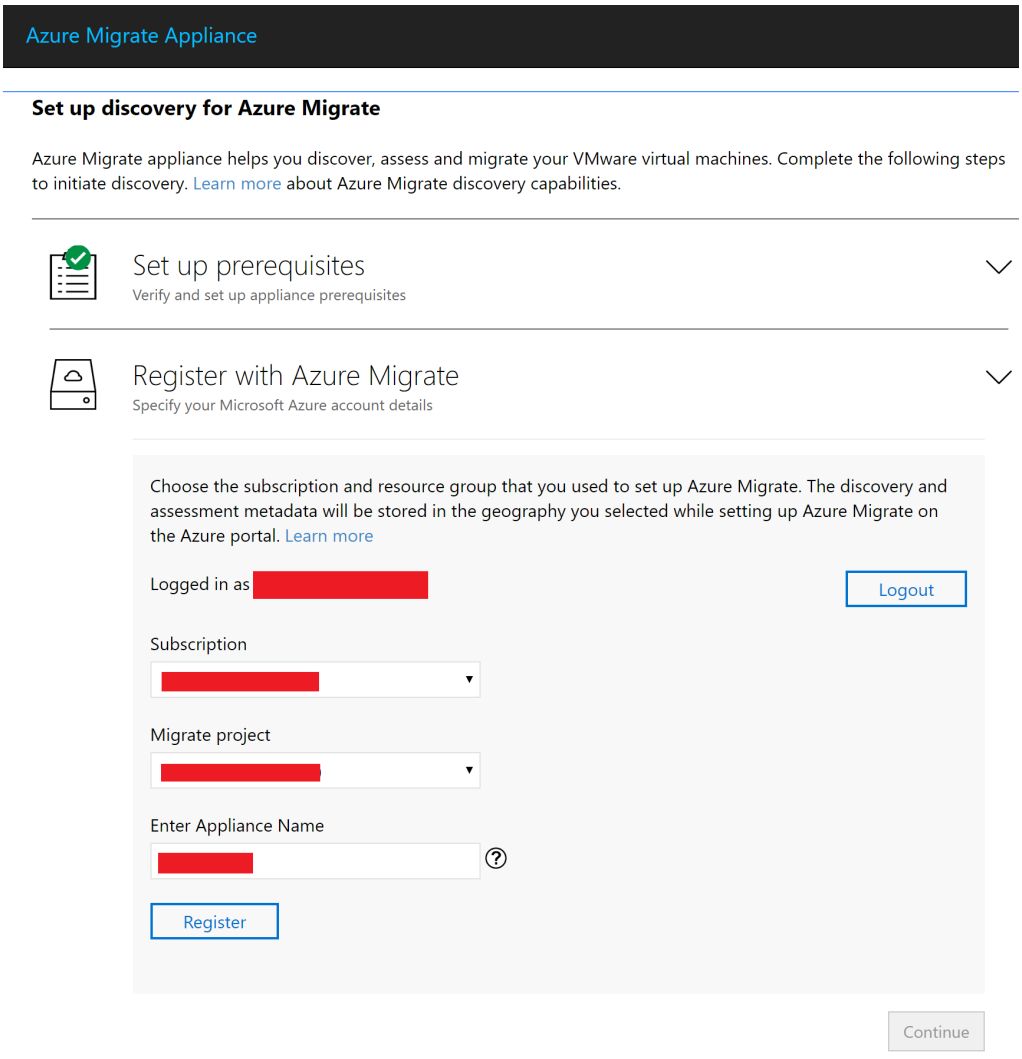
1. Set up a network. Contoso already set up one that can be used for Azure Migrate: Server Migration when it deployed the Azure infrastructure.
    - The SmartHotel360 application is a production application, and the VMs will be migrated to the Azure production network (`VNET-PROD-EUS2`) in the primary region (`East US 2`).
    - Both VMs will be placed in the `ContosoRG` resource group, which is used for production resources.
    - The application front-end VM (`WEBVM`) will migrate to the front-end subnet (`PROD-FE-EUS2`) in the production network.
    - The application database VM (`SQLVM`) will migrate to the database subnet (`PROD-DB-EUS2`) in the production network.
2. Provision the Azure Migrate: Server Migration tool.
    a. From Azure Migrate, download the OVA image and import it into VMware.



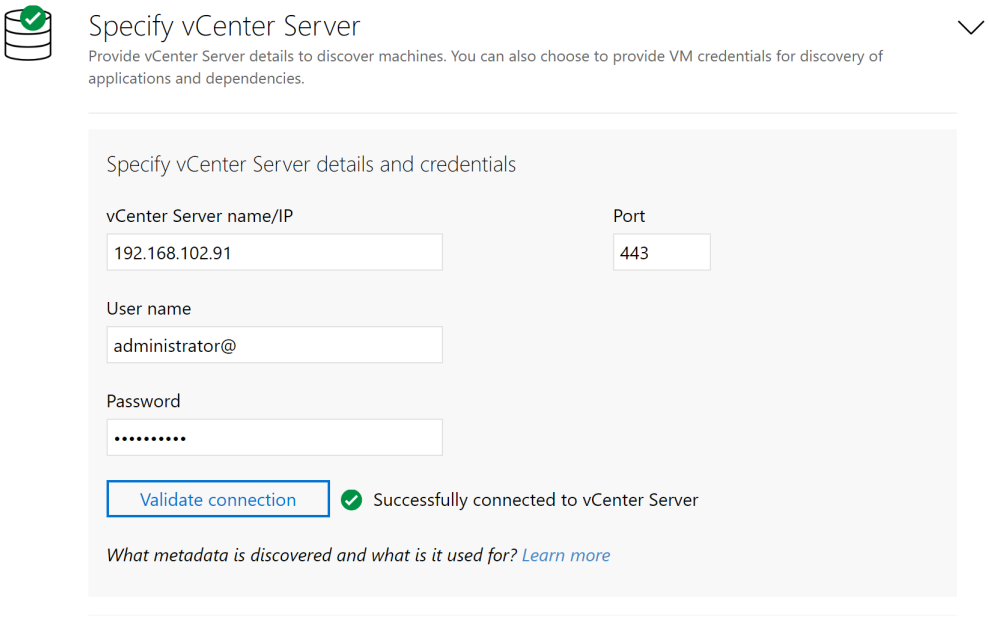
    b. Start the imported image and configure the tool, including the following steps:
        - Set up the prerequisites.



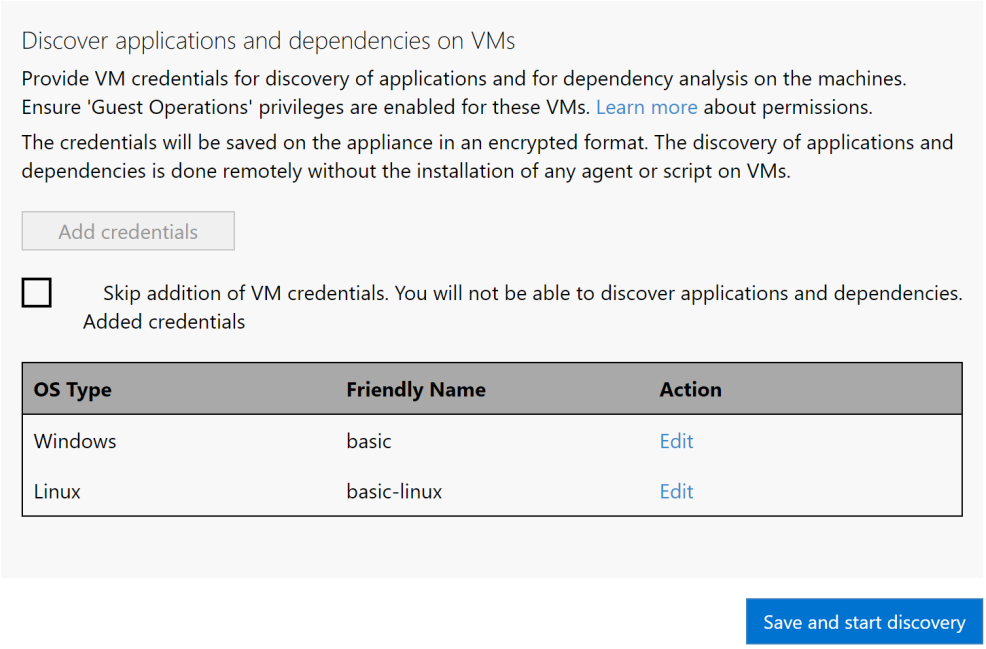
- Point the tool to the Azure subscription.



- Set the VMware vCenter credentials.



- Add any Windows-based credentials for discovery.



When you complete the configuration, the tool will take some time to enumerate all the VMs. You'll see them populate the Azure Migrate tool in Azure when this process finishes.

Need more help?

Learn about how to set up the Azure Migrate: Server Migration tool.

Prepare on-premises VMs

After migration, Contoso wants to connect to the Azure VMs and allow Azure to manage the VMs. The Contoso admins must do the following steps before migration:

1. For access over the internet:
   - Enable RDP or SSH on the on-premises VM before migration.
   - Ensure that TCP and UDP rules are added for the **Public** profile.
   - Check that RDP or SSH is allowed in the operating system firewall.
2. For access over Site-to-Site VPN:
   - Enable RDP or SSH on the on-premises VM before migration.
   - Check that RDP or SSH is allowed in the operating system firewall.
   - For Windows, set the operating system's SAN policy on the on-premises VM to **OnlineAll**.
3. Install the Azure Windows agent.

Other considerations:

- For Windows, there should be no Windows updates pending on the VM when you're triggering a migration. If there are, the admins won't be able to sign in to the VM until the updates finish.
- After migration, the admins can check **Boot diagnostics** to view a screenshot of the VM. If this doesn't work, they should verify that the VM is running and review troubleshooting tips.

Need more help?

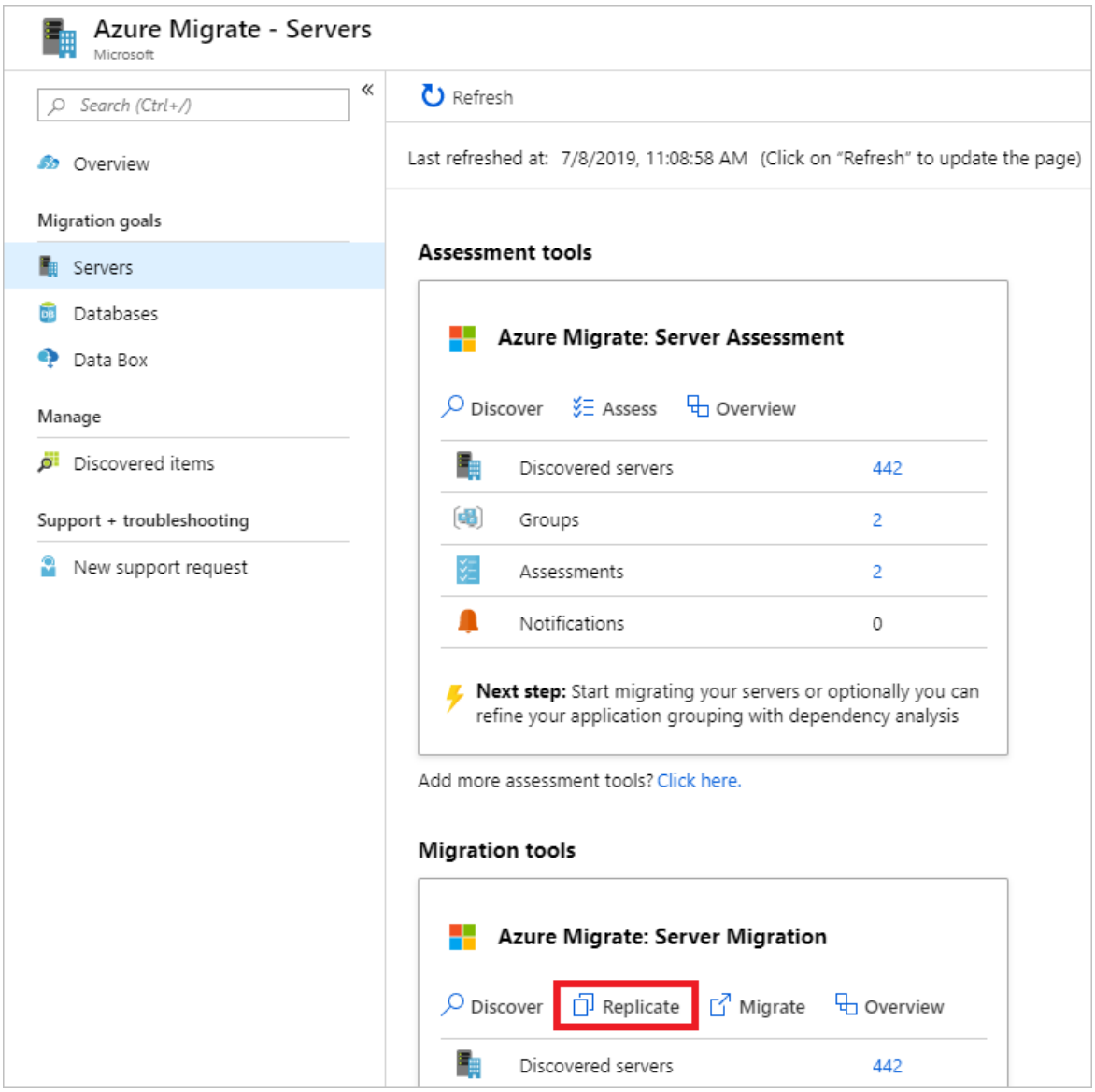
Learn about how to prepare VMs for migration.

Step 2: Replicate the on-premises VMs

Before the Contoso admins can run a migration to Azure, they need to set up and enable replication.

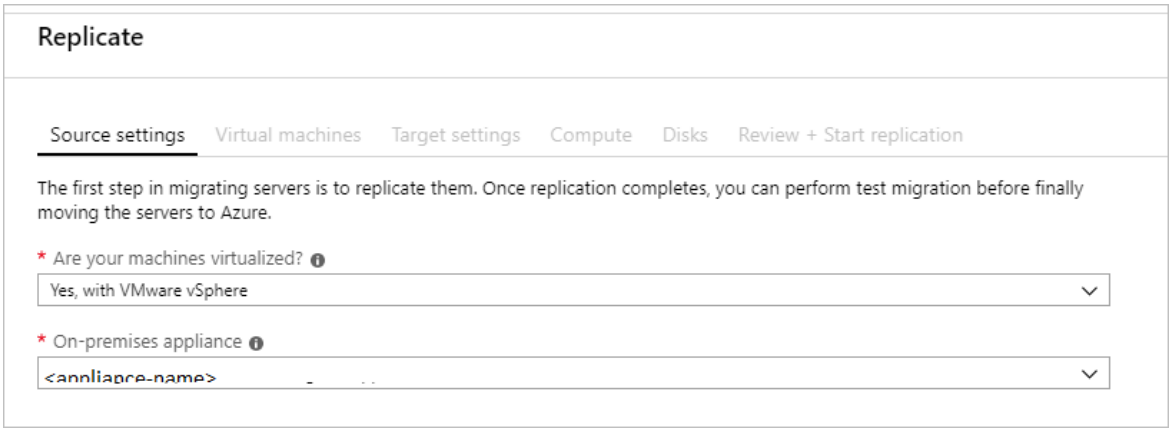
With discovery completed, they can begin replication of VMware VMs to Azure.

1. In the Azure Migrate project, go to **Servers** > **Azure Migrate: Server Migration**. Then select **Replicate**.



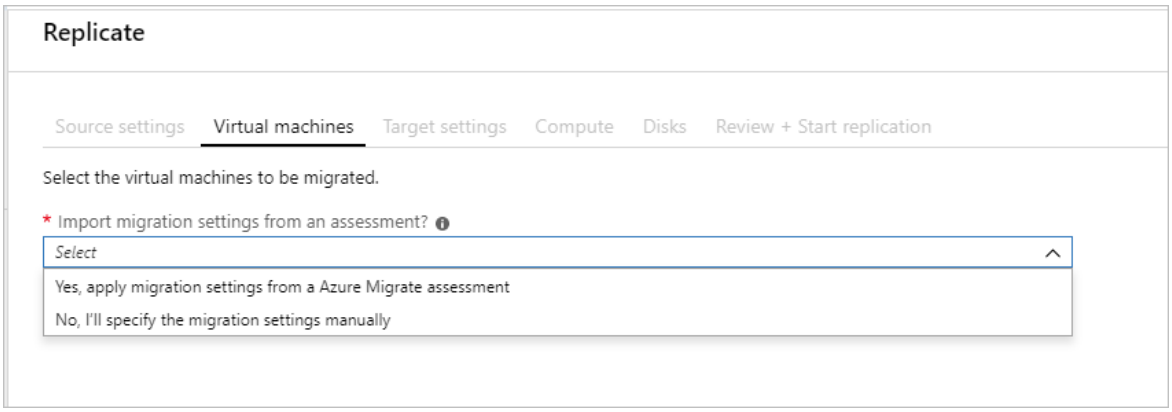
2. In **Replicate** > **Source settings** > **Are your machines virtualized?**, select **Yes, with VMware vSphere**.

3. In **On-premises appliance**, select the name of the Azure Migrate appliance that you set up, and then select **OK**.



4. In **Virtual machines**, select the machines that you want to replicate.

- If you've run an assessment for the VMs, you can apply VM sizing and disk type (premium or standard) recommendations from the assessment results. To do this, in **Import migration settings from an Azure Migrate assessment?**, select the **Yes** option.
- If you didn't run an assessment, or you don't want to use the assessment settings, select the **No** option.
- If you selected to use the assessment, select the VM group and assessment name.



5. In **Virtual machines**, search for VMs as needed and check each VM that you want to migrate. Then select **Next: Target settings**.
6. In **Target settings**, select the subscription and target region to which you'll migrate. Then specify the resource group in which the Azure VMs will reside after migration. In **Virtual Network**, select the Azure virtual network or subnet to which the Azure VMs will be joined after migration.
7. In **Azure Hybrid Benefit**:
   - Select **No** if you don't want to apply Azure Hybrid Benefit. Then select **Next**.
   - Select **Yes** if you have Windows Server machines that are covered with active Software Assurance or Windows Server subscriptions and you want to apply the benefit to the machines that you're migrating. Then select **Next**.
8. In **Compute**, review the VM name, size, OS disk type, and availability set. VMs must conform with Azure requirements.
   - **VM size:** If you're using assessment recommendations, the VM size drop-down list will contain the recommended size. Otherwise, Azure Migrate picks a size based on the closest match in the Azure subscription. Alternatively, pick a manual size in **Azure VM size**.
   - **OS disk:** Specify the OS (boot) disk for the VM. The OS disk has the operating system bootloader and installer.
   - **Availability set:** If the VM should be in an Azure availability set after migration, specify the set. The set must be in the target resource group that you specify for the migration.
9. In **Disks**, specify whether the VM disks should be replicated to Azure, and select the disk type (standard SSD/HDD or premium-managed disks) in Azure. Then select **Next**.

   You can exclude disks from replication. If you exclude disks, they won't be present on the Azure VM after migration.
10. In **Review and start replication**, review the settings, and then select **Replicate** to start the initial replication for the servers.

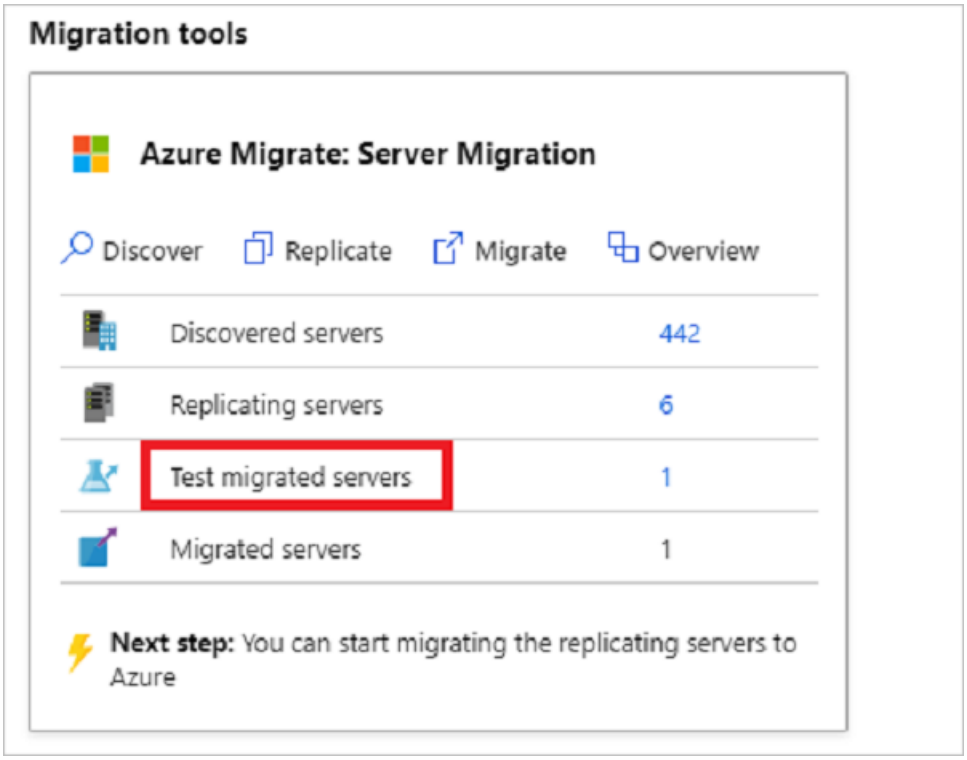
> **NOTE**
> You can update replication settings at any time before replication starts, in **Manage** > **Replicating machines**. Settings can't be changed after replication starts.

Step 3: Migrate the VMs with Azure Migrate: Server Migration

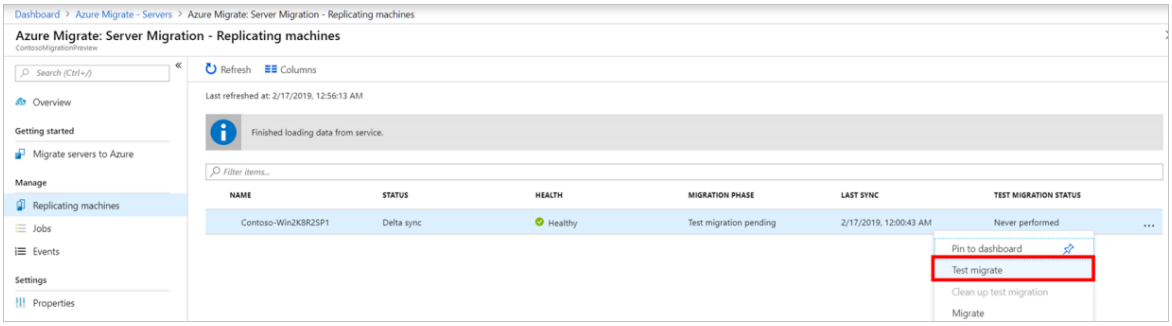
The Contoso admins run a quick test migration and then a full migration to migrate the VMs.

Run a test migration

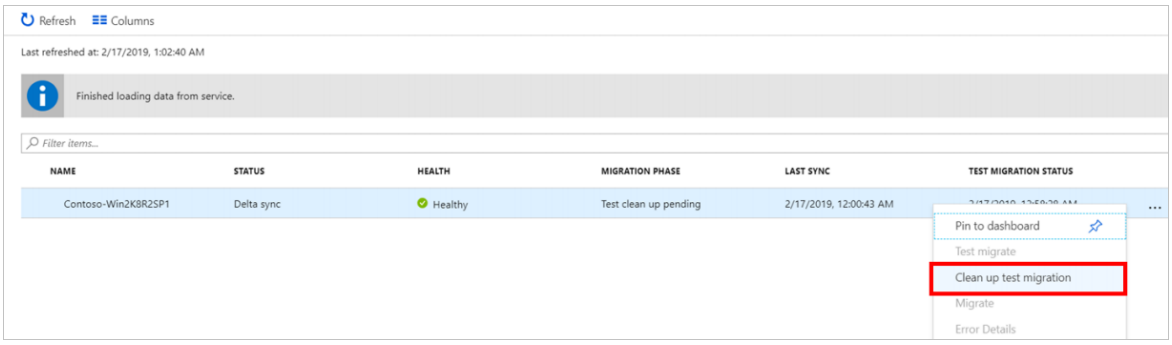
1. In **Migration goals** > **Servers** > **Azure Migrate: Server Migration**, select **Test migrated servers**.



2. Select and hold (or right-click) the VM to test, and then select **Test migrate**.



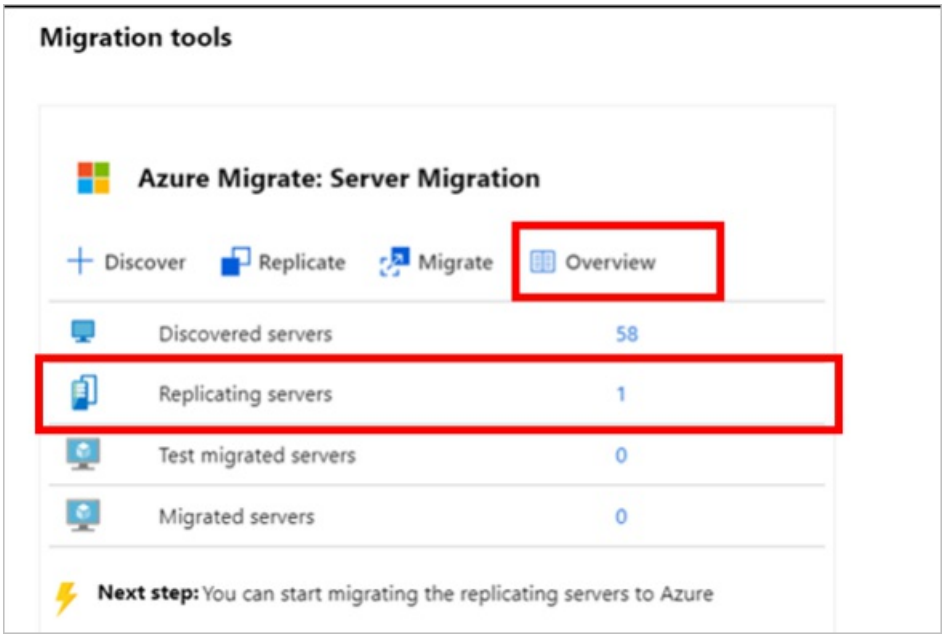
3. In **Test Migration**, select the Azure virtual network in which the Azure VM will be located after the migration. We recommend that you use a nonproduction virtual network.
4. The **Test migration** job starts. Monitor the job in the portal notifications.
5. After the migration finishes, view the migrated Azure VM in **Virtual Machines** in the Azure portal. The machine name has a **-Test** suffix.
6. After the test is done, select and hold (or right-click) the Azure VM in **Replicating machines**, and then select **Clean up test migration**.



Migrate the VMs

Now the Contoso admins run a full migration.

1. In the Azure Migrate project, select **Servers** > **Azure Migrate: Server Migration** > **Replicating servers**.



2. In **Replicating machines**, select and hold (or right-click) the VM, and then select **Migrate**.
3. In **Migrate** > **Shut down virtual machines and perform a planned migration with no data loss**, select **Yes** > **OK**.

   By default, Azure Migrate shuts down the on-premises VM and runs an on-demand replication to synchronize any VM changes that occurred since the last replication. This ensures no data loss. If you don't want to shut down the VM, select **No**.
4. A migration job starts for the VM. Track the job in Azure notifications.
5. After the job finishes, you can view and manage the VM from the **Virtual Machines** page.

Need more help?

- Learn about how to run a test migration.
- Learn about how to migrate VMs to Azure.

Clean up after migration

With migration complete, the SmartHotel360 application tiers are now running on Azure VMs.

Now, Contoso needs to do these cleanup steps:

- After the migration is complete, stop replication.
- Remove the `WEBVM` machine from the vCenter inventory.
- Remove the `SQLVM` machine from the vCenter inventory.
- Remove `WEBVM` and `SQLVM` from local backup jobs.
- Update internal documentation to show the new location and IP addresses for the VMs.
- Review any resources that interact with the VMs, and update any relevant settings or documentation to reflect the new configuration.

Review the deployment

With the application now running, Contoso needs to fully operationalize and secure it in Azure.

Security

The Contoso security team reviews the Azure VMs to determine any security issues. To control access, the team reviews the network security groups (NSGs) for the VMs. NSGs are used to ensure that only traffic allowed to the application can reach it. The team also considers securing the data on the disk by using Azure Disk Encryption and Key Vault.

For more information, see Security best practices for IaaS workloads in Azure.

Business continuity and disaster recovery

For business continuity and disaster recovery, Contoso takes the following actions:

- Keep data safe: Contoso backs up the data on the VMs by using Azure Backup.
- Keep applications up and running: Contoso replicates the application VMs in Azure to a secondary region by using Azure Site Recovery.

Licensing and cost optimization

Contoso has existing licensing for its VMs and will take advantage of the Azure Hybrid Benefit. Contoso will convert the existing Azure VMs to take advantage of this pricing.

Contoso will enable Azure Cost Management + Billing to help monitor and manage Azure resources.

Conclusion

In this article, Contoso rehosted the SmartHotel360 application in Azure. The admins migrated the application VMs to Azure VMs by using the Azure Migrate: Server Migration tool. You can also review the Azure DevOps projects which have been published in the DevOps generator. Once in the generator, download the Server Migration Project under the Cloud Adoption Framework navigation.

Migrate SQL Server databases to Azure

5/26/2021 • 10 minutes to read • Edit Online

This article demonstrates how a fictional company Contoso assessed, planned and migrated their various on-premises SQL Server databases to Azure.

As Contoso considers migrating to Azure, the company needs a technical and financial assessment to determine whether its on-premises workloads are good candidates for cloud migration. In particular, the Contoso team wants to assess machine and database compatibility for migration. Additionally, it wants to estimate capacity and costs for running Contoso's resources in Azure.

Business drivers

Contoso is having various issues with maintaining all the wide array of versions of SQL Server workloads that exist on their network. After the latest investor's meeting, the CFO and CTO have made the decision to move all these workloads to Azure. This will allow them to shift from a structured capital expense model to a fluid operating expense model.

The IT leadership team has worked closely with business partners to understand the business and technical requirements:

- **Increase security:** Contoso needs to be able to monitor and protect all data resources in a more timely and efficient manner. They would also like to get a more centralized reporting system setup on database access patterns.
- **Optimize compute resources:** Contoso has deployed a large on-premises server infrastructure. They have several SQL Server instances that consume but do not really use the underlying CPU, memory and disk allocated in efficient ways.
- **Increase efficiency:** Contoso needs to remove unnecessary procedures, and streamline processes for developers and users. The business needs IT to be fast and not waste time or money, thus delivering faster on customer requirements. Database administration should be reduced and/or minimized after the migration.
- **Increase agility:** Contoso IT needs to be more responsive to the needs of the business. It must be able to react faster than the changes in the marketplace, to enable the success in a global economy. It mustn't get in the way, or become a business blocker.
- **Scale:** As the business grows successfully, Contoso IT must provide systems that are able to grow at the same pace. There are several legacy hardware environments that cannot be upgraded any further and are past or near end of support.
- **Costs:** Business and applications owners want to know they won't be stuck with high cloud costs as compared to running the applications on-premises.

Migration goals

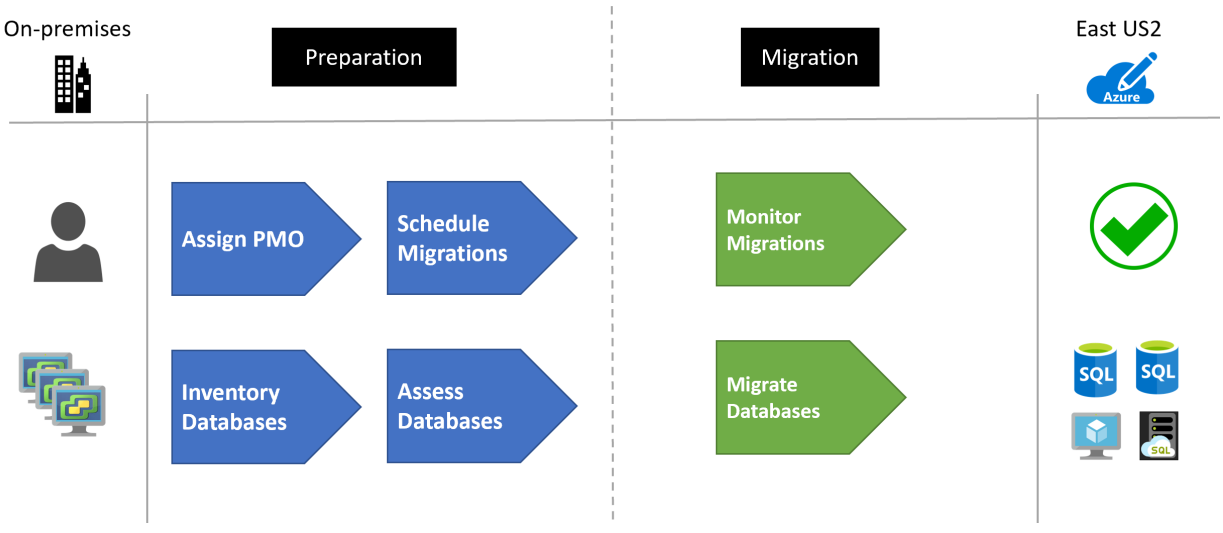
The Contoso cloud team has pinned down goals for the various migrations. These goals were used to determine the best migration methods.

REQUIREMENTS	DETAILS
Performance	After migration, applications in Azure should have the same performance capabilities that applications have today in Contoso's on-premises environment. Moving to the cloud doesn't mean that application performance is less critical.
Compatibility	Contoso needs to understand the compatibility of its applications and databases with Azure. Contoso also needs to understand its Azure hosting options.
Data sources	All databases will be moved to Azure with no exceptions. Based on the database and application analysis of the SQL features being used, they will move to PaaS, IaaS or managed instances. All databases must move.
Application	Applications must be moved to the cloud wherever possible. If they cannot move, then they will be allowed to connect to the migrated database over the Azure network through private connections only.
Costs	Contoso wants to understand not only its migration options, but also the costs associated with the infrastructure after it moves to the cloud.
Management	Resource management groups must be created for the various departments along with resource groups to managed all SQL databases that are migrated. All resources must be tagged with department information for chargeback requirements.
Limitations	Initially, not all branch offices that run applications will have a direct ExpressRoute link to Azure, so these offices will need to connect through virtual network gateways.

Solution design

Contoso has already performed a migration assessment of their digital estate using Azure Migrate.

The assessment results in multiple workloads spread across multiple departments. The overall size of the migration project will require a full project management office (PMO), to manage the specifics of communication, resources and schedule planning.



Solution review

Contoso evaluates their proposed design by putting together a pros and cons list.

CONSIDERATION	DETAILS
Pros	Azure will provide a single pane of glass into the database workloads Costs will be monitored via Azure Cost Management + Billing. Business chargeback billing will be easy to perform with the Azure Billing APIs. Server and software maintenance will be reduced to only the IaaS-based environments.
Cons	Due to the requirement of IaaS-based virtual machines, there will still need to be management of the software on those machines.

Budget and management

Before the migration can occur, the necessary Azure structure is required to be in place to support the administration and billing aspects of the solution.

For the management requirements, several management groups were created to support the organizational structure.

For the billing requirements, each of the Azure resources are then tagged with the appropriate billing tags.

Migration process

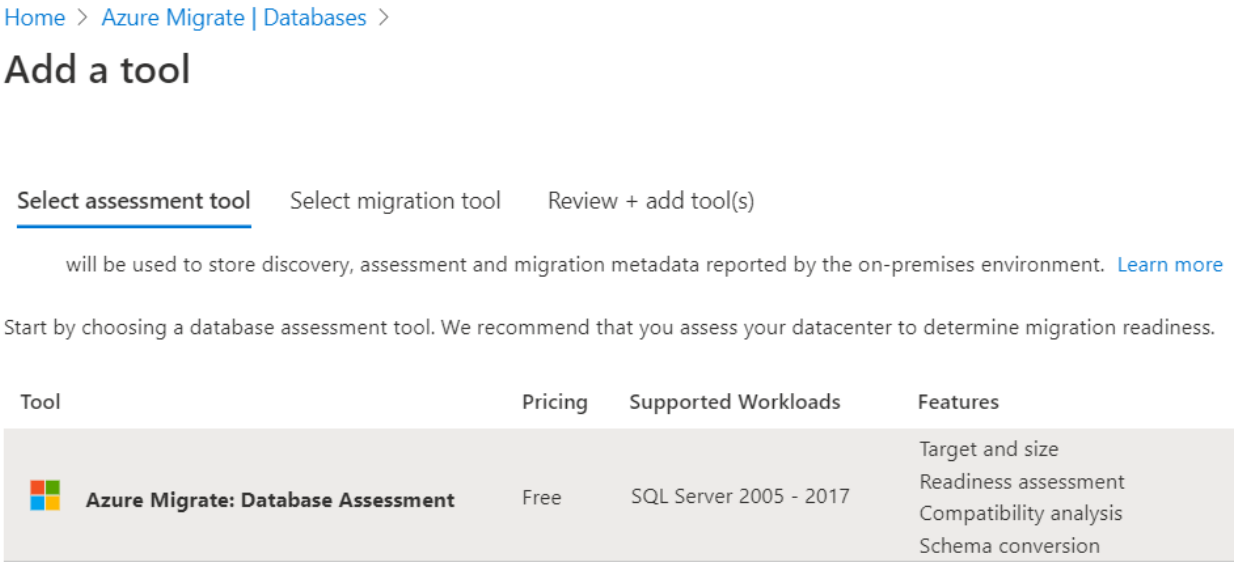
Data migrations follow a standard repeatable pattern. This involves the following steps based on Microsoft best practices:

- Pre-migration:
  - **Discovery:** Inventory database assets and application stack.
  - **Assess:** Assess workloads and fix recommendations.
  - **Convert:** Convert source schema to work in the target.
- Migration:
  - **Migrate:** Migrate the source schema, source data and objects to target.
  - **Sync data:** Sync data (for minimal downtime).
  - **Cutover:** Cut over the source to target.
- Post-migration:
  - **Remediate applications:** Iteratively make and necessary changes to your applications.
  - **Perform tests:** Iteratively run functional and performance tests.
  - **Optimize:** Based on tests, address performance issues and then retest to confirm performance improvements.
  - **Retire assets:** Old VMs and hosting environments are backed up and retired.

Step 1: Discovery

Contoso used Azure Migrate to surface the dependencies across the Contoso environment. Azure Migrate automatically discovered application components on Windows and Linux systems and mapped the communication between services. Azure Migrate also surfaced the connections between Contoso servers, processes, inbound and outbound connection latency, and ports across their TCP-connected architecture.

Contoso also added Data Migration Assistant to their Azure Migrate project. By selecting this tool they're able to assess the databases for migration to Azure.



Step 2: Application assessment

The assessment determined that Contoso uses mainly .NET-based applications. However, some projects have used other technologies such as PHP and Node.js. Vendor-purchased systems also introduced applications not based on .NET. Contoso identified the following applications:

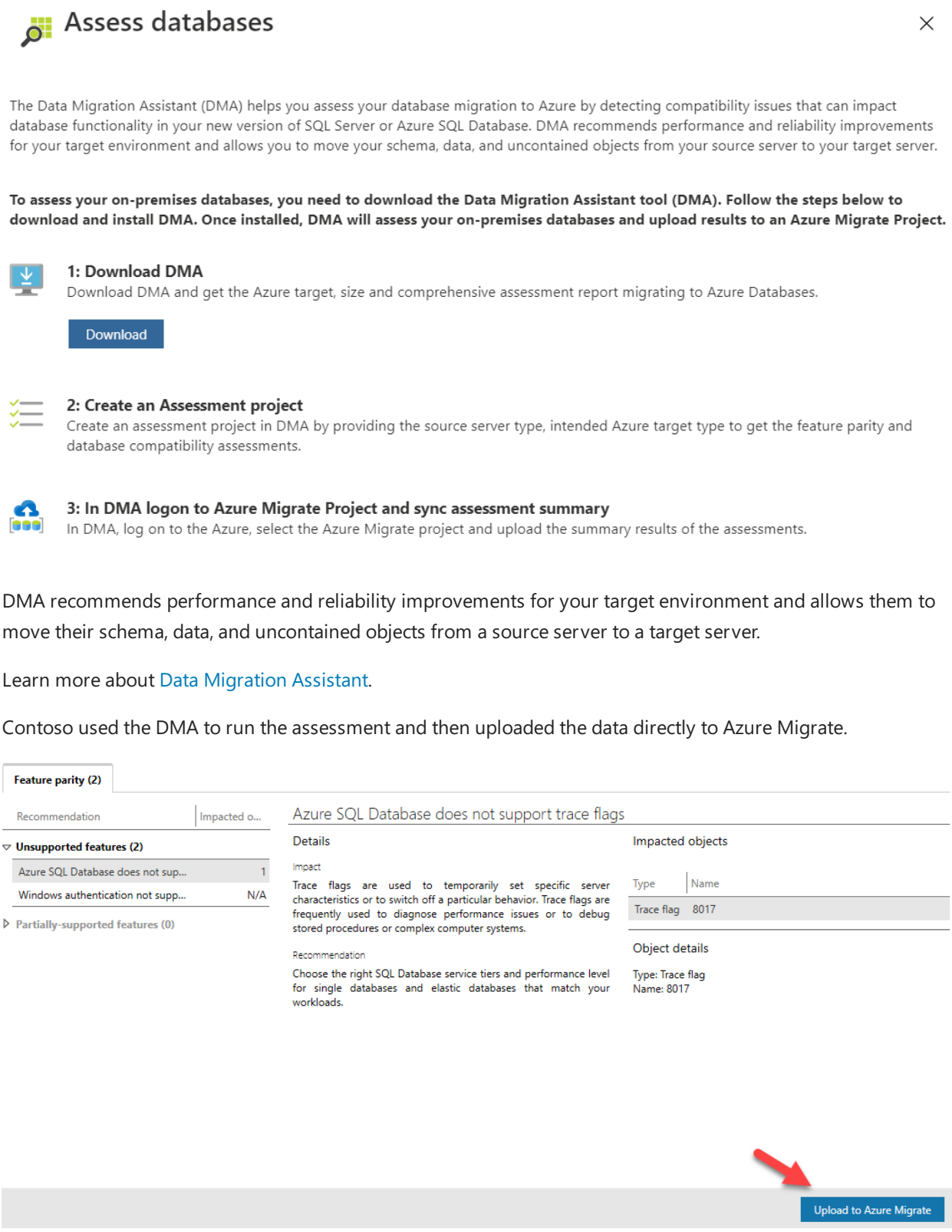
- ~800 Windows .NET applications
- ~50 PHP applications
- 25 Node.js applications
- 10 Java applications

Step 3: Database assessment

As each database workload was discovered, Data Migration Assistant (DMA) tool was run to determine which features were being used. DMA helps Contoso assess their database migrations to Azure by detecting compatibility issues that can impact database functionality in a new version of SQL Server or Azure SQL Database.

Contoso followed these steps to assess their databases and then upload results data to Azure Migrate:

1. Download DMA.
2. Create an assessment project.
3. In DMA, logon to the Azure Migrate project and sync the assessment summary.



DMA recommends performance and reliability improvements for your target environment and allows them to move their schema, data, and uncontained objects from a source server to a target server.

Learn more about Data Migration Assistant.

Contoso used the DMA to run the assessment and then uploaded the data directly to Azure Migrate.

With the database information now loaded into Azure Migrate, Contoso has identified over 1,000 database instances that must be migrated. Of these instances, roughly 40 percent can be moved to SQL Database for Azure. The remaining 60 percent must be moved either to either SQL Server running on Azure Virtual Machines or to Azure SQL Managed Instance. Of those 60 percent, about 10 percent require a virtual machine-based approach, the remaining instances will be moved to Azure SQL Managed Instance.

When DMA was not able to be executed on a data source, the following guidelines were followed on the database migrations.

> **NOTE**
> Contoso discovered various open-source databases during the assessment phase. Separately, they followed the guidance in migrate open-source databases to Azure for their migration planning.

Step 4: Migration planning

With the information at hand, Contoso uses the following guidelines to determine which migration method to use for each database.

TARGET	DATABASE USAGE	DETAILS	ONLINE MIGRATION	OFFLINE MIGRATION	MAX SIZE	MIGRATION GUIDE
Azure SQL Database (PaaS)	SQL Server (data only)	These databases simply use basic tables, columns, stored procedures and functions	Data Migration Assistant, transactional replication	BACPAC, bcp	1 TiB	Link
Azure SQL Managed Instance	SQL Server (advanced features)	These databases use triggers and other advanced concepts such as custom .NET types, service brokers, and so on.	Data Migration Assistant, transactional replication	BACPAC, bcp, native backup/restore	2 TiB - 8 TiB	Link

TARGET	DATABASE USAGE	DETAILS	ONLINE MIGRATION	OFFLINE MIGRATION	MAX SIZE	MIGRATION GUIDE
SQL Server on Azure Virtual Machines (IaaS)	SQL Server (third-party integrations)	The SQL Server must have non-supported SQL Managed Instance features (cross-instance service brokers, cryptographic providers, buffer pool, compatibility levels below 100, database mirroring, FILESTREAM, PolyBase, anything that requires access to file shares, external scripts, extended stored procedures, and others) or third-party software installed to support the activities of the database.	Transactional replication	BACPAC, bcp, snapshot replication, native backup/restore, convert physical machine to VM	4 GiB - 64 TiB	Link

Due to the large number of databases, Contoso created a project management office (PMO) to keep track of every database migration instance. Accountability and responsibilities were assigned to each business and application team.

Contoso also performed a workload readiness review. This review examined the infrastructure, database and network components.

Step 5: Test migrations

The first part of the migration preparation involved a test migration of each of the databases to the pre-setup environments. In order to save time, they scripted all of the operations for the migrations and recorded the timings for each. In order to speed up the migration, they identified what migration operations could be run concurrently.

Any rollback procedures were identified for each of the database workloads in case of some unexpected failures.

For the IaaS-based workloads, they set up all the required third-party software beforehand.

After the test migration, Contoso was able to use the various Azure cost estimation tools to get a more accurate picture of the future operational costs of their migration.

Step 6: Migration

For the production migration, Contoso identified the time frames for all database migrations and what could be sufficiently executed in a weekend window (midnight Friday through midnight Sunday) with minimal downtime to the business.

Based on their documented test procedures, they execute each migration via scripting as much as possible, limiting any manual tasks to minimize errors.

If any migrations fail during the window, they're rolled back and re-scheduled in the next migration window.

Clean up after migration

Contoso identified the archival window for all database workloads. As the window expires, the resources will be retired from the on-premises infrastructure.

This includes:

- Removing the production data from on-premises servers.
- Retiring the hosting server when the last workload window expires.

Review the deployment

With the migrated resources in Azure, Contoso needs to fully operationalize and secure their new infrastructure.

Security

- Contoso needs to ensure that their new Azure database workloads are secure. Learn more.
- In particular, Contoso should review the firewall and virtual network configurations.
- Setup Private Link so that all database traffic is kept inside Azure and the on-premises network.
- Enable Microsoft Defender for Identity for Azure SQL Database.

Backups

- Ensure that the Azure databases are backed up using geo-restore. This allows backups to be used in a paired region in case of a regional outage.
- **Important:** Ensure that the Azure resource has a resource lock to prevent it from being deleted. Deleted servers cannot be restored.

Licensing and cost optimization

- Many Azure database workloads can be scaled up or down, therefore performance monitoring of the server and databases is important to ensure you're meeting your needs but also keeping costs at a minimum.
- Both CPU and storage have costs associated. There are several pricing tiers to select from. Be sure the appropriate pricing plan is selected for the data workloads.
- Elastic pools are to be implemented for databases that have compatible resource utilization patterns.
- Each read replica is billed based on the compute and storage selected
- Use reserved capacity to save on costs.

Conclusion

In this article, Contoso assessed, planned, and migrated their Microsoft SQL Server workloads to Azure.

An Azure DevOps project has been developed for you to study in your SQL migration journey, and it aligns with the Cloud Adoption Framework. This project will guide you through the key decisions required. Select this link to navigate to the Azure DevOps project.

Rehost an on-premises application by migrating to Azure VMs and Azure SQL Managed Instance

3/6/2021 • 26 minutes to read • Edit Online

This article shows how the fictional company Contoso migrates a two-tier Windows .NET front-end application running on VMware virtual machines (VMs) to an Azure VM by using Azure Migrate. It also shows how Contoso migrates the application database to Azure SQL Managed Instance.

The SmartHotel360 application used in this example is provided as open-source software. If you want to use it for your own testing purposes, download it from GitHub.

Business drivers

Contoso's IT leadership team has worked closely with the company's business partners to understand what the business wants to achieve with this migration. They want to:

- **Address business growth.** Contoso is growing. As a result, pressure has increased on the company's on-premises systems and infrastructure.
- **Increase efficiency.** Contoso needs to remove unnecessary procedures and streamline processes for its developers and users. The business needs IT to be fast and not waste time or money for the company to deliver faster on customer requirements.
- **Increase agility.** Contoso IT needs to be more responsive to the needs of the business. It must react faster than the changes that occur in the marketplace for the company to be successful in a global economy. IT at Contoso must not get in the way or become a business blocker.
- **Scale.** As the company's business grows successfully, Contoso IT must provide systems that can grow at the same pace.

Migration goals

The Contoso cloud team has identified goals for this migration. The company uses migration goals to determine the best migration method.

- After migration, the application in Azure should have the same performance capabilities that the application has today in Contoso's on-premises VMware environment. Moving to the cloud doesn't mean that application performance is less critical.
- Contoso doesn't want to invest in the application. The application is critical and important to the business, but Contoso simply wants to move the application in its current form to the cloud.
- Database administration tasks should be minimized after the application is migrated.
- Contoso doesn't want to use Azure SQL Database for this application. It's looking for alternatives.

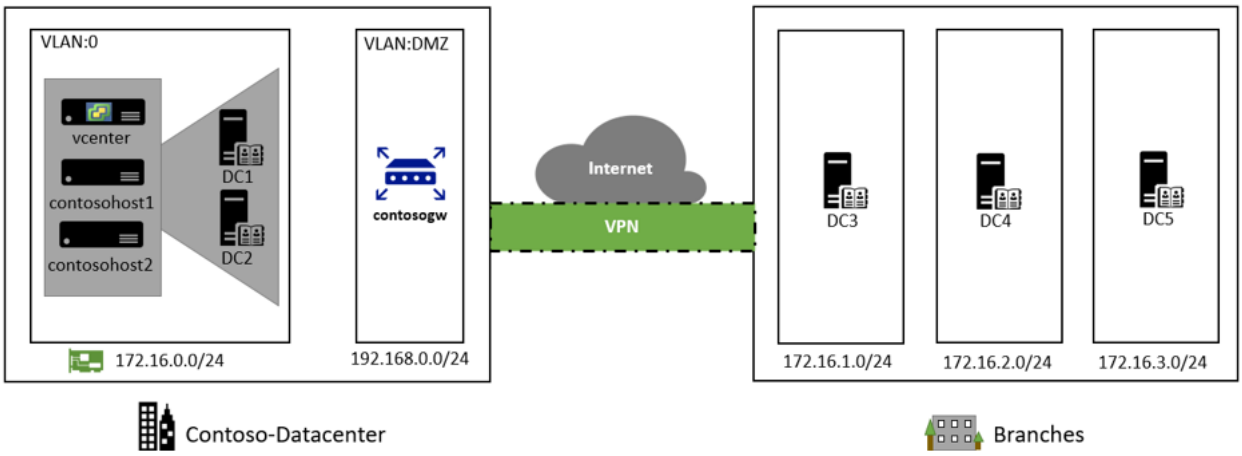
Solution design

After pinning down the company's goals and requirements, Contoso designs and reviews a deployment solution and identifies the migration process. The Azure services that it will use for the migration also are identified.

Current architecture

- Contoso has one main datacenter (`contoso-datacenter`). The datacenter is located in New York City in the eastern United States.
- Contoso has three additional local branches across the United States.

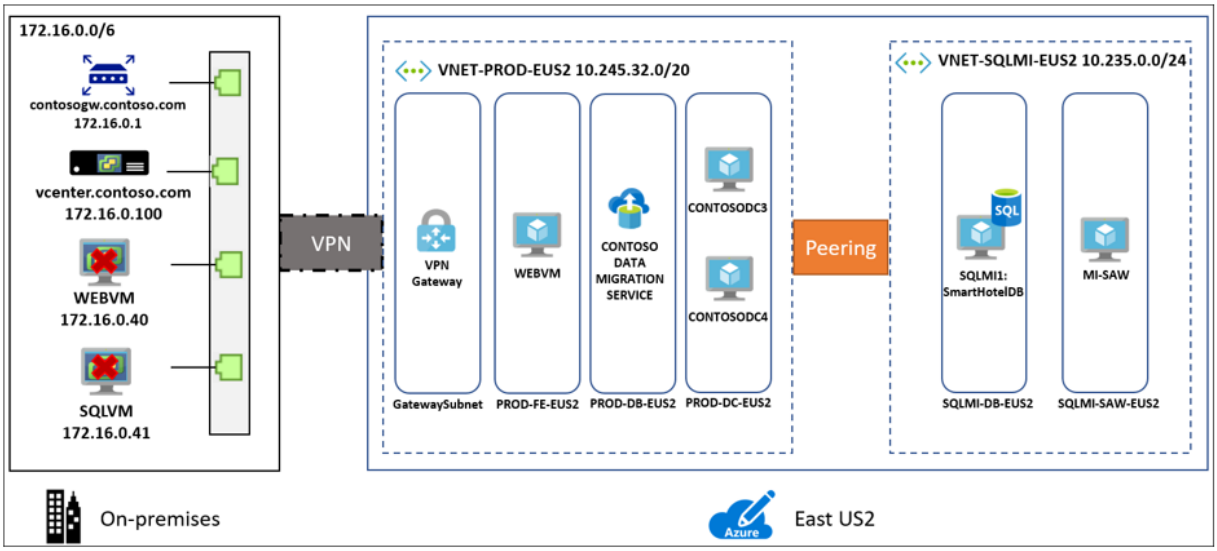
- The main datacenter is connected to the internet with a fiber-optic Metro Ethernet connection (500 megabits per second).
- Each branch is connected locally to the internet by using business-class connections with IPsec VPN tunnels back to the main datacenter. The setup allows Contoso's entire network to be permanently connected and optimizes internet connectivity.
- The main datacenter is fully virtualized with VMware. Contoso has two ESXi 6.5 virtualization hosts that are managed by vCenter Server 6.5.
- Contoso uses Active Directory for identity management. Contoso uses DNS servers on the internal network.
- Contoso has an on-premises domain controller (`contosodc1`).
- The domain controllers run on VMware VMs. The domain controllers at local branches run on physical servers.
- The SmartHotel360 application is tiered across two VMs (`WEBVM` and `SQLVM`) that are located on a VMware ESXi version 6.5 host (`contosohost1.contoso.com`).
- The VMware environment is managed by vCenter Server 6.5 (`vcenter.contoso.com`) running on a VM.



Proposed architecture

In this scenario, Contoso wants to migrate its two-tier on-premises travel application as follows:

- Migrate the application database (`SmartHotelDB`) to a SQL managed instance.
- Migrate the front end, `WEBVM`, to an Azure VM.
- The on-premises VMs in the Contoso datacenter will be decommissioned when the migration is finished.



Database considerations

As part of the solution design process, Contoso did a feature comparison between Azure SQL Database and SQL

Managed Instance. The following considerations helped the company decide to use SQL Managed Instance.

- SQL Managed Instance aims to deliver almost 100% compatibility with the latest on-premises SQL Server version. We recommend SQL Managed Instance for customers who are running SQL Server on-premises or on infrastructure as a service (IaaS) VMs and want to migrate their applications to a fully managed service with minimal design changes.
- Contoso is planning to migrate a large number of applications from on-premises to IaaS. Many of these applications are ISV provided. Contoso realizes that using SQL Managed Instance will help ensure database compatibility for these applications, rather than using SQL Database, which might not be supported.
- Contoso can perform a lift-and-shift migration to SQL Managed Instance by using the fully automated Azure Database Migration Service. With this service in place, Contoso can reuse it for future database migrations.
- SQL Managed Instance supports SQL Server Agent, an important component of the SmartHotel360 application. Contoso needs this compatibility. Otherwise, it will have to redesign maintenance plans required by the application.
- With Software Assurance, Contoso can exchange its existing licenses for discounted rates on a SQL managed instance by using the Azure Hybrid Benefit for SQL Server. For this reason, Contoso can save up to 30 percent on SQL Managed Instance.
- SQL Managed Instance is fully contained in the virtual network, so it provides greater isolation and security for Contoso's data. Contoso can get the benefits of the public cloud while keeping the environment isolated from the public internet.
- SQL Managed Instance supports many security features. They include Always Encrypted, dynamic data masking, row-level security, and threat detection.

Solution review

Contoso evaluates the proposed design by putting together a list of pros and cons.

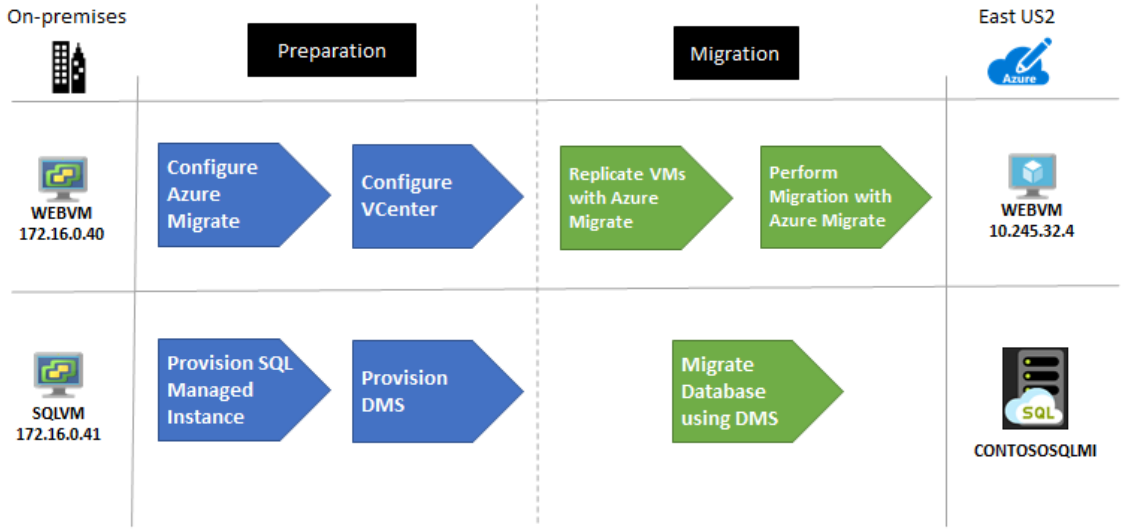
CONSIDERATION	DETAILS
Pros	`WEBVM` will be moved to Azure without changes, which makes the migration simple. SQL Managed Instance supports Contoso's technical requirements and goals. SQL Managed Instance will provide 100 percent compatibility with Contoso's current deployment while moving the company away from SQL Server 2008 R2. Contoso can take advantage of its investment in Software Assurance and use the Azure Hybrid Benefit for SQL Server and Windows Server. Contoso can reuse Azure Database Migration Service for additional future migrations. SQL Managed Instance has built-in fault tolerance that Contoso doesn't need to configure. This feature ensures that the data tier is no longer a single point of failure.

CONSIDERATION	DETAILS
Cons	`WEBVM` is running Windows Server 2008 R2. Although this operating system is supported by Azure, it's no longer a supported platform. To learn more, see Support policy for Microsoft SQL Server products. The web tier remains a single point of failover with only `WEBVM` providing services. Contoso will need to continue supporting the application web tier as a VM rather than moving to a managed service, such as Azure App Service. For the data tier, SQL Managed Instance might not be the best solution if Contoso wants to customize the operating system or the database server, or if the company wants to run third-party applications along with SQL Server. Running SQL Server on an IaaS VM could provide this flexibility.

Migration process

Contoso will migrate the web and data tiers of its SmartHotel360 application to Azure by completing these steps:

1. Contoso already has its Azure infrastructure in place, so it just needs to add a couple of specific Azure components for this scenario.
2. The data tier will be migrated by using Azure Database Migration Service. This service connects to the on-premises SQL Server VM across a Site-to-Site VPN connection between the Contoso datacenter and Azure. The service then migrates the database.
3. The web tier will be migrated by using a lift-and-shift migration by using Azure Migrate. The process entails preparing the on-premises VMware environment, setting up and enabling replication, and migrating the VMs by failing them over to Azure.



Azure services

SERVICE	DESCRIPTION	COST
Azure Database Migration Service	Azure Database Migration Service enables seamless migration from multiple database sources to Azure data platforms with minimal downtime.	Learn about supported regions and Azure Database Migration Service pricing.
Azure SQL Managed Instance	SQL Managed Instance is a managed database service that represents a fully managed SQL Server instance in the Azure cloud. It uses the same code as the latest version of SQL Server Database Engine and has the latest features, performance improvements, and security patches.	Using a SQL managed instance running in Azure incurs charges based on capacity. Learn more about SQL Managed Instance pricing.
Azure Migrate	Contoso uses Azure Migrate to assess its VMware VMs. Azure Migrate assesses the migration suitability of the machines. It provides sizing and cost estimates for running in Azure.	Azure Migrate is available at no additional charge. They might incur charges depending on the tools (first-party or independent software vendor) they decide to use for assessment and migration. Learn more about Azure Migrate pricing.

Prerequisites

Contoso and other users must meet the following prerequisites for this scenario.

REQUIREMENTS	DETAILS
Azure subscription	Contoso already created a subscription in the first article in this series. If you don't have an Azure subscription, create a free account. If you create a free account, you're the administrator of your subscription and can perform all actions. If you use an existing subscription and you're not the administrator of the subscription, work with the admin to assign you Owner or Contributor permissions to the necessary resource groups and resources.
Azure infrastructure	Contoso set up its Azure infrastructure as described in Azure infrastructure for migration.
On-premises servers	The on-premises vCenter Server should be running version 5.5, 6.0, or 6.5. An ESXi host should be running version 5.5, 6.0, or 6.5. One or more VMware VMs should be running on the ESXi host.
On-premises VMs	Review Linux machines that are endorsed to run on Azure.

REQUIREMENTS	DETAILS
Database Migration Service	For Azure Database Migration Service, you need a compatible on-premises VPN device. You must be able to configure the on-premises VPN device. It must have an external-facing public IPv4 address. The address can't be located behind a NAT device. Make sure you can access your on-premises SQL Server database. Windows Firewall should be able to access the source database engine. Learn how to configure Windows Firewall for database engine access. If there's a firewall in front of your database machine, add rules to allow access to the database and files via SMB port 445. The credentials that are used to connect to the source SQL Server instance and that target SQL Managed Instance must be members of the sysadmin server role. You need a network share in your on-premises database that Azure Database Migration Service can use to back up the source database. Make sure that the service account running the source SQL Server instance has write permissions on the network share. Make a note of a Windows user and password that has full control permissions on the network share. Azure Database Migration Service impersonates these user credentials to upload backup files to the Azure Storage container. The SQL Server Express installation process sets the TCP/IP protocol to **Disabled** by default. Make sure that it's enabled.

Scenario steps

Here's how Contoso plans to set up the deployment:

- **Step 1: Prepare a SQL managed instance.** Contoso needs an existing managed instance to which the on-premises SQL Server database will migrate.
- **Step 2: Prepare Azure Database Migration Service.** Contoso must register the database migration provider, create an instance, and then create a Database Migration Service project. Contoso also must set up a shared access signature (SAS) uniform resource identifier (URI) for the Database Migration Service instance. An SAS URI provides delegated access to resources in Contoso's storage account so that Contoso can grant limited permissions to storage objects. Contoso sets up an SAS URI so that Azure Database Migration Service can access the storage account container to which the service uploads the SQL Server backup files.
- **Step 3: Prepare Azure for the Azure Migrate: Server Migration tool.** Contoso adds the server migration tool to its Azure Migrate project.
- **Step 4: Prepare on-premises VMware for Azure Migrate: Server Migration.** Contoso prepares accounts for VM discovery and prepares to connect to Azure VMs after migration.
- **Step 5: Replicate the on-premises VMs.** Contoso sets up replication and starts replicating VMs to Azure Storage.

- **Step 6: Migrate the database via Azure Database Migration Service.** Contoso migrates the database.
- **Step 7: Migrate the VMs with Azure Migrate: Server Migration.** Contoso runs a test migration to make sure everything's working and then runs a full migrate to move the VM to Azure.

Step 1: Prepare a SQL managed instance

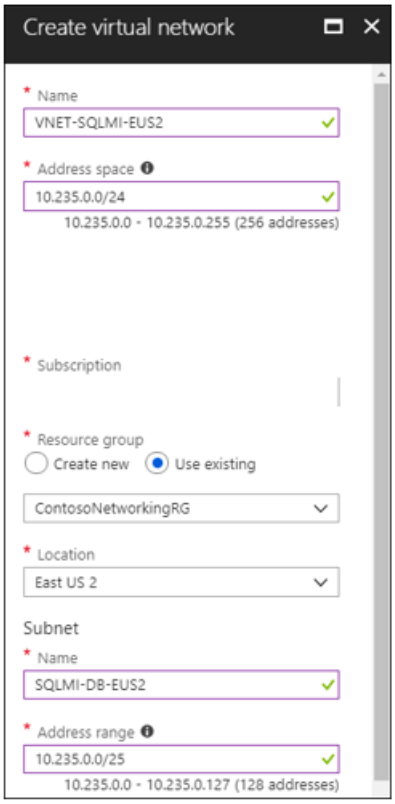
To set up a SQL managed instance, Contoso needs a subnet that meets the following requirements:

- The subnet must be dedicated. It must be empty. It can't contain any other cloud service. The subnet can't be a gateway subnet.
- After the managed instance is created, Contoso shouldn't add resources to the subnet.
- The subnet can't have a network security group associated with it.
- The subnet must have a user-defined route table. The only route assigned should be `0.0.0.0/0` next-hop internet.
- If an optional custom DNS is specified for the virtual network, the virtual IP address `168.63.129.16` for the recursive resolvers in Azure must be added to the list. Learn how to configure custom DNS for a SQL managed instance.
- The subnet must not have a service endpoint (storage or SQL) associated with it. Service endpoints should be disabled on the virtual network.
- The subnet must have a minimum of 16 IP addresses. Learn how to size the managed instance subnet.
- In Contoso's hybrid environment, custom DNS settings are required. Contoso configures DNS settings to use one or more of the company's Azure DNS servers. Learn more about DNS customization.

Set up a virtual network for the managed instance

To set up the virtual network, the Contoso admins:

1. Create a new virtual network (`VNET-SQLMI-EU2`) in the primary region (`East US 2`). It adds the virtual network to the `ContosoNetworkingRG` resource group.
2. Assign an address space of `10.235.0.0/24` . They ensure that the range doesn't overlap with any other networks in its enterprise.
3. Add two subnets to the network:
   - `SQLMI-DS-EUS2` (`10.235.0.0/25`).
   - `SQLMI-SAW-EUS2` (`10.235.0.128/29`). This subnet is used to attach a directory to the managed instance.



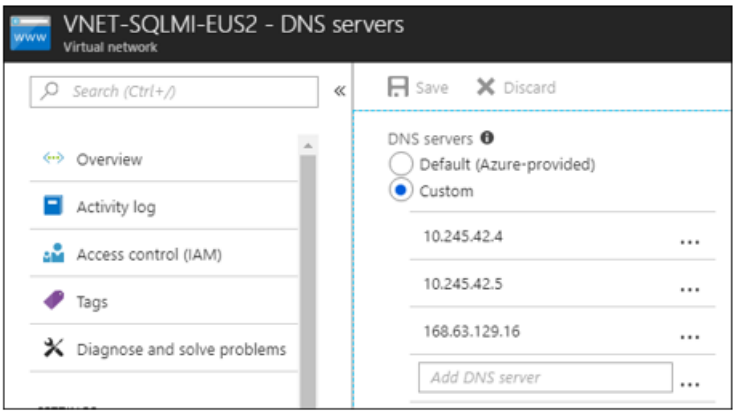
4. After the virtual network and subnets are deployed, they peer networks as follows:

   - Peers `VNET-SQLMI-EUS2` with `VNET-HUB-EUS2` (the hub virtual network in `East US 2`).
   - Peers `VNET-SQLMI-EUS2` with `VNET-PROD-EUS2` (the production network).

NAME	PEERING STATUS	PEER
VNET-SQLMI-EUS2-to-VNET-PROD-EUS2	Connected	VNET-PROD-EUS2
VNET-SQLMI-EUS2-to-VNET-HUB-EUS2	Connected	VNET-HUB-EUS2

5. Set custom DNS settings. DNS points first to Contoso's Azure domain controllers. Azure DNS is secondary. The Contoso Azure domain controllers are located as follows:

   - Located in the `PROD-DC-EUS2` subnet, in the `East US 2` production network (`VNET-PROD-EUS2`).
   - `CONTOSODC3` address: `10.245.42.4` .
   - `CONTOSODC4` address: `10.245.42.5` .
   - Azure DNS resolver: `168.63.129.16` .



Need more help?

- Read the SQL Managed Instance overview.
- Learn how to create a virtual network for a SQL managed instance.
- Learn how to set up peering.
- Learn how to update Azure Active Directory DNS settings.

Set up routing

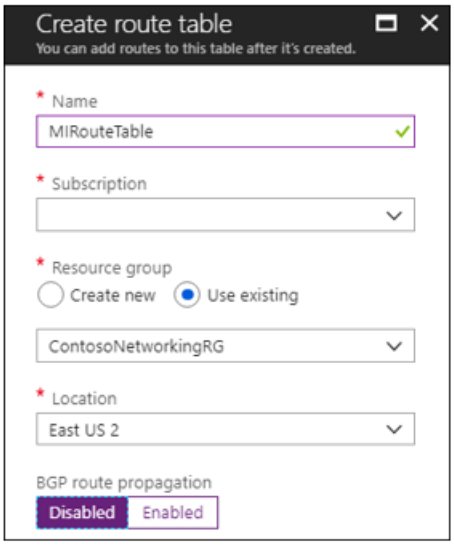
The managed instance is placed in a private virtual network. Contoso needs a route table for the virtual network to communicate with the Azure management service. If the virtual network can't communicate with the service that manages it, the virtual network becomes inaccessible.

Contoso considers these factors:

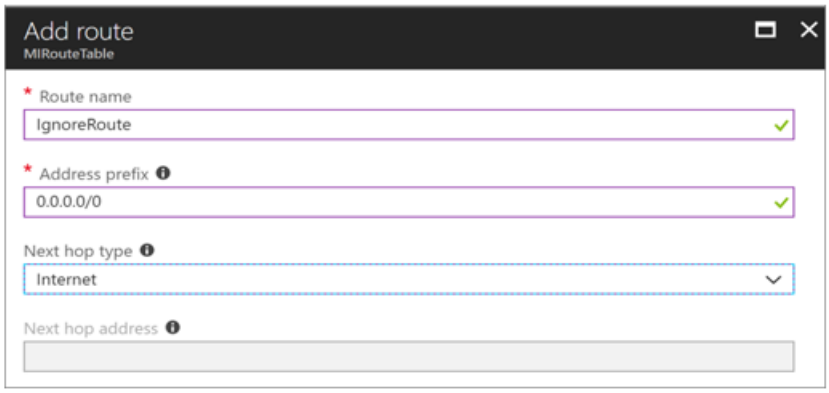
- The route table contains a set of rules (routes) that specify how packets sent from the managed instance should be routed in the virtual network.
- The route table is associated with subnets where managed instances are deployed. Each packet that leaves a subnet is handled based on the associated route table.
- A subnet can be associated with only one route table.
- There are no additional charges for creating route tables in Microsoft Azure.

To set up routing, the Contoso admins do the following steps:

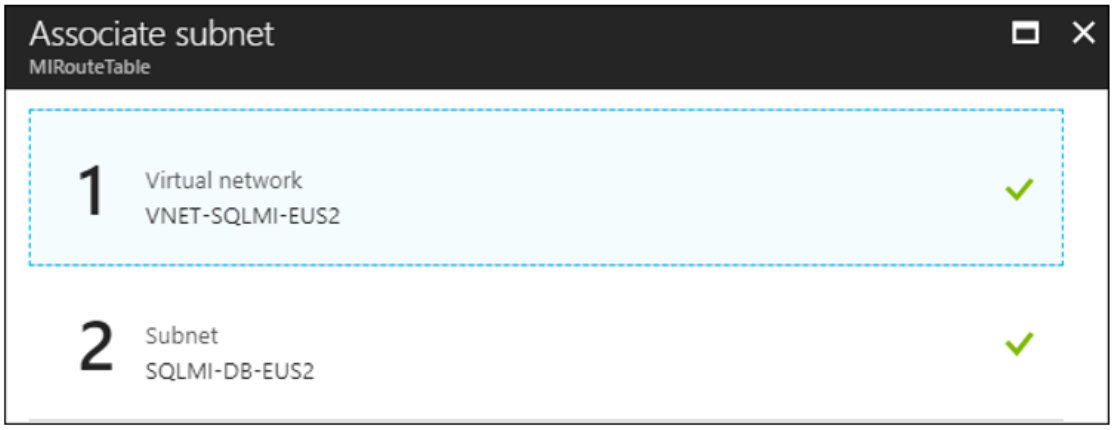
1. Create a user-defined route table in the `ContosoNetworkingRG` resource group.



2. To comply with SQL Managed Instance requirements, after the route table (`MIRouteTable`) is deployed, they add a route that has an address prefix of `0.0.0.0/0`. The **Next hop type** option is set to **Internet**.



3. Associate the route table with the `SQLMI-DB-EUS2` subnet (in the `VNET-SQLMI-EUS2` network).



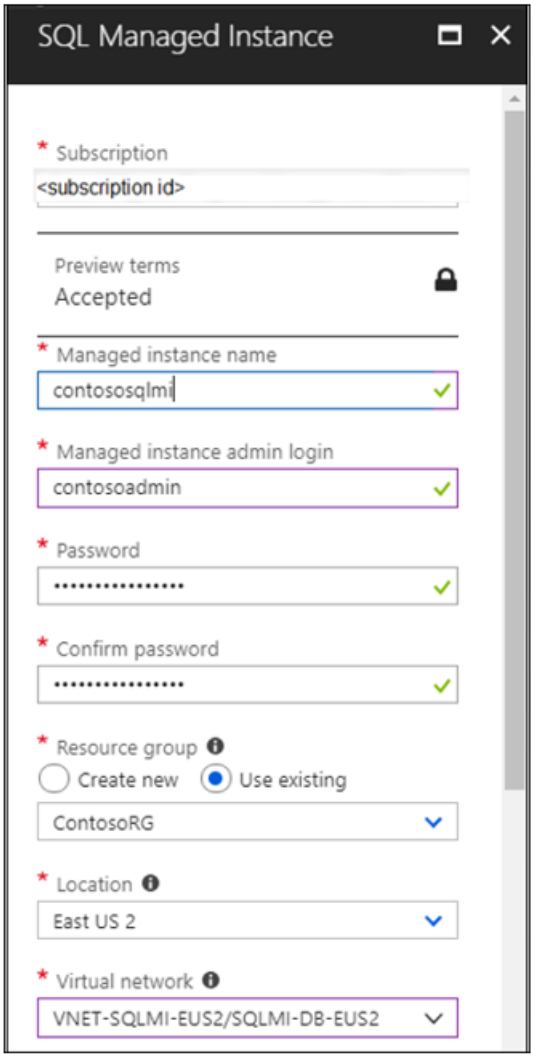
Need more help?

Learn how to set up routes for a managed instance.

Create a managed instance

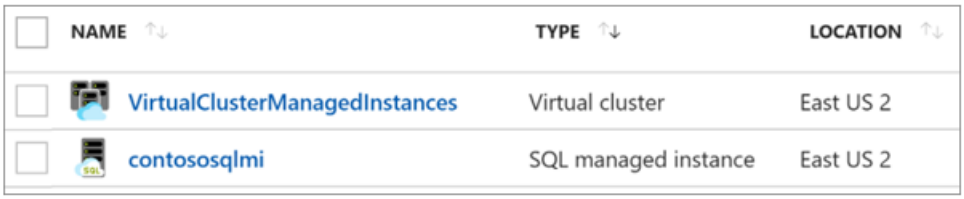
Now the Contoso admins can provision a SQL managed instance:

1. Because the managed instance serves a business application, they deploy the managed instance in the company's primary region (`East US 2`). They add the managed instance to the `ContosoRG` resource group.
2. They select a pricing tier, size compute, and storage for the instance. Learn more about SQL Managed Instance pricing.



3. After the managed instance is deployed, two new resources appear in the `ContosoRG` resource group:

   - The SQL managed instance.
   - A virtual cluster in case Contoso has multiple managed instances.



Need more help?

Learn how to provision a managed instance.

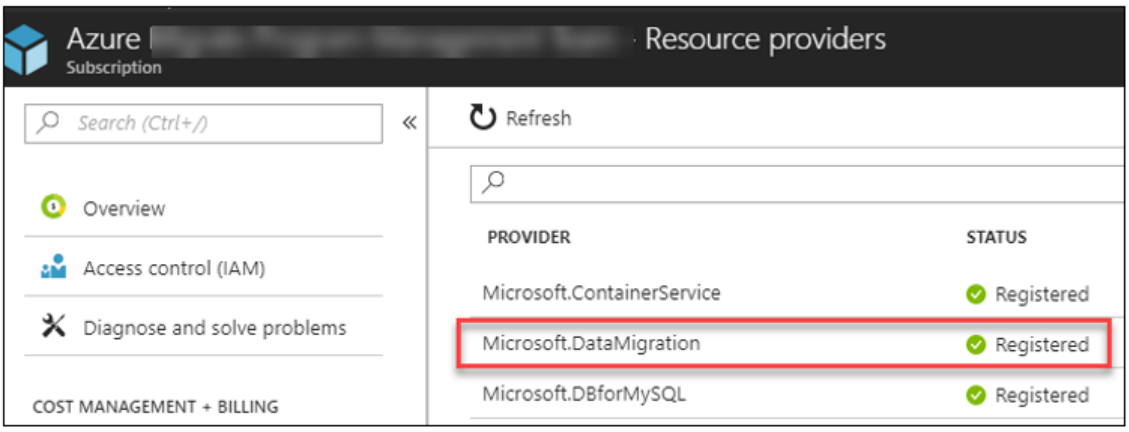
Step 2: Prepare Azure Database Migration Service

To prepare Azure Database Migration Service, the Contoso admins need to do a few things:

- Register the Database Migration Service provider in Azure.
- Grant permission for Database Migration Service to access Azure Storage for uploading the backup files that are used to migrate a database. To provide access to Azure Storage, create an Azure Blob Storage container. Generate an SAS URI for the Blob Storage container.
- Create an Azure Database Migration Service project.

They complete the following steps:

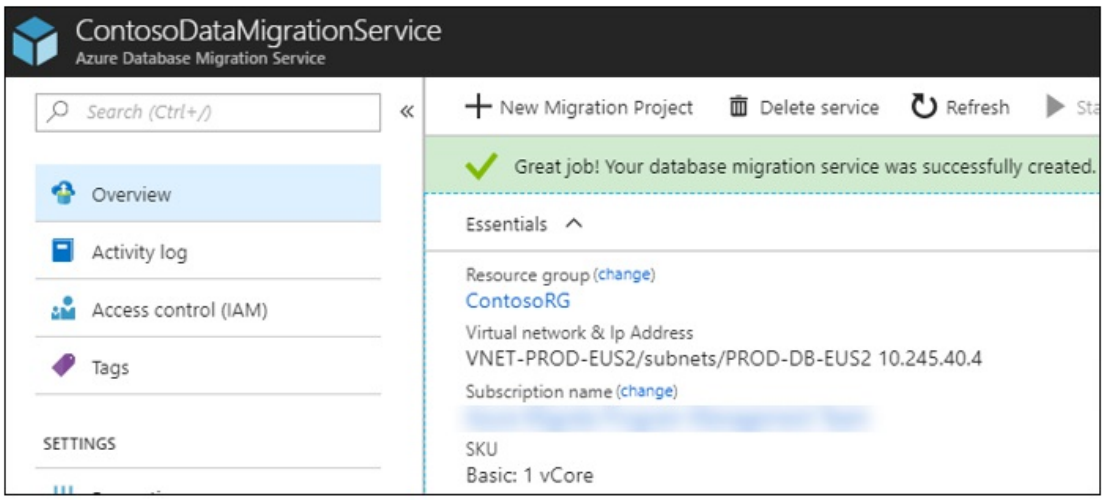
1. Register the database migration provider under its subscription.



2. Create an Azure Blob Storage container. Contoso generates an SAS URI so that Azure Database Migration Service can access it.



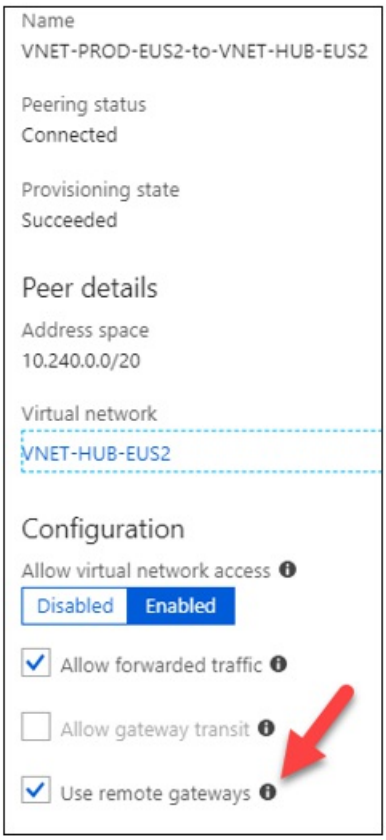
3. Create an Azure Database Migration Service instance.



4. Place the Database Migration Service instance in the `PROD-DC-EUS2` subnet of the `VNET-PROD-DC-EUS2` virtual network.

   - The instance is placed here because the service must be in a virtual network that can access the on-premises SQL Server VM via a VPN gateway.

- `VNET-PROD-EUS2` is peered to `VNET-HUB-EUS2` and is allowed to use remote gateways. The **Use remote gateways** option ensures that the instance can communicate as required.



Need more help?

- Learn how to set up Azure Database Migration Service.
- Learn how to create and use SAS.

Step 3: Prepare Azure for the Azure Migrate: Server Migration tool

Here are the Azure components Contoso needs to migrate the VMs to Azure:

- A virtual network in which Azure VMs will be located when they're created during migration.
- The Azure Migrate: Server Migration tool provisioned.

The Contoso admins set up these components:

1. Set up a network. Contoso already set up a network that can be used for Azure Migrate: Server Migration when it deployed the Azure infrastructure.
    - The SmartHotel360 application is a production application, and the VMs will be migrated to the Azure production network (`VNET-PROD-EUS2`) in the primary region (`East US 2`).
    - Both VMs will be placed in the `ContosoRG` resource group, which is used for production resources.
    - The application front-end VM (`WEBVM`) will migrate to the front-end subnet (`PROD-FE-EUS2`) of the production network.
    - The application database VM (`SQLVM`) will migrate to the database subnet (`PROD-DB-EUS2`) of the production network.

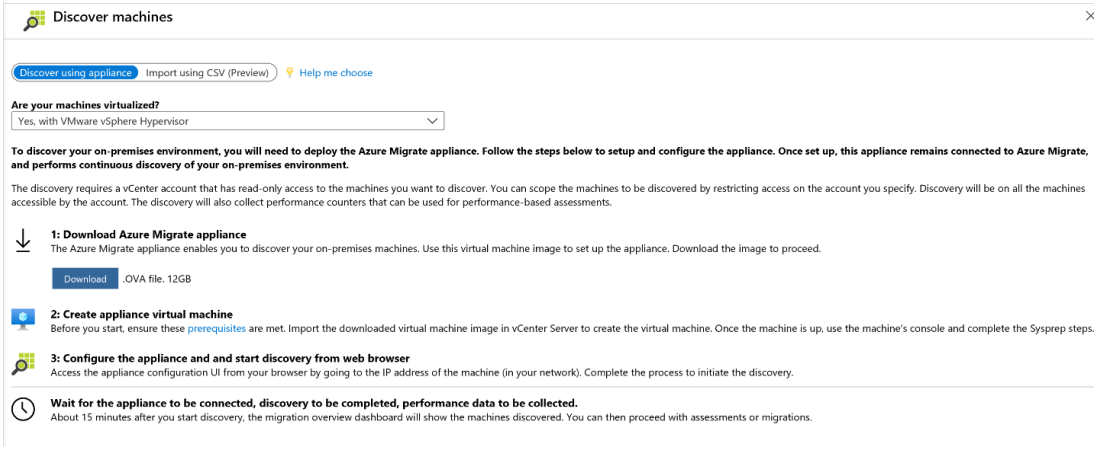
Step 4: Prepare on-premises VMware for Azure Migrate: Server Migration

Here are the Azure components Contoso needs to migrate the VMs to Azure:

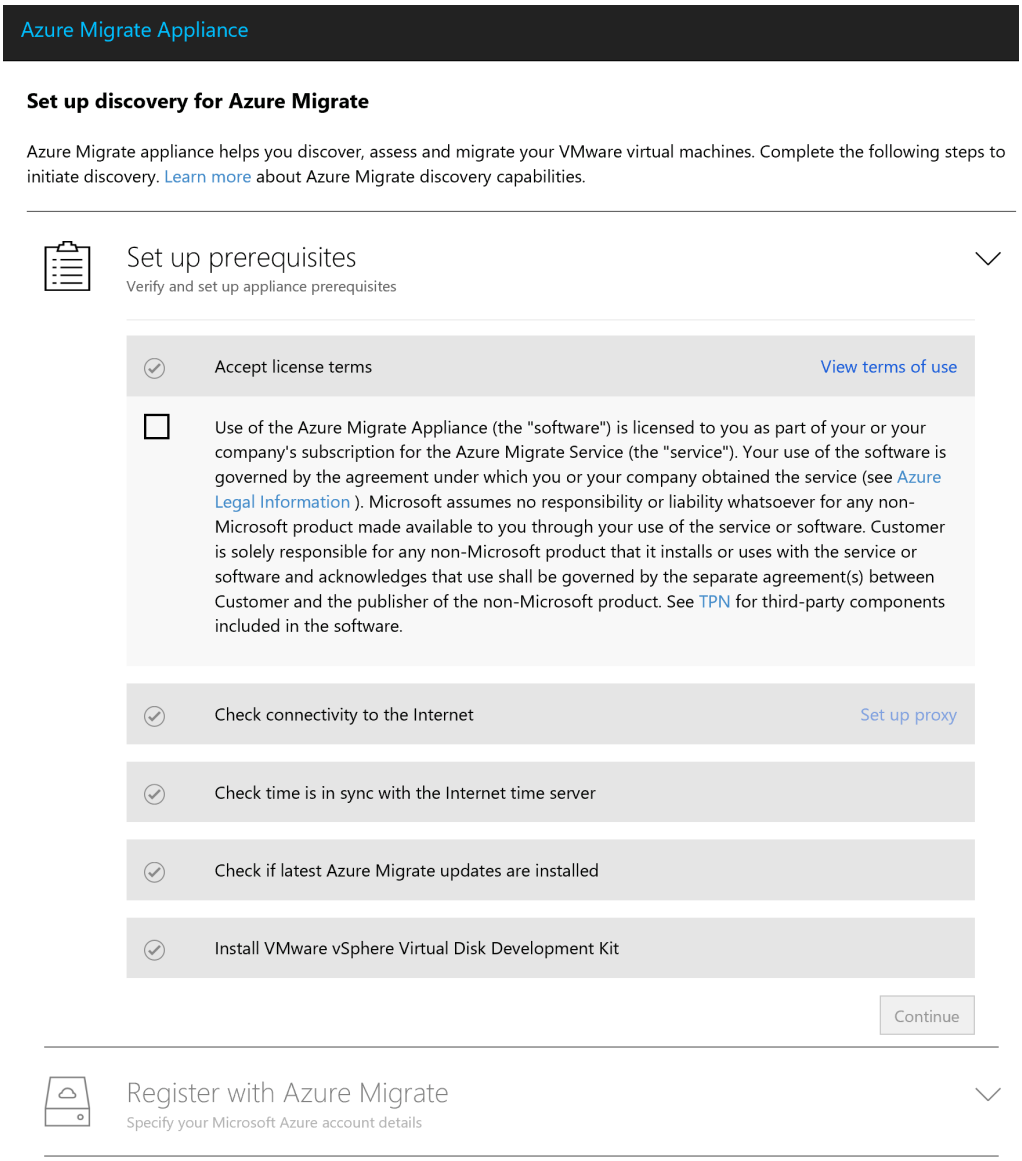
- A virtual network in which Azure VMs will be located when they're created during migration.
- The Azure Migrate appliance, provisioned and configured.

The Contoso admins set up these components by following these steps:

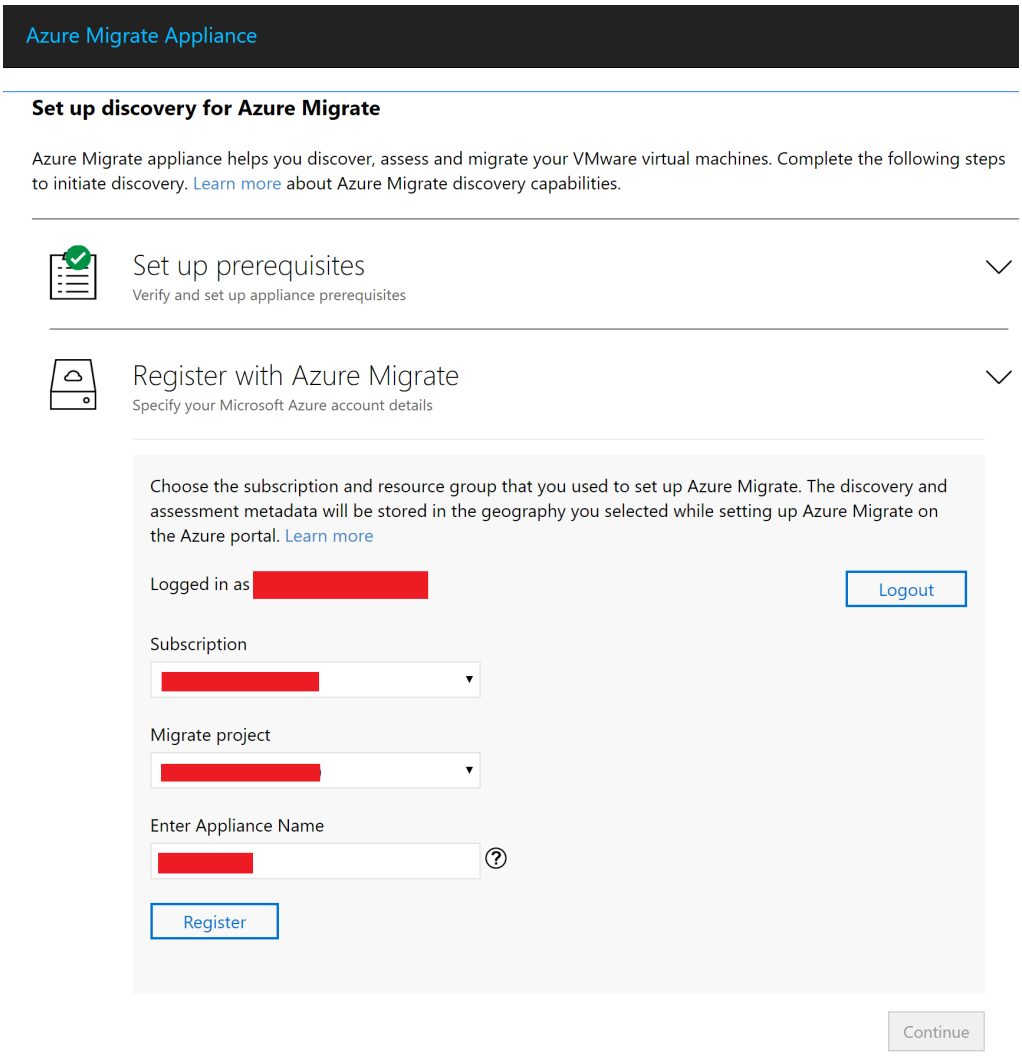
1. Set up a network. Contoso already set up a network that can be used for Azure Migrate: Server Migration when it deployed the Azure infrastructure.
    - The SmartHotel360 application is a production application, and the VMs will be migrated to the Azure production network (`VNET-PROD-EUS2`) in the primary region (`East US 2`).
    - Both VMs will be placed in the `ContosoRG` resource group, which is used for production resources.
    - The application front-end VM (`WEBVM`) will migrate to the front-end subnet (`PROD-FE-EUS2`) in the production network.
    - The application database VM (`SQLVM`) will migrate to the database subnet (`PROD-DB-EUS2`) in the production network.
2. Provision the Azure Migrate appliance.
    a. From Azure Migrate, download the OVA image and import it into VMware.



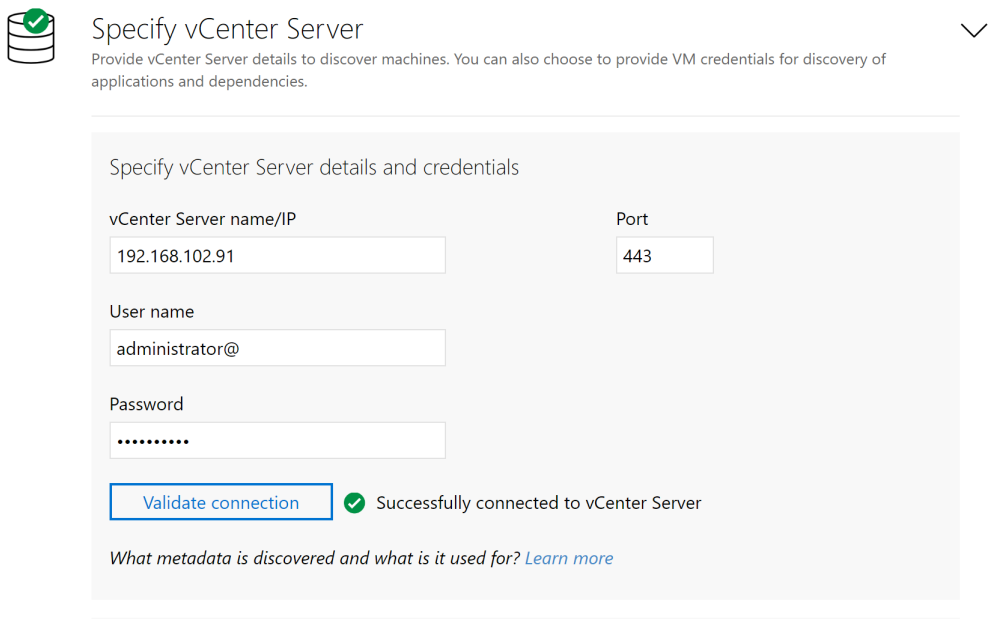
    b. Start the imported image and configure the tool by following these steps:
        a. Set up the prerequisites.



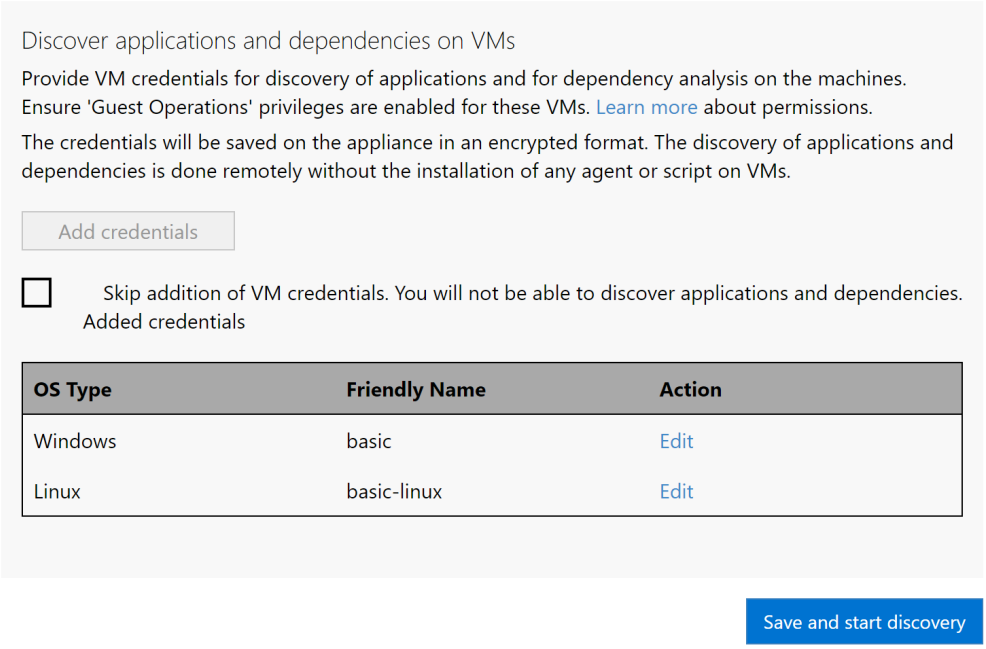
b. Point the tool to the Azure subscription.



c. Set the VMware vCenter credentials.



d. Add any Linux-based or Windows-based credentials for discovery.



3. After configuration, it takes some time for the tool to enumerate all the virtual machines. After the process is finished, the Contoso admins can see the VMs populated in the Azure Migrate tool in Azure.

Need more help?

Learn about how to set up the Azure Migrate appliance.

Prepare on-premises VMs

After migration, Contoso wants to connect to the Azure VMs and allow Azure to manage the VMs. The Contoso admins must do the following steps before migration:

1. For access over the internet, they:
   - Enable RDP or SSH on the on-premises VM before migration.
   - Ensure that TCP and UDP rules are added for the **Public** profile.
   - Check that RDP or SSH is allowed in the operating system firewall.
2. For access over Site-to-Site VPN, they:
   - Enable RDP or SSH on the on-premises VM before migration.
   - Check that RDP or SSH is allowed in the operating system firewall.
   - For Windows, set the operating system's SAN policy on the on-premises VM to **OnlineAll**.
3. They install the Azure agent:
   - Azure Linux agent
   - Azure Windows agent
4. Other considerations:
   - For Windows, there should be no Windows updates pending on the VM when triggering a migration. If there are, they won't be able to sign in to the VM until the update finishes.
   - After migration, they can check **boot diagnostics** to view a screenshot of the VM. If this doesn't work, they should verify that the VM is running and review these troubleshooting tips.

Need more help?

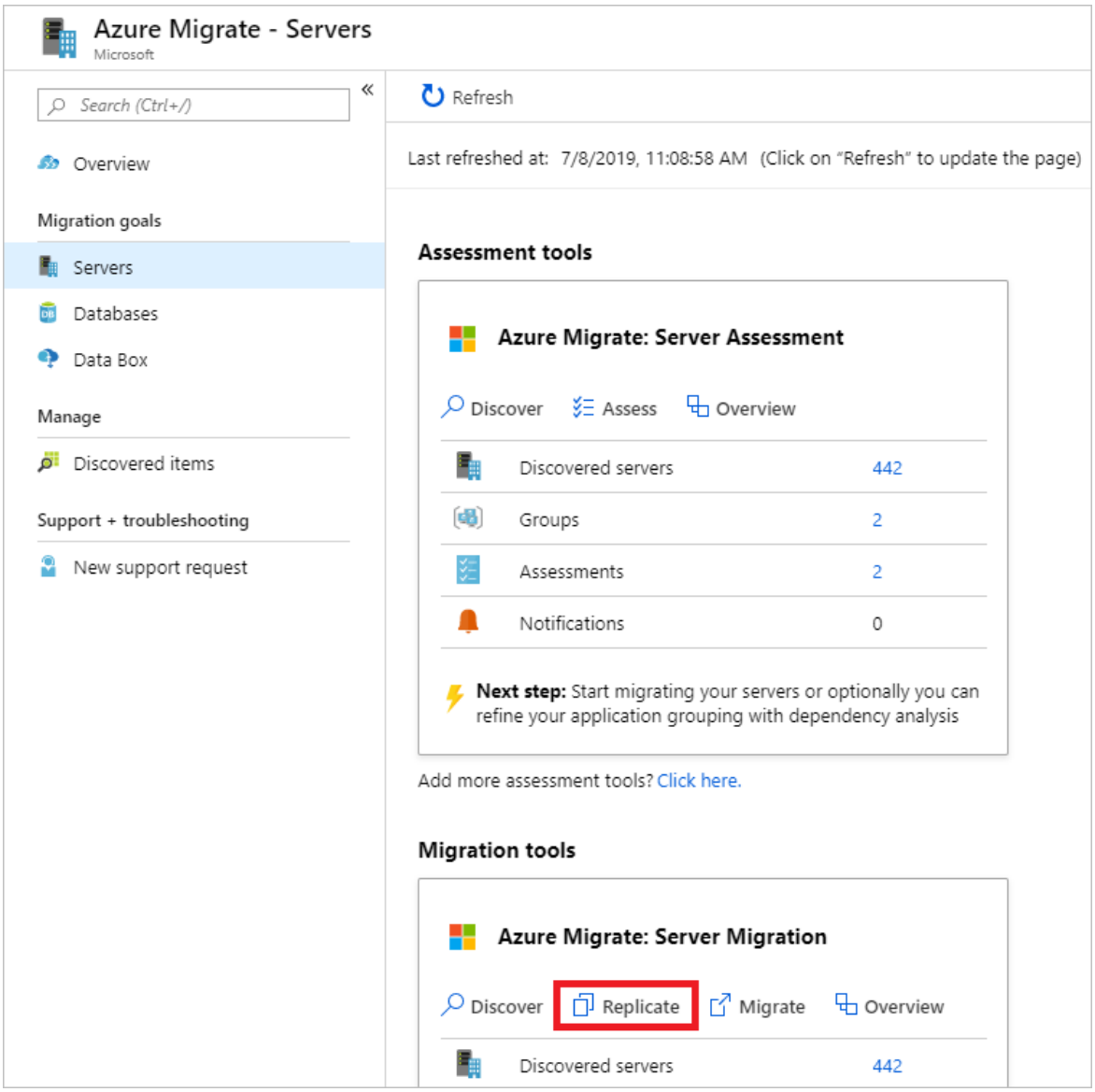
Learn about how to prepare VMs for migration.

Step 5: Replicate the on-premises VMs

Before the Contoso admins can run a migration to Azure, they need to set up and enable replication.

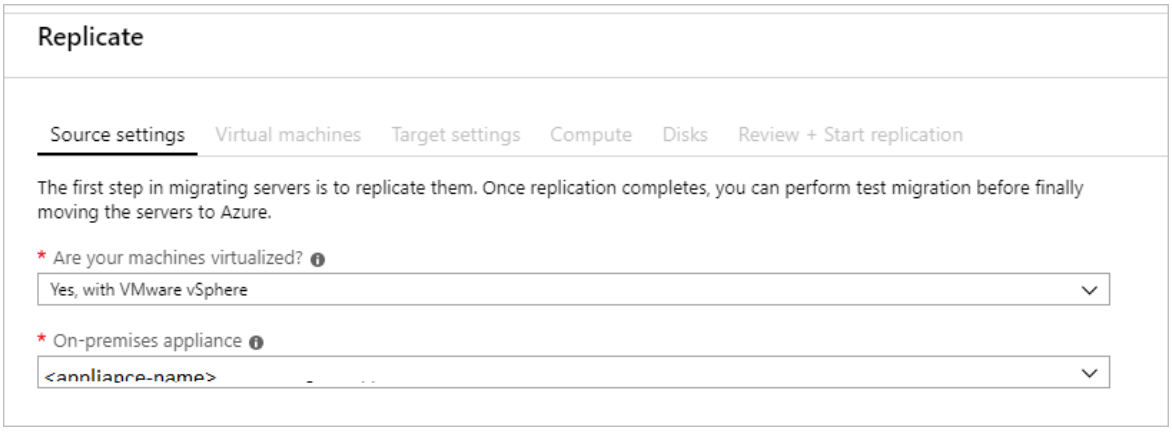
With discovery completed, they can begin replication of VMware VMs to Azure.

1. In the Azure Migrate project, they go to **Servers** > **Azure Migrate: Server Migration**. Then they select **Replicate**.

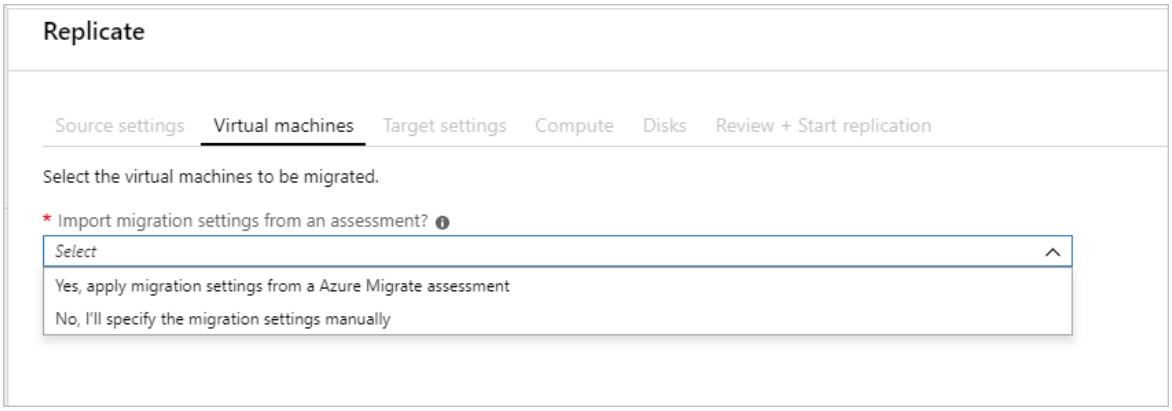


2. In **Replicate** > **Source settings** > **Are your machines virtualized?**, they select **Yes, with VMware vSphere**.

3. In **On-premises appliance**, they select the name of the Azure Migrate appliance that was set up and then select **OK**.



4. In **Virtual machines**, they select the machines they want to replicate:
   - If they've run an assessment for the VMs, they can apply VM sizing and disk type (premium/standard) recommendations from the assessment results. In **Import migration settings from an Azure Migrate assessment?**, they select the **Yes** option.
   - If they didn't run an assessment or they don't want to use the assessment settings, they select the **No** option.
   - If they selected to use the assessment, they select the VM group and assessment name.



5. In **Virtual machines**, they search for VMs as needed and check each VM they want to migrate. Then they select **Next: Target settings**.
6. In **Target settings**, they select the subscription and target region to which they'll migrate. They also specify the resource group in which the Azure VMs will reside after migration. In **Virtual Network**, they select the Azure virtual network/subnet to which the Azure VMs will be joined after migration.
7. In **Azure Hybrid Benefit**, they:
   - Select **No** if they don't want to apply Azure Hybrid Benefit. Then they select **Next**.
   - Select **Yes** if they have Windows Server machines that are covered with active Software Assurance or Windows Server subscriptions and they want to apply the benefit to the machines they're migrating. Then they select **Next**.
8. In **Compute**, they review the VM name, size, OS disk type, and availability set. VMs must conform with Azure requirements.
   - **VM size:** If they're using assessment recommendations, the VM size drop-down list contains the recommended size. Otherwise, Azure Migrate picks a size based on the closest match in the Azure subscription. Alternatively, they can pick a manual size in **Azure VM size.**
   - **OS disk:** They specify the OS (boot) disk for the VM. The OS disk is the disk that has the operating system bootloader and installer.
   - **Availability set:** If the VM should be in an Azure availability set after migration, they specify the set. The set must be in the target resource group specified for the migration.
9. In **Disks**, they specify whether the VM disks should be replicated to Azure. Then they select the disk type (standard SSD/HDD or premium-managed disks) in Azure and select **Next**.
   - They can exclude disks from replication.
   - If disks are excluded, they won't be present on the Azure VM after migration.
10. In **Review + start replication**, they review the settings. Then they select **Replicate** to start the initial replication for the servers.

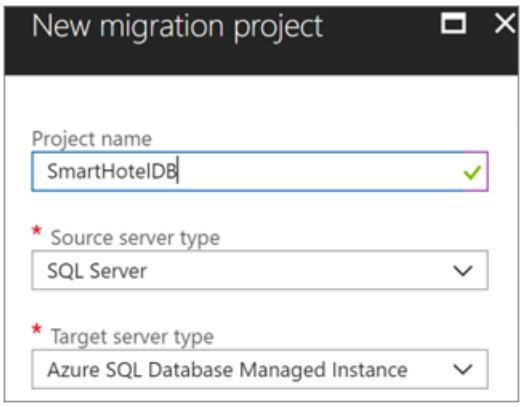
> **NOTE**
>
> Replication settings can be updated any time before replication starts in **Manage** > **Replicating machines**. Settings can't be changed after replication starts.

Step 6: Migrate the database via Azure Database Migration Service

The Contoso admins need to create a Database Migration Service project and then migrate the database.

Create an Azure Database Migration Service project

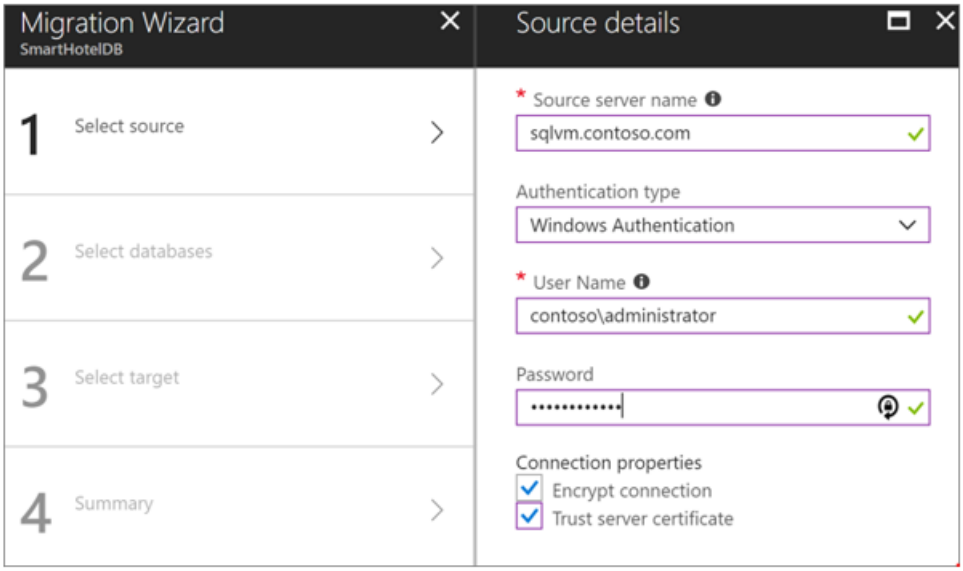
1. The admins create a Database Migration Service project. They select the **SQL Server** source server type and **Azure SQL Managed Instance** as the target.



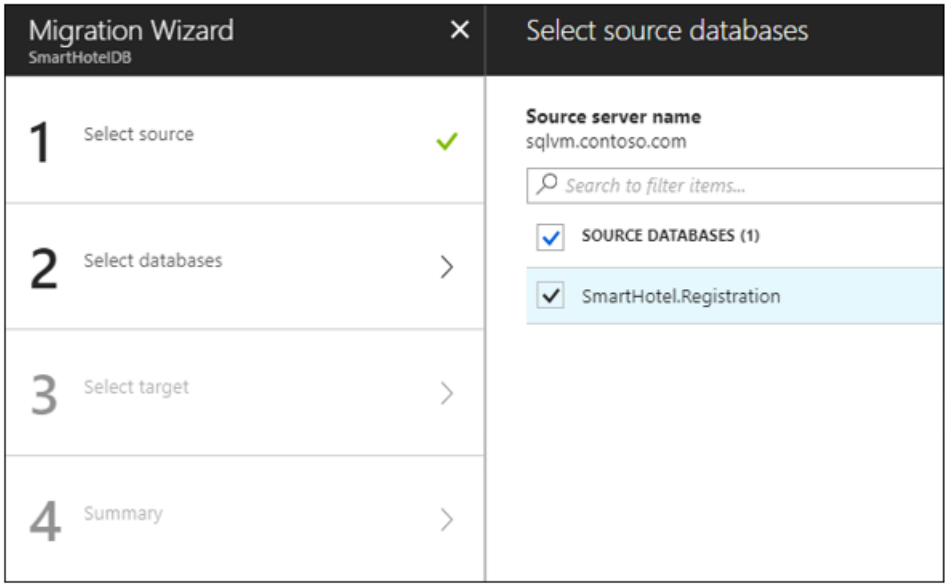
2. The Migration Wizard opens.

Migrate the database

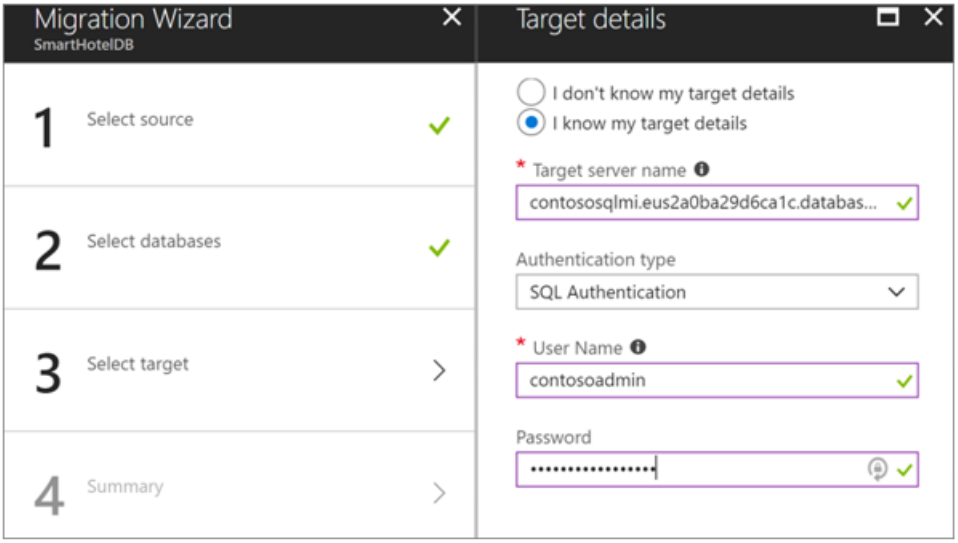
1. In the Migration Wizard, they specify the source VM on which the on-premises database is located. They enter the credentials to access the database.



2. They select the database to migrate (`SmartHotel.Registration`).

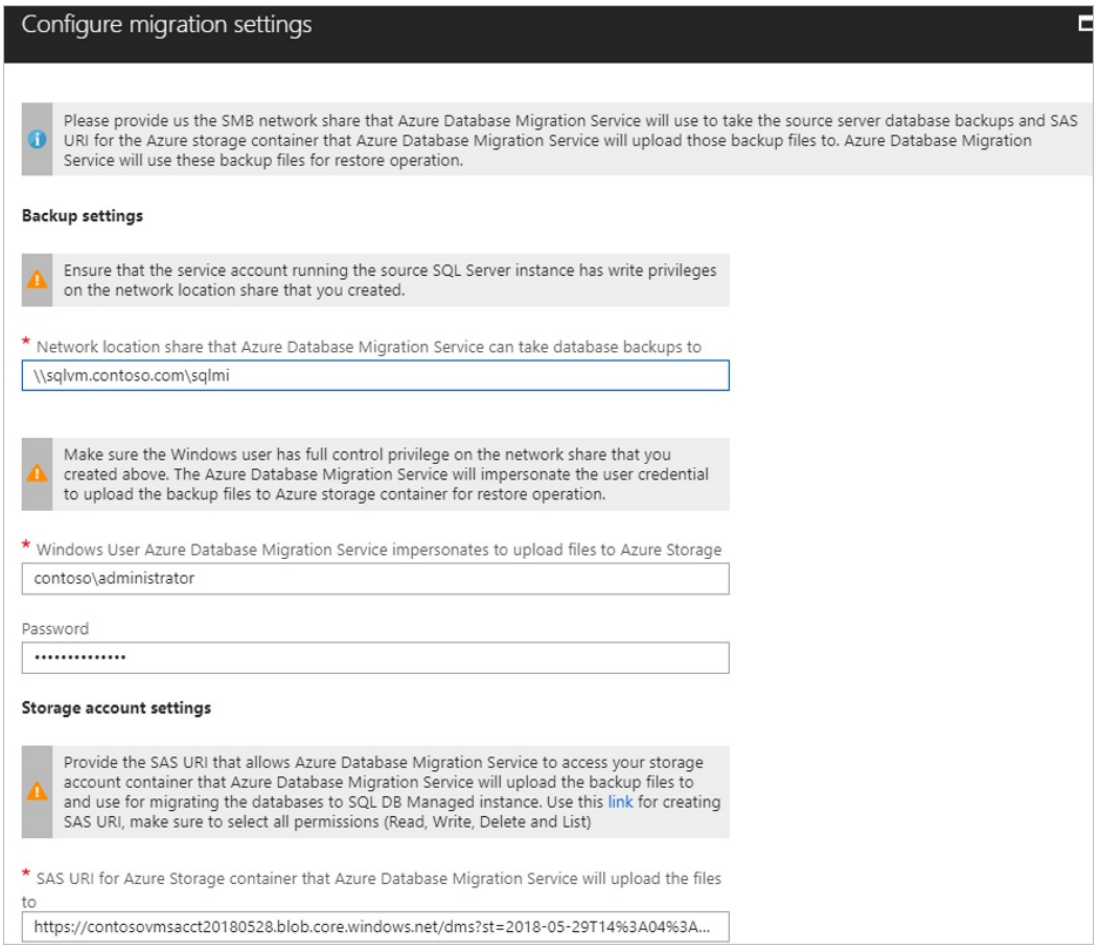


3. For the target, they enter the name of the managed instance in Azure and the access credentials.

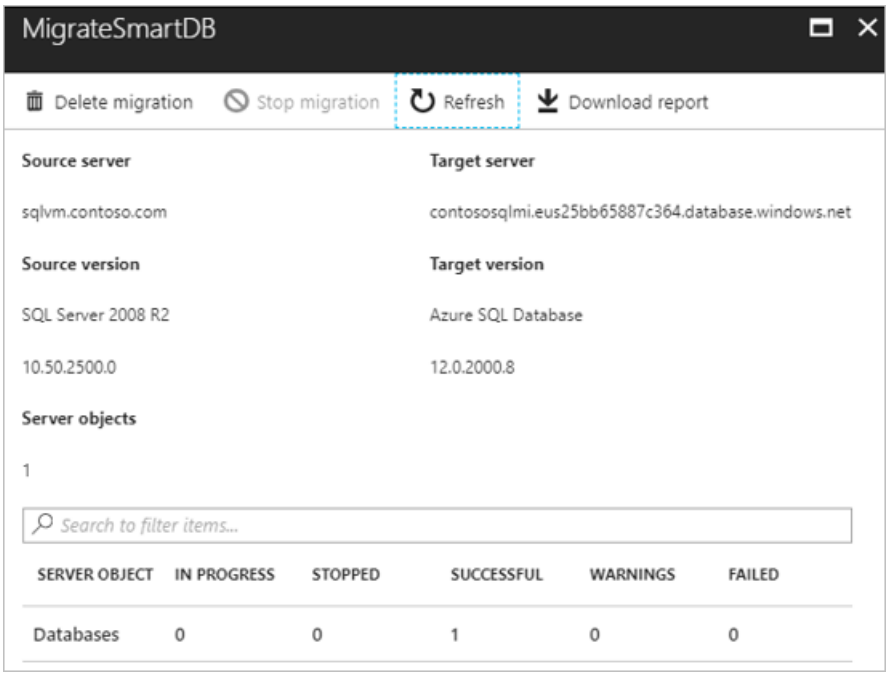


4. In **New Activity** > **Run migration**, they specify settings to run the migration:

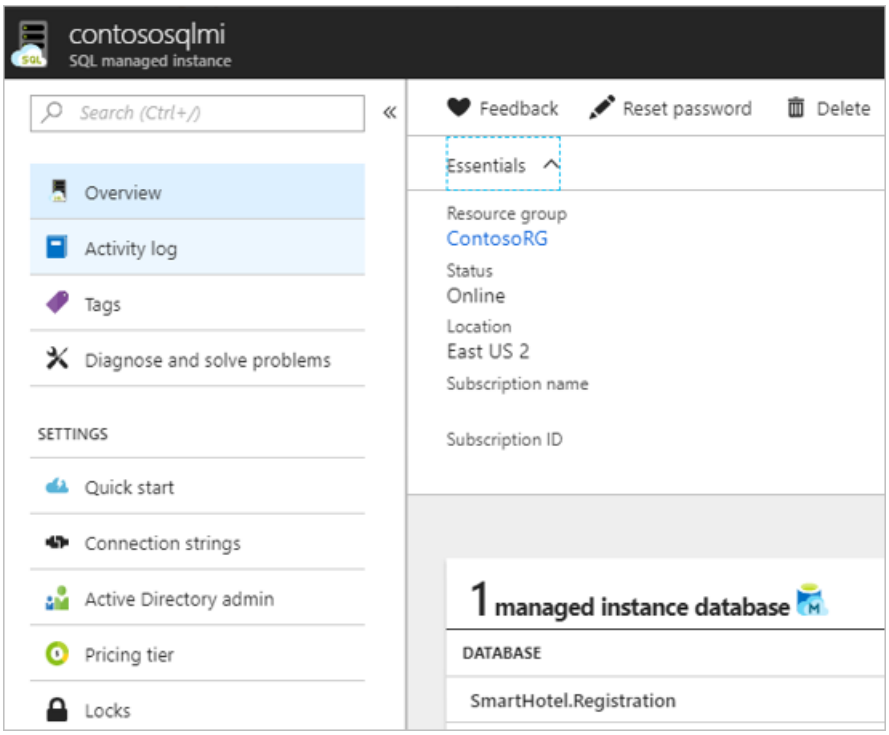
   - Source and target credentials.
   - The database to migrate.
   - The network share created on the on-premises VM. Azure Database Migration Service takes source backups to this share.
     - The service account that runs the source SQL Server instance must have write permissions on this share.
     - The FQDN path to the share must be used.
   - The SAS URI that provides Azure Database Migration Service with access to the storage account container to which the service uploads the backup files for migration.



5. They save the migration settings and then run the migration.
6. In **Overview**, they monitor the migration status.



7. When migration is finished, they verify that the target databases exist on the managed instance.

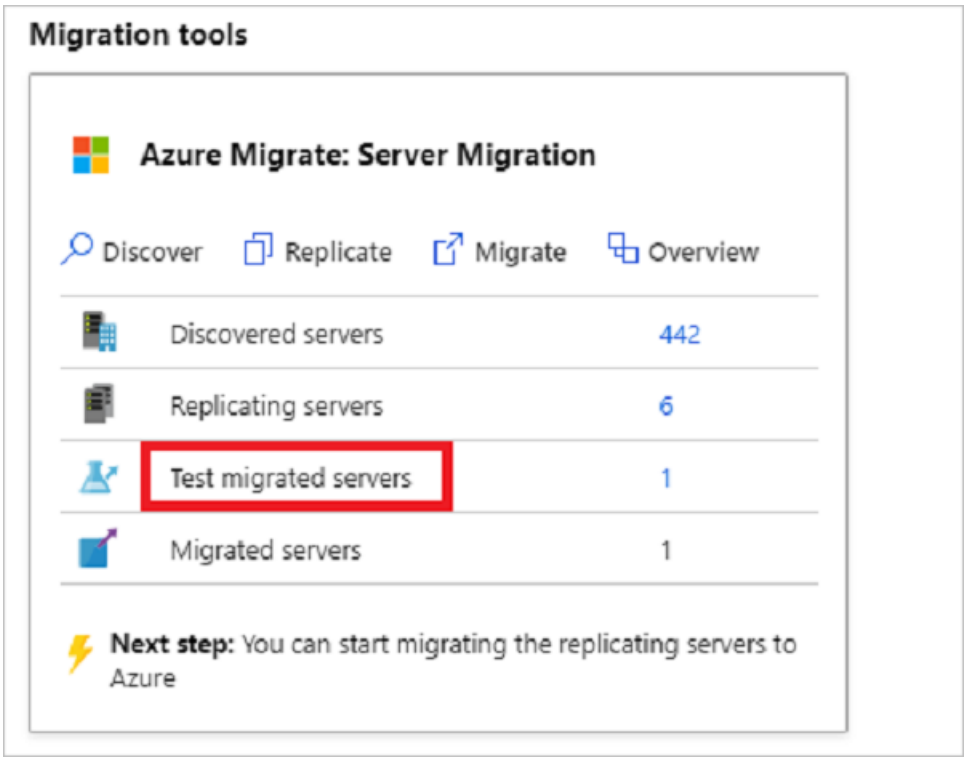


Step 7: Migrate the VMs with Azure Migrate: Server Migration

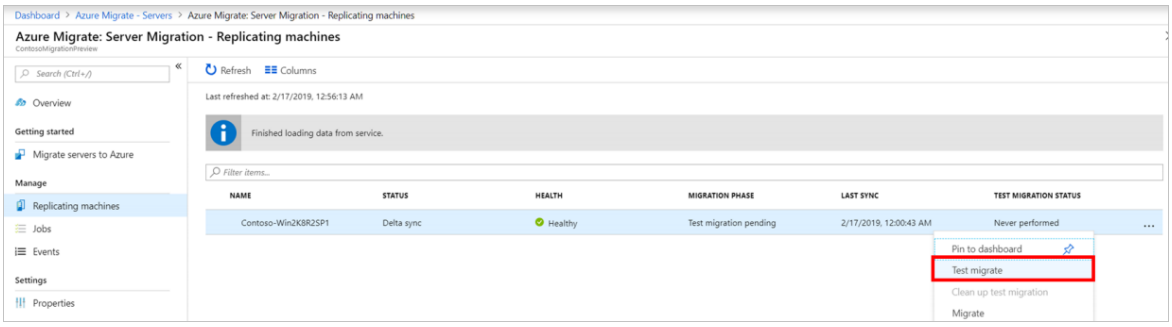
The Contoso admins run a quick test migration and verify the VM is working properly. Then they migrate the VM.

Run a test migration

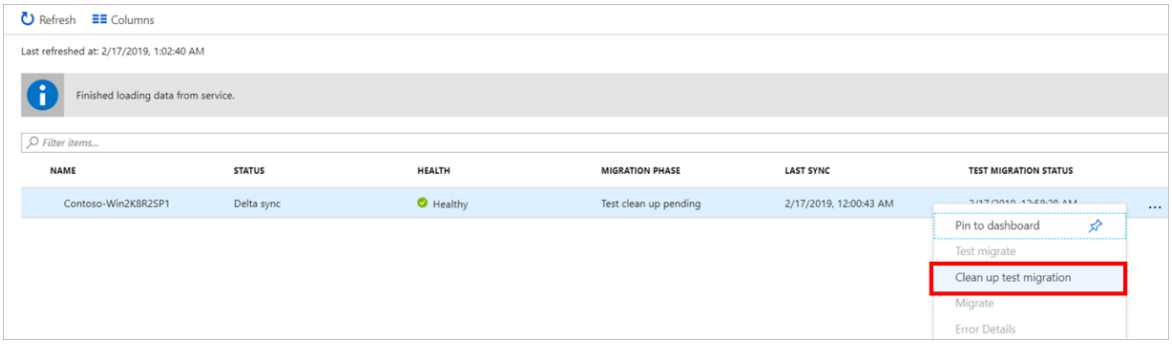
1. In **Migration goals** > **Servers** > **Azure Migrate: Server Migration**, they select **Test migrated servers**.



2. They select and hold (or right-click) the VM to test, and then they select **Test migrate**.



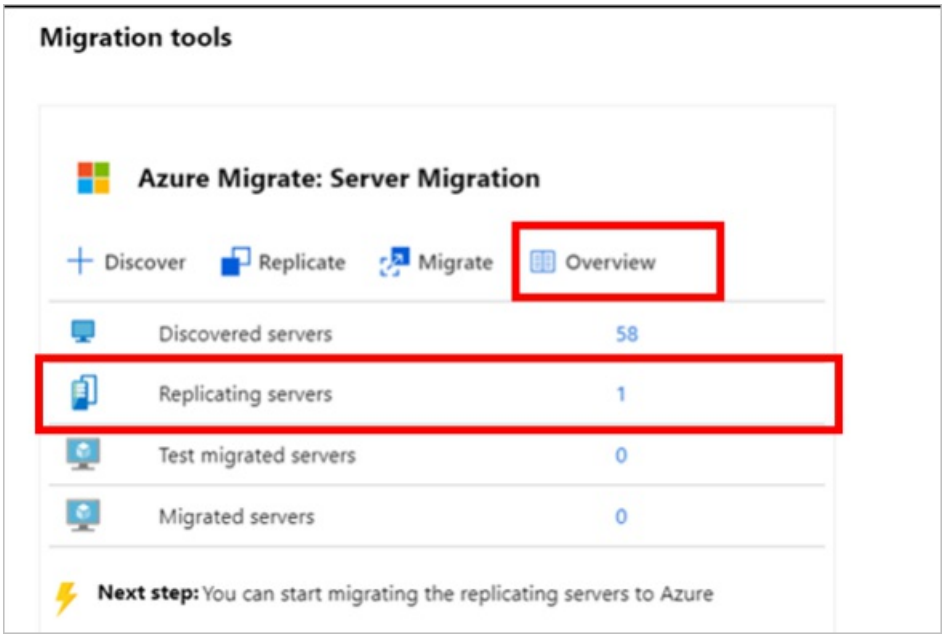
3. In **Test migration**, they select the Azure virtual network in which the Azure VM will be located after the migration. We recommend using a nonproduction virtual network.
4. The **Test migration** job starts. They monitor the job in the portal notifications.
5. After the migration finishes, they view the migrated Azure VM in **Virtual Machines** in the Azure portal. The machine name has a suffix **-Test**.
6. After the test is done, they select and hold (or right-click) the Azure VM in **Replicating machines** and then select **Clean up test migration**.



Migrate the VM

Now the Contoso admins run a full migration to complete the move.

1. In the Azure Migrate project, they go to **Servers** > **Azure Migrate: Server Migration** and select **Replicating servers**.



2. In **Replicating machines**, they select and hold (or right-click) the VM, and then they select **Migrate**.
3. In **Migrate** > **Shut down virtual machines and perform a planned migration with no data loss**,

they select **Yes** > **OK**.

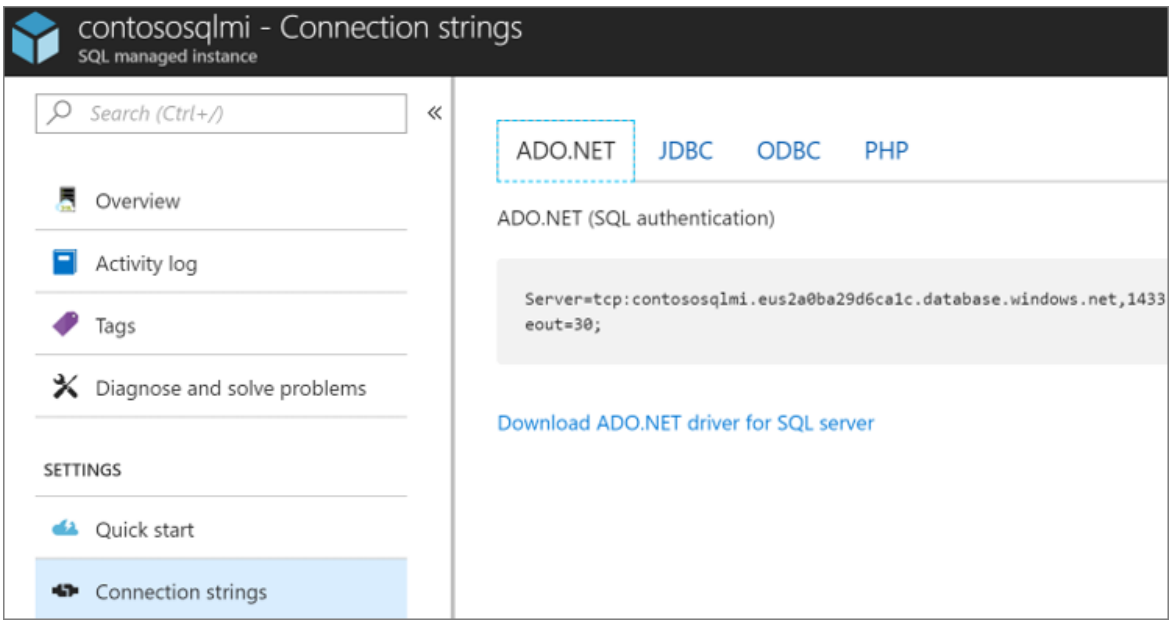
- By default, Azure Migrate shuts down the on-premises VM and runs an on-demand replication to synchronize any VM changes that occurred since the last replication occurred. This action ensures no data loss.
- If they don't want to shut down the VM, they select **No**.

4. A migration job starts for the VM. They track the job in Azure notifications.
5. After the job finishes, they can view and manage the VM from the **Virtual Machines** page.
6. Finally, they update the DNS records for `WEBVM` on one of the Contoso domain controllers.

Update the connection string

As the final step in the migration process, the Contoso admins update the connection string of the application to point to the migrated database that's running on the SQL managed instance.

1. In the Azure portal, they find the connection string by selecting **Settings** > **Connection strings**.



2. They update the string with the user name and password of the SQL managed instance.
3. After the string is configured, they replace the current connection string in the `web.config` file of its application.
4. After they update the file and save it, they restart IIS on `WEBVM` by running `iisreset /restart` in a command prompt window.
5. After IIS is restarted, the application uses the database that's running on the SQL managed instance.
6. At this point, they can shut down the on-premises `SQLVM` machine. The migration is finished.

Need more help?

- Learn how to run a test failover.
- Learn how to create a recovery plan.
- Learn how to fail over to Azure.

Clean up after migration

With the migration finished, the SmartHotel360 application is running on an Azure VM and the SmartHotel360 database is available in the Azure SQL managed instance.

Now, Contoso needs to perform these cleanup tasks:

- Remove the `WEBVM` machine from the vCenter Server inventory.
- Remove the `SQLVM` machine from the vCenter Server inventory.
- Remove `WEBVM` and `SQLVM` from local backup jobs.
- Update internal documentation to show the new location and IP address for `WEBVM`.
- Remove `SQLVM` from internal documentation. Alternatively, Contoso can revise the documentation to show `SQLVM` as deleted and no longer in the VM inventory.
- Review any resources that interact with the decommissioned VMs. Update any relevant settings or documentation to reflect the new configuration.

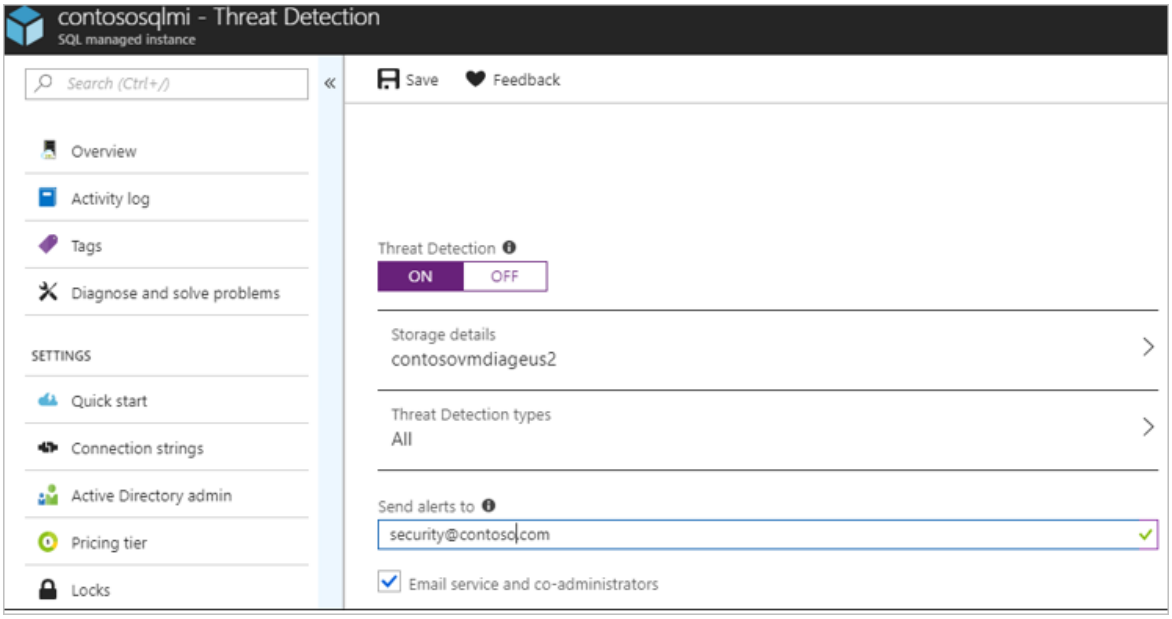
Review the deployment

With the migrated resources in Azure, Contoso needs to fully operationalize and secure its new infrastructure.

Security

The Contoso security team checks the Azure VMs and the SQL managed instance for any security issues in the implementation:

- The team reviews the network security groups that are used to control access for the VM. Network security groups help ensure that only traffic that's allowed to the application can pass.
- Contoso's security team also is considering securing the data on the disk by using Azure Disk Encryption and Azure Key Vault.
- The team enables threat detection on the managed instance. Threat detection sends an alert to Contoso's security team/service desk system to open a ticket if a threat is detected. Learn more about threat detection for SQL Managed Instance.



To learn more about security practices for VMs, see Security best practices for IaaS workloads in Azure.

Business continuity and disaster recovery

For business continuity and disaster recovery, Contoso takes the following actions:

- **Keep data safe.** Contoso backs up the data on the VMs by using the Azure Backup service. For more information, see An overview of Azure VM backup.
- **Keep applications up and running.** Contoso replicates the application VMs in Azure to a secondary region using Site Recovery. To learn more, see Set up disaster recovery to a secondary Azure region for an

Azure VM.

- **Learn more.** Contoso learns more about managing SQL Managed Instance, which includes database backups.

Licensing and cost optimization

- Contoso has existing licensing for `WEBVM`. To take advantage of pricing with Azure Hybrid Benefit, Contoso converts the existing Azure VM.
- Contoso will use Azure Cost Management + Billing to ensure the company stays within budgets established by the IT leadership.

Conclusion

In this article, Contoso rehosts the SmartHotel360 application in Azure by migrating the application front-end VM to Azure by using Azure Migrate. Contoso migrates the on-premises database to a SQL managed instance by using Azure Database Migration Service.

Rehost an on-premises application with Azure VMs and SQL Server Always On availability groups

5/18/2021 • 24 minutes to read • Edit Online

This article demonstrates how the fictional company Contoso rehosts a two-tier Windows .NET application running on VMware virtual machines (VMs) as part of a migration to Azure. Contoso migrates the application front-end VM to an Azure VM, and the application database to another Azure VM with SQL Server, running in a Windows Server failover cluster with SQL Server Always On availability groups.

The SmartHotel360 application used in this example is provided as open-source software. If you want to use it for your own testing purposes, download it from GitHub.

Business drivers

The IT leadership team has worked closely with business partners to understand what they want to achieve with this migration. They want to:

- **Address business growth.** Contoso is growing, and as a result there's pressure on on-premises systems and infrastructure.
- **Increase efficiency.** Contoso needs to remove unnecessary procedures and streamline processes for developers and users. The business needs IT to be fast and not waste time or money to deliver faster on customer requirements.
- **Increase agility.** Contoso IT needs to be more responsive to the needs of the business. It must react faster than the changes in the marketplace to enable success in a global economy. IT mustn't get in the way or become a business blocker.
- **Scale.** As the business grows successfully, Contoso IT must provide systems that grow at the same pace.

Migration goals

The Contoso cloud team has pinned down goals for this migration. These goals were used to determine the best migration method:

- After migration, the application in Azure should have the same performance capabilities as it does today in VMware. The application will remain as critical in the cloud as it is on-premises.
- Contoso doesn't want to invest in this application. It's important to the business, but in its current form, Contoso simply wants to move it safely to the cloud.
- The on-premises database for the application has had availability issues. Contoso want to deploy it in Azure as a high-availability cluster with failover capabilities.
- Contoso wants to upgrade from its current SQL Server 2008 R2 platform to SQL Server 2017.
- Contoso doesn't want to use Azure SQL Database for this application and is looking for alternatives.

Solution design

After pinning down the company's goals and requirements, Contoso designs and reviews a deployment solution and identifies the migration process. The Azure services that it will use for the migration also are identified.

Current architecture

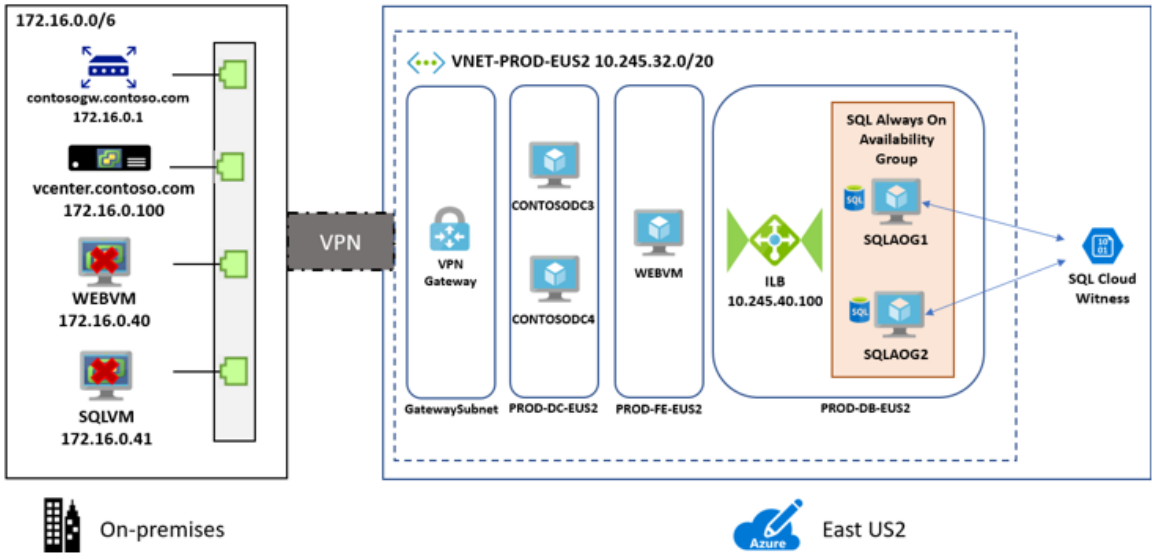
- The application is tiered across two VMs (`WEBVM` and `SQLVM`).
- The VMs are located on VMware ESXi host `contosohost1.contoso.com` (version 6.5).

- The VMware environment is managed by vCenter Server 6.5 (`vcenter.contoso.com`) that runs on a VM.
- Contoso has an on-premises datacenter (`contoso-datacenter`) with an on-premises domain controller (`contosodc1`).

Proposed architecture

In this scenario:

- Contoso will migrate the application front end `WEBVM` to an Azure infrastructure as a service (IaaS) VM.
  - The front-end VM in Azure will be deployed in the `ContosoRG` resource group (used for production resources).
  - It will be located in the Azure production network (`VNET-PROD-EUS2`) in the primary region (`East US 2`).
- The application database will be migrated to an Azure VM running SQL Server.
  - It will be located in Contoso's Azure database network (`PROD-DB-EUS2`) in the primary region (`East US 2`).
  - It will be placed in a Windows Server failover cluster with two nodes that uses SQL Server Always On availability groups.
  - In Azure, the two SQL Server VM nodes in the cluster will be deployed in the `ContosoRG` resource group.
  - The VM nodes will be located in the Azure production network (`VNET-PROD-EUS2`) in the primary region (`East US 2`).
  - VMs will run Windows Server 2016 with SQL Server 2017 Enterprise edition. Contoso doesn't have licenses for this operating system. It will use an image in Azure Marketplace that provides the license as a charge to the company's Azure Enterprise Agreement commitment.
  - Apart from unique names, both VMs use the same settings.
- Contoso will deploy an internal load balancer that listens for traffic on the cluster and directs it to the appropriate cluster node.
  - The internal load balancer will be deployed in `ContosoNetworkingRG` (used for networking resources).
- The on-premises VMs in the Contoso datacenter will be decommissioned after the migration is done.



Database considerations

As part of the solution design process, Contoso did a feature comparison between Azure SQL Database and SQL Server. The following considerations helped the company to decide to use an Azure IaaS VM running SQL Server:

- Using an Azure VM running SQL Server seems to be an optimal solution if Contoso needs to customize the operating system or the database server, or if it might want to colocate and run third-party applications on the same VM.

Solution review

Contoso evaluates its proposed design by putting together a list of pros and cons.

CONSIDERATION	DETAILS
Pros	`WEBVM` will be moved to Azure without changes, which makes the migration simple. The SQL Server tier will run on SQL Server 2017 and Windows Server 2016, which retires the current Windows Server 2008 R2 operating system. Running SQL Server 2017 supports Contoso's technical requirements and goals. IT provides 100 percent compatibility while moving away from SQL Server 2008 R2. Contoso can take advantage of its investment in Software Assurance by using the Azure Hybrid Benefit. A high-availability SQL Server deployment in Azure provides fault tolerance so that the application data tier is no longer a single point of failover.
Cons	`WEBVM` is running Windows Server 2008 R2. The operating system is supported by Azure for specific roles (July 2018). To learn more, see Microsoft server software support for Azure Virtual Machines. The web tier of the application remains a single point of failover. Contoso needs to continue supporting the web tier as an Azure VM rather than moving to a managed service such as Azure App Service. With the chosen solution, Contoso will need to continue managing two SQL Server VMs rather than moving to a managed platform, such as Azure SQL Managed Instance. In addition, with Software Assurance, Contoso could exchange its existing licenses for discounted rates on Azure SQL Managed Instance.

Azure services

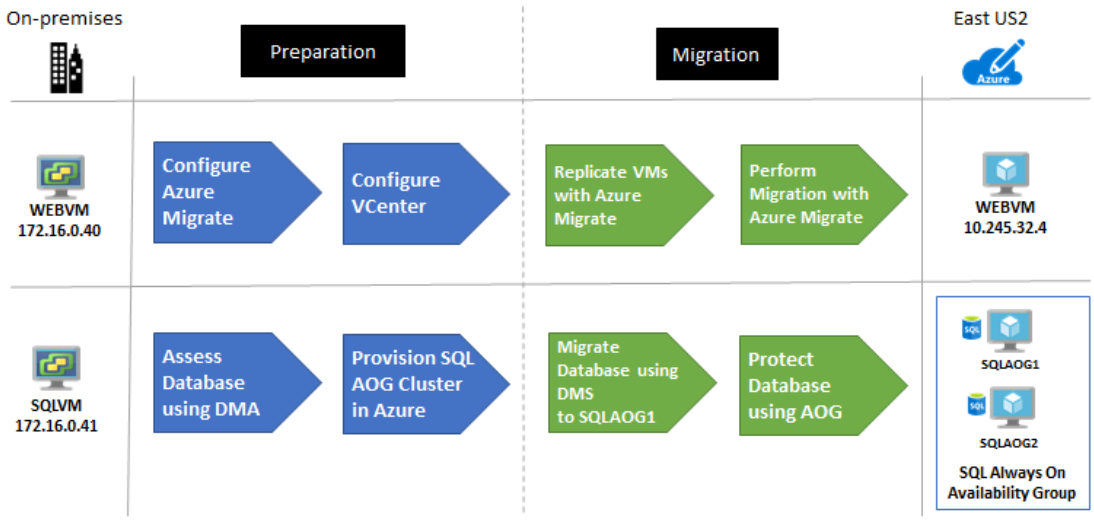
SERVICE	DESCRIPTION	COST
Azure Database Migration Service	Azure Database Migration Service enables seamless migration from multiple database sources to Azure data platforms with minimal downtime.	Learn about supported regions and Azure Database Migration Service pricing.

SERVICE	DESCRIPTION	COST
Azure Migrate	Contoso uses Azure Migrate to assess its VMware VMs. Azure Migrate assesses the migration suitability of the machines. It provides sizing and cost estimates for running in Azure.	Azure Migrate is available at no additional charge. They might incur charges depending on the tools (first-party or independent software vendor) they decide to use for assessment and migration. Learn more about Azure Migrate pricing.

Migration process

The Contoso admins will migrate the application VMs to Azure.

- They'll migrate the front-end VM to Azure VM by using Azure Migrate:
  - As a first step, they'll prepare and set up Azure components and prepare the on-premises VMware infrastructure.
  - With everything prepared, they can start replicating the VM.
  - After replication is enabled and working, they migrate the VM by using Azure Migrate.
- After they've verified the database, they'll migrate the database to a SQL Server cluster in Azure by using Azure Database Migration Service.
  - As a first step, they'll need to provision SQL Server VMs in Azure, set up the cluster and an internal load balancer, and configure Always On availability groups.
  - With this in place, they can migrate the database.
- After the migration, they'll enable Always On availability groups for the database.



Prerequisites

Here's what Contoso needs to do for this scenario.

REQUIREMENTS	DETAILS

REQUIREMENTS	DETAILS
Azure subscription	Contoso already created a subscription in an earlier article in this series. If you don't have an Azure subscription, create a free account. If you create a free account, you're the administrator of your subscription and can perform all actions. If you use an existing subscription and you're not the administrator, work with the admin to assign you Owner or Contributor permissions.
Azure infrastructure	Contoso set up the Azure infrastructure as described in Azure infrastructure for migration. Learn more about specific prerequisites requirements for Azure Migrate: Server Migration.
On-premises servers	The on-premises vCenter Server should be running version 5.5, 6.0, 6.5, or 6.7. An ESXi host running version 5.5, 6.0, 6.5, or 6.7. One or more VMware VMs running on the ESXi host.
On-premises VMs	Review Linux machines that are endorsed to run on Azure.

Scenario steps

Here's how Contoso will run the migration:

- **Step 1: Prepare a SQL Server Always On availability group cluster.** Create a cluster for deploying two SQL Server VM nodes in Azure.
- **Step 2: Deploy and set up the cluster.** Prepare a SQL Server cluster in Azure. Databases are migrated into this existing cluster.
- **Step 3: Deploy Azure Load Balancer.** Deploy a load balancer to balance traffic to the SQL Server nodes.
- **Step 4: Prepare Azure for Azure Migrate.** Create an Azure Storage account to hold replicated data.
- **Step 5: Prepare on-premises VMware for Azure Migrate.** Prepare accounts for VM discovery and agent installation. Prepare on-premises VMs so that users can connect to Azure VMs after migration.
- **Step 6: Replicate the on-premises VMs to Azure.** Enable VM replication to Azure.
- **Step 7: Migrate the database via Azure Database Migration Service.** Migrate the database to Azure by using Azure Database Migration Service.
- **Step 8: Protect the database with SQL Server Always On.** Create an Always On availability group for the cluster.
- **Step 9: Migrate the VM with Azure Migrate.** Run a test migration to make sure everything's working as expected. Then run a migration to Azure.

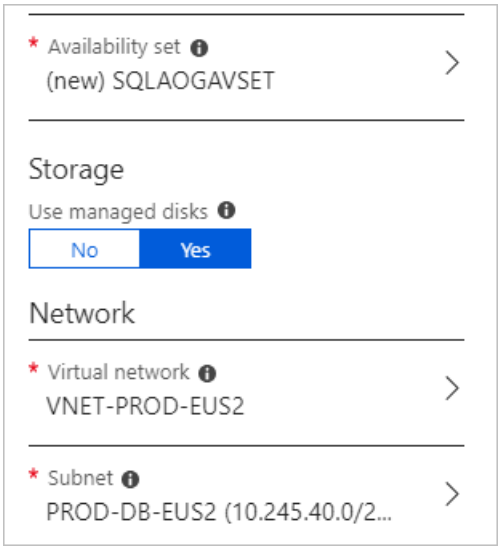
Step 1: Prepare a SQL Server Always On availability group cluster

To set up the cluster, the Contoso admins:

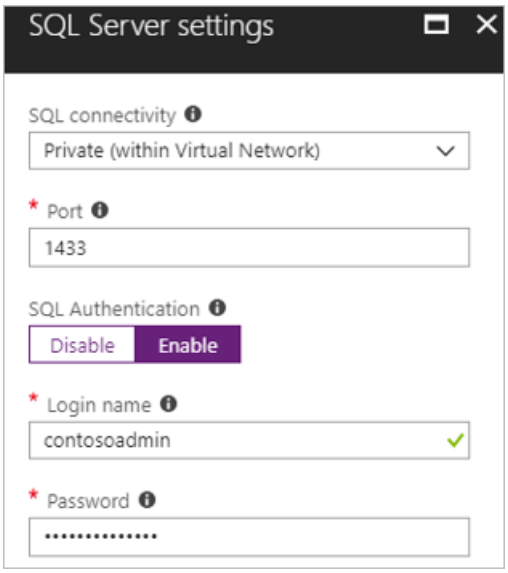
1. Create two SQL Server VMs by selecting SQL Server 2017 Enterprise Windows Server 2016 image in the Azure Marketplace.



2. In **Create Virtual Machine Wizard** > **Basics**, they configure:
   - Names for the VMs: `SQLAOG1` and `SQLAOG2`.
   - Because machines are business-critical, enable SSD for the VM disk type.
   - Specify machine credentials.
   - They deploy the VMs in the primary region (`East US 2`) in the `ContosoRG` resource group.
3. In **Size**, they start with `D2S v3` instances for both VMs. They'll scale later as needed.
4. In **Settings**, they do the following actions:
   - Because these VMs are critical databases for the application, they use managed disks.
   - They place the machines in the database subnet (`PROD-DB-EUS2`) of the production network (`VNET-PROD-EUS2`) in the primary region (`East US 2`).
   - They create a new availability set (`SQLAOGAVSET`) with two fault domains and five update domains.



5. In **SQL Server settings**, they limit SQL connectivity to the virtual network (private) on default port 1433. For authentication, they use the same credentials as used on-site (`contosoadmin`).



Need more help?

- Get help with how to provision a SQL Server VM.
- Learn about how to configure VMs for different SQL Server SKUs.

Step 2: Deploy and set up the cluster

To set up the cluster, the Contoso admins:

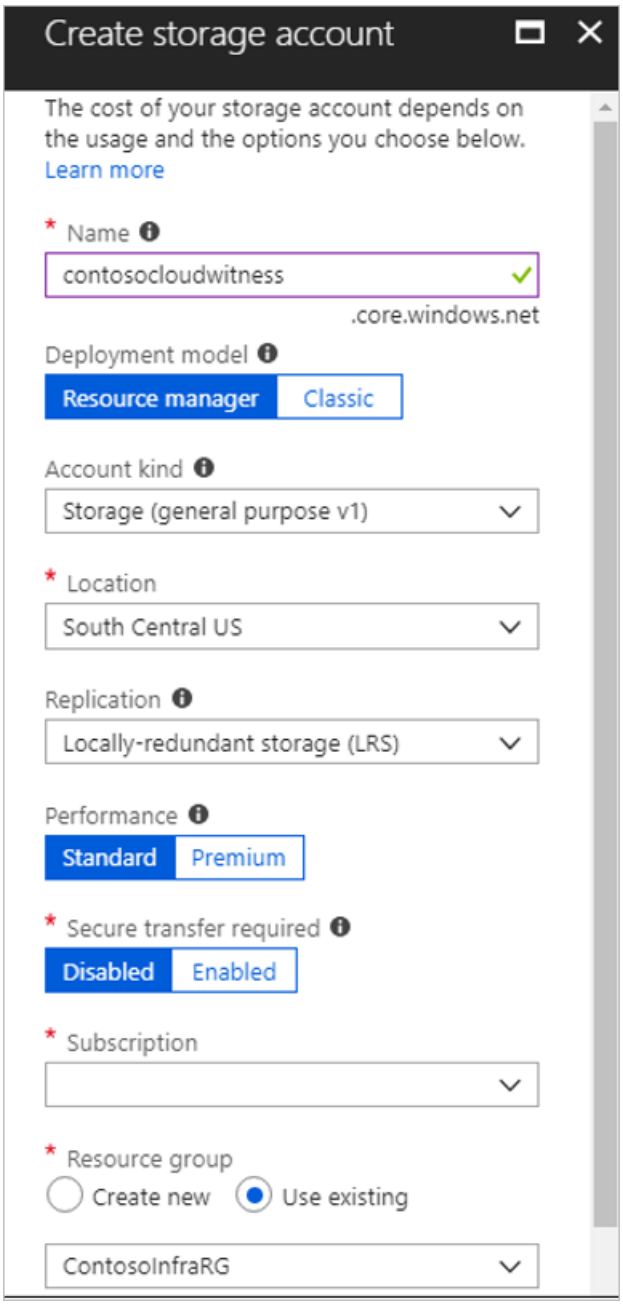
1. Set up an Azure Storage account to act as the cloud witness.
2. Add the SQL Server VMs to the Active Directory domain in the Contoso on-premises datacenter.
3. Create the cluster in Azure.
4. Configure the cloud witness.
5. Enable SQL Always On availability groups.

Set up a storage account as a cloud witness

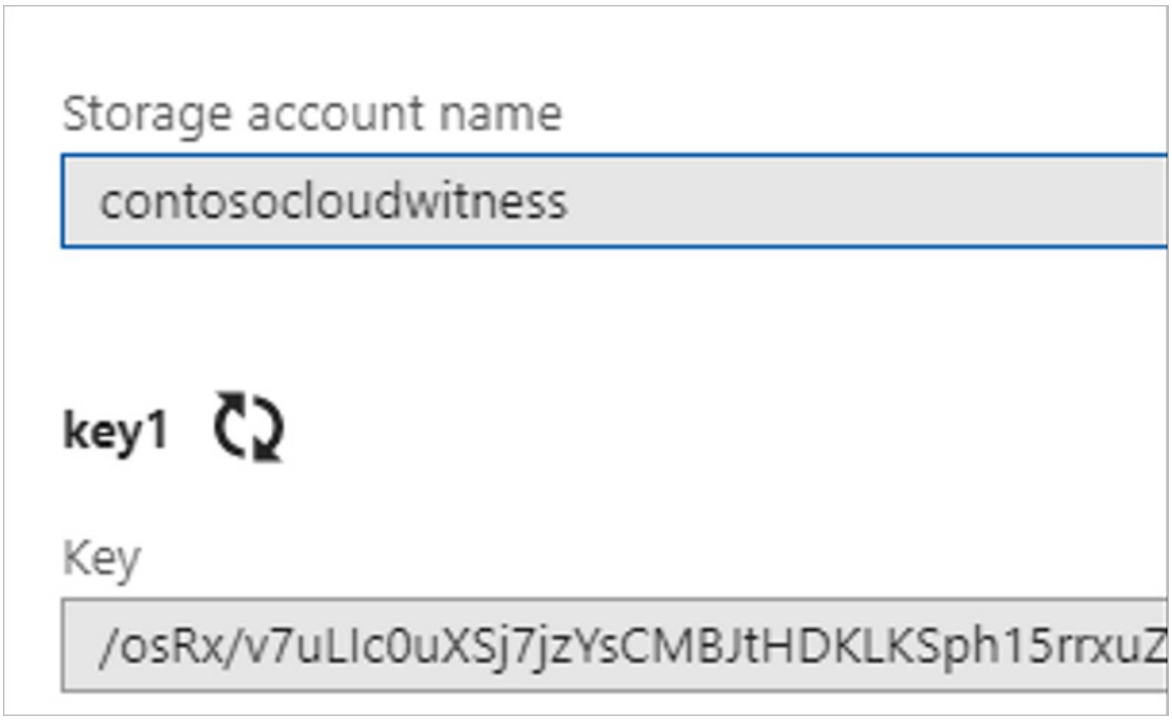
To set up a cloud witness, Contoso needs an Azure Storage account that will hold the blob file used for cluster arbitration. The same storage account can be used to set up cloud witness for multiple clusters.

To create a storage account, the Contoso admins:

1. Specify a recognizable name for the account (`contosocloudwitness`).
2. Deploy a general all-purpose account, with LRS.
3. Place the account in a third region (`South Central US`). They place it outside the primary and secondary region so that it remains available during regional failure.
4. Place it in the resource group that holds infrastructure resources, `ContosoInfraRG`.



5. When they create the storage account, primary and secondary access keys are generated for it. They need the primary access key to create the cloud witness. The key appears under the storage account name > **Access keys**.

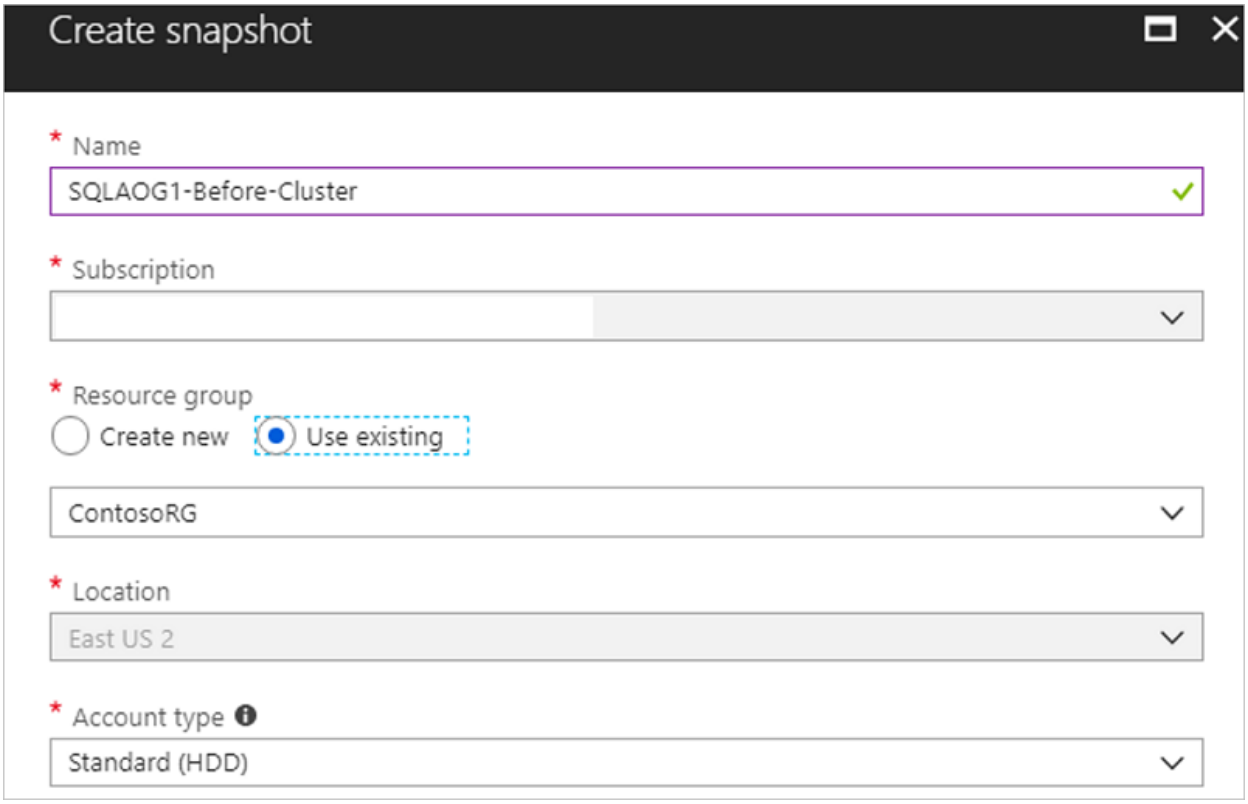


Add SQL Server VMs to Contoso domain

1. Contoso adds `SQLAOG1` and `SQLAOG2` to the `contoso.com` domain.
2. On each VM, the admins install the Windows Failover Cluster feature and tools.

Set up the cluster

Before the Contoso admins set up the cluster, they take a snapshot of the OS disk on each machine.



1. They run a script to create the Windows failover cluster.

ClusterCreateSQLAOG.ps1

```
<#
Microsoft Cloud Workshop: BCDR
.File Name
 - ClusterCreateSQLAOG.ps1

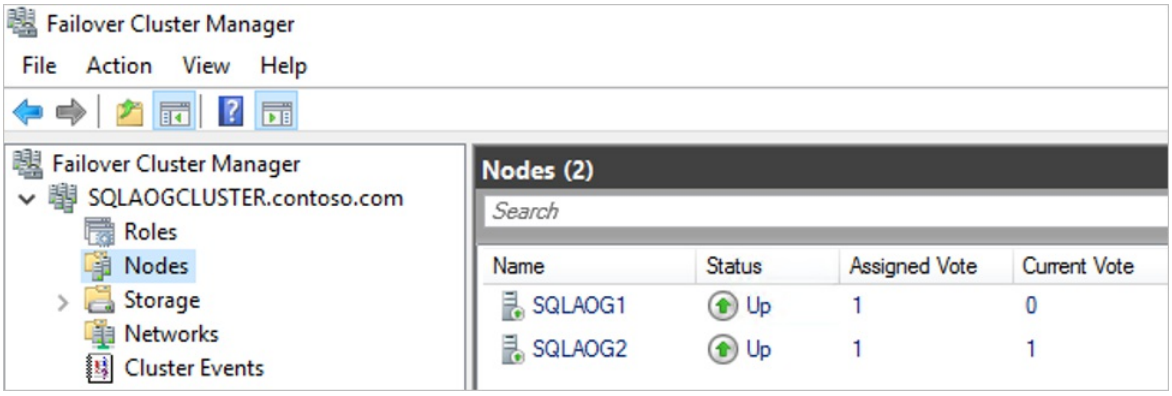
.What calls this script?
 - This is run manually on SQLAOG1 to create the Windows Failover Cluster

.What does this script do?
 - The Result will be a Cluster Named AOGCLUSTER running on the 10.245.40.99 IP Address
   with the hostname of SQLAOGCLUSTER.CONTOSO.COM

 - Two Cluster Networks will be created Cluster Network 1 (10.245.40.0/24)
#>
New-Cluster -Name SQLAOGCLUSTER -Node SQLAOG1,SQLAOG2 -StaticAddress 10.245.40.99

```

2. After the cluster is created, they verify that the VMs appear as cluster nodes.

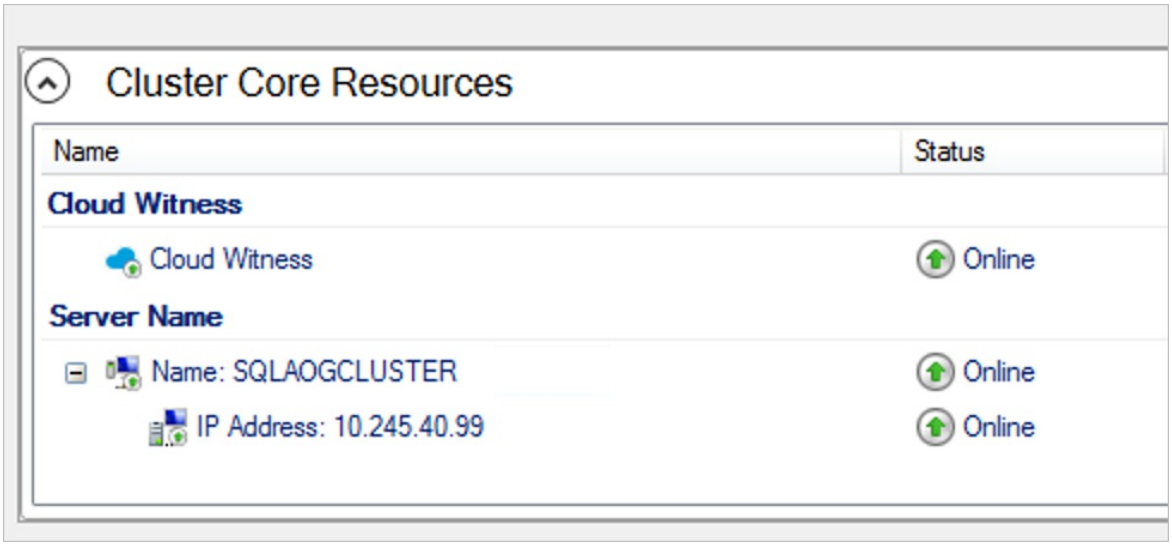


Configure the cloud witness

1. The Contoso admins configure the cloud witness by using the **Quorum Configuration Wizard** in Failover Cluster Manager.

2. In the wizard, they select to create a cloud witness with the storage account.

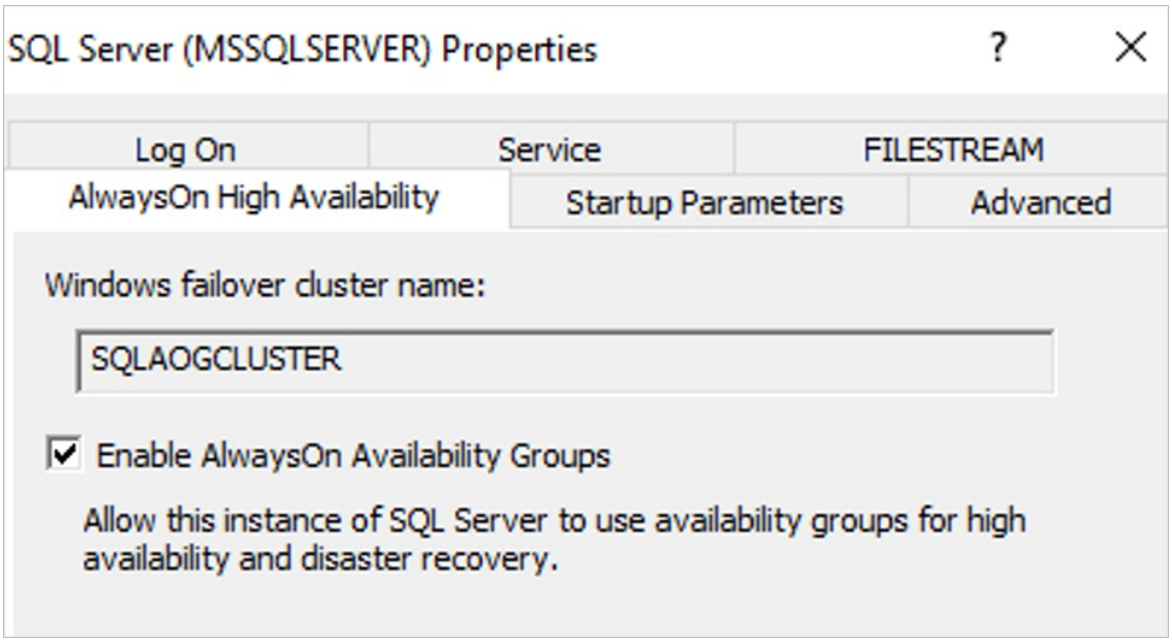
3. After the cloud witness is configured, it appears in the Failover Cluster Manager snap-in.



Enable SQL Server Always On availability groups

The Contoso admins can now enable Always On availability groups:

1. In SQL Server Configuration Manager, they enable **Always On availability groups** for the **SQL Server (MSSQLSERVER)** service.



2. They restart the service for changes to take effect.

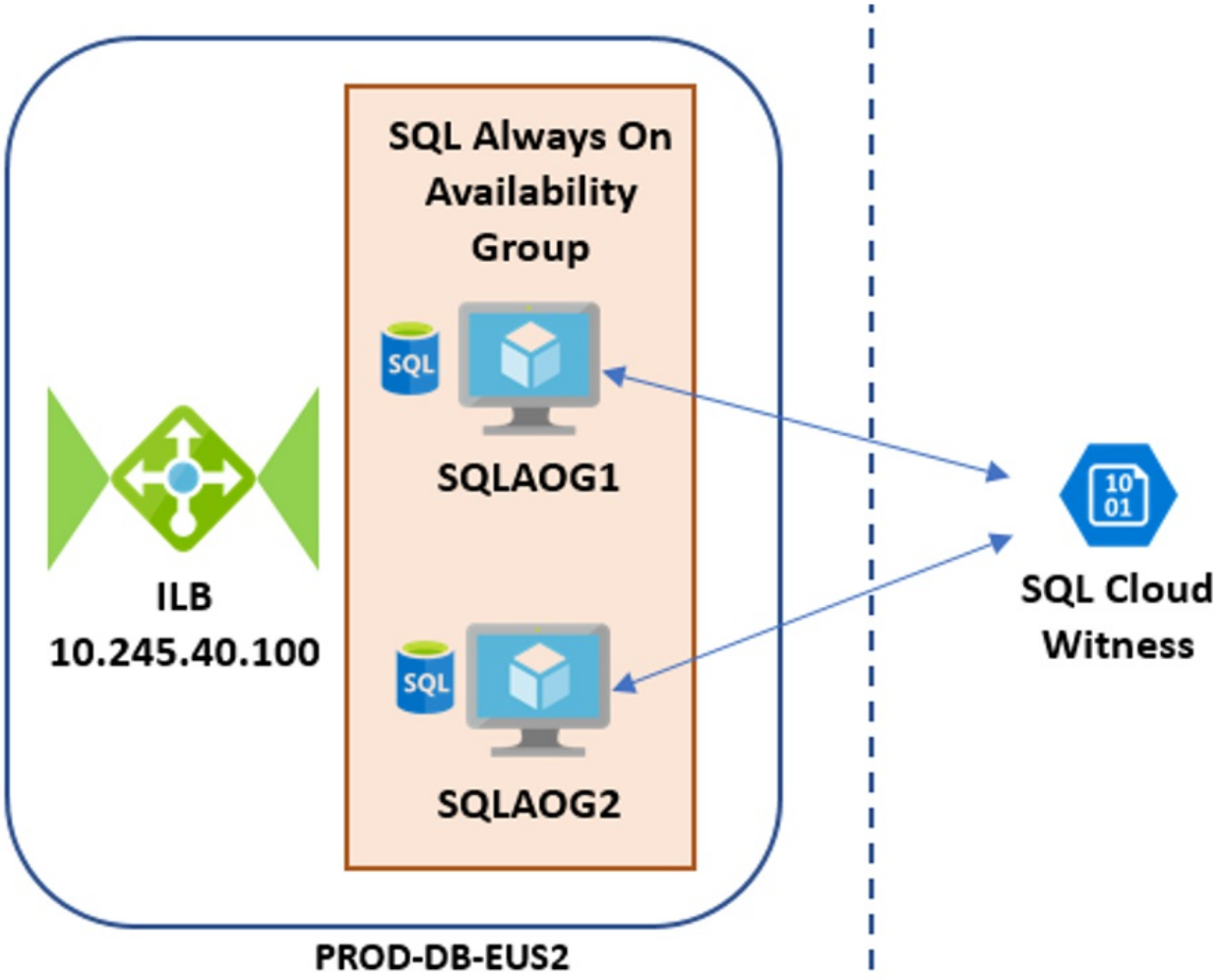
With Always On availability groups enabled, Contoso can set up the Always On availability group that will protect the `SmartHotel360` database.

Need more help?

- Read about cloud witness and setting up a storage account for it.
- Get instructions for how to set up a cluster and create an availability group.

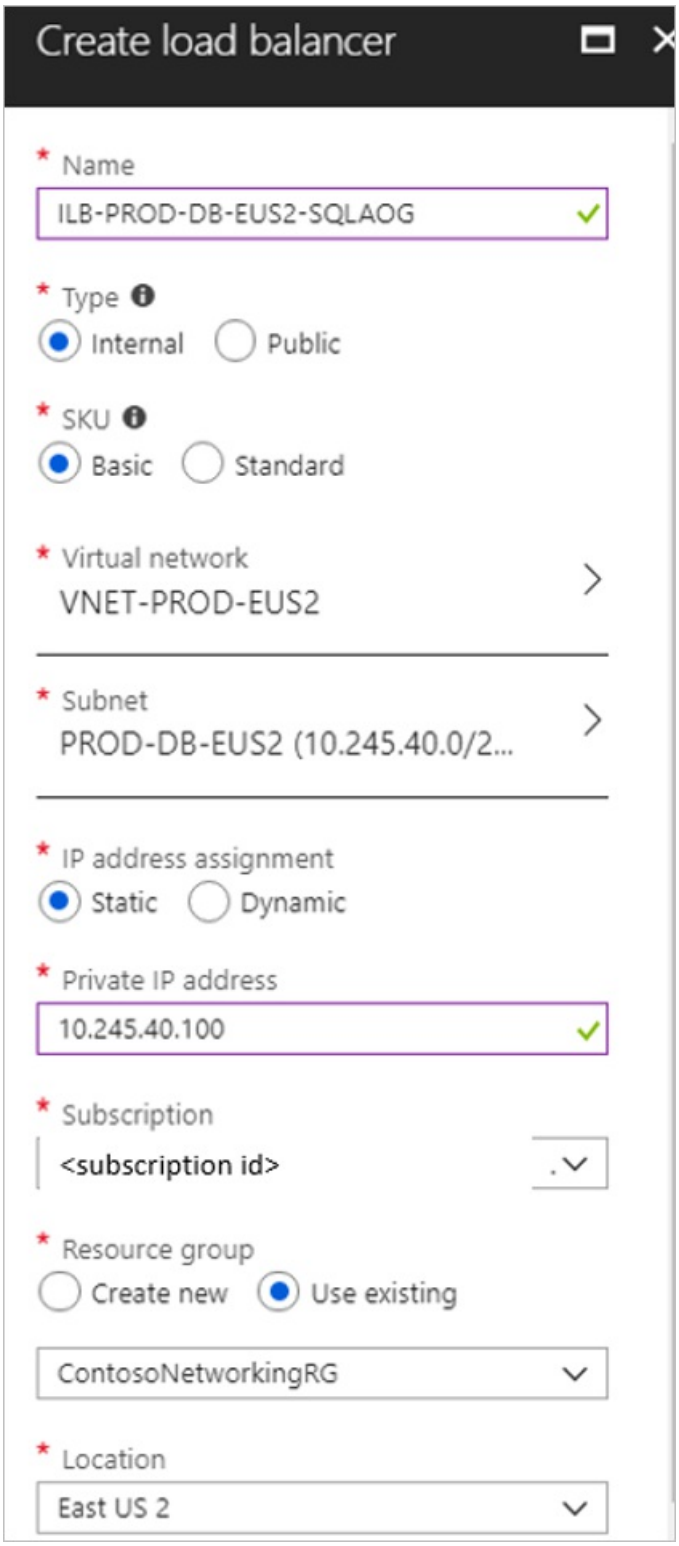
Step 3: Deploy Azure Load Balancer

The Contoso admins now want to deploy an internal load balancer that sits in front of the cluster nodes. The load balancer listens for traffic and directs it to the appropriate node.



To create the load balancer, the Contoso admins:

1. In the Azure portal, go to **Networking** > **Load balancer**, and set up a new internal load balancer: `ILB-PROD-DB-EUS2-SQLAOG`.
2. Place the load balancer in the database subnet (`PROD-DB-EUS2`) of the production network (`VNET-PROD-EUS2`).
3. Assign it a static IP address (`10.245.40.100`).
4. As a networking element, deploy the load balancer in the networking resource group `ContosoNetworkingRG`.

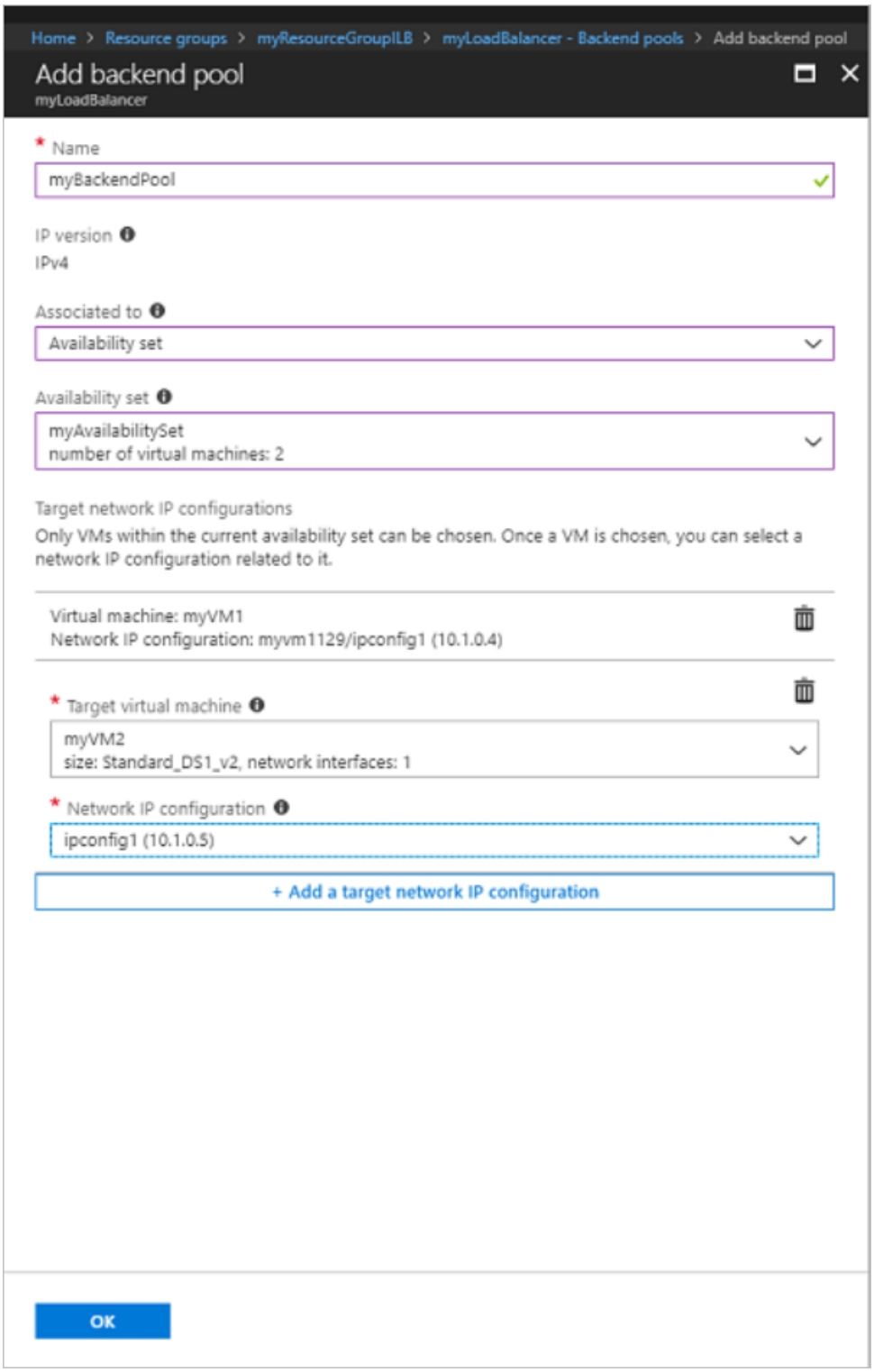


After the internal load balancer is deployed, the Contoso admins need to set it up. They create a back-end address pool, set up a health probe, and configure a load-balancing rule.

Add a back-end pool

To distribute traffic to the VMs in the cluster, the Contoso admins set up a back-end address pool that contains the IP addresses of the NICs for VMs that will receive network traffic from the load balancer.

1. In the load balancer settings in the portal, Contoso adds a back-end pool: `ILB-PROD-DB-EUS-SQLAOG-BEPOOL`.
2. The admins associate the pool with availability set `SQLAOGAVSET`. The VMs in the set (`SQLAOG1` and `SQLAOG2`) are added to the pool.

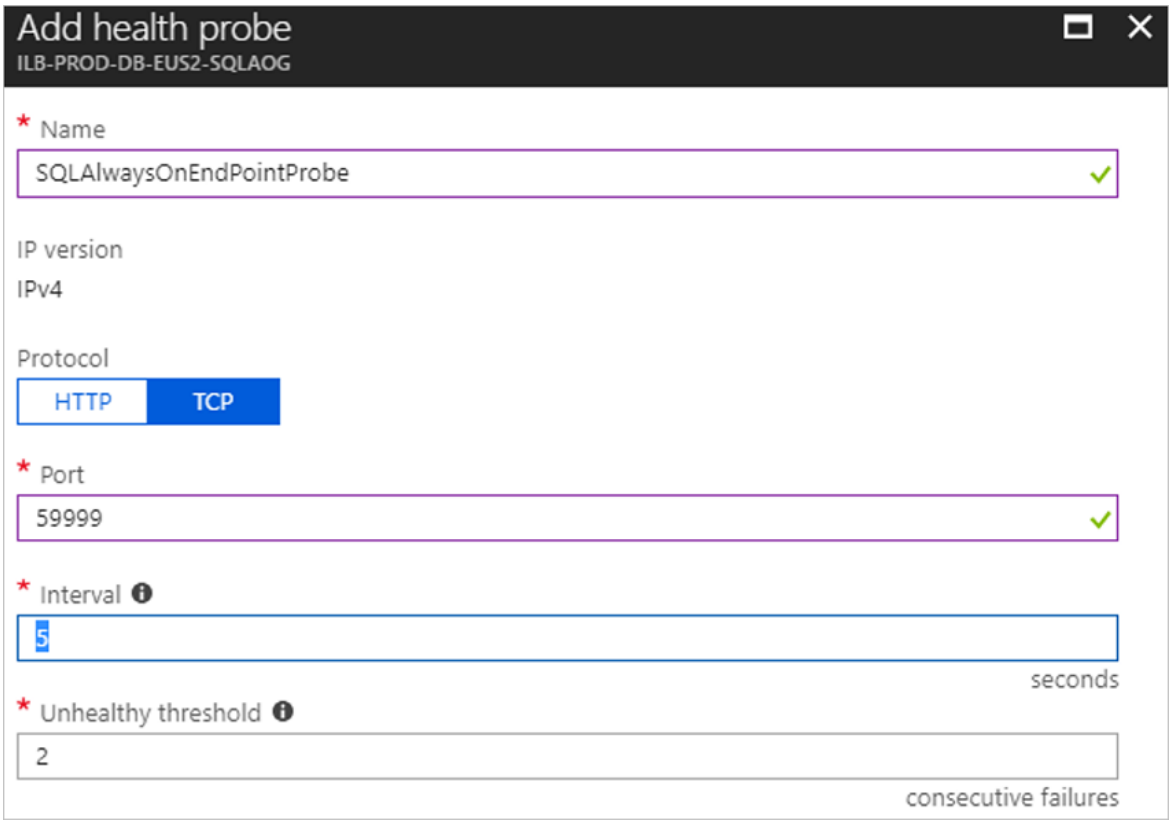


Create a health probe

The Contoso admins create a health probe so that the load balancer can monitor the application health. The probe dynamically adds or removes VMs from the load balancer rotation based on how they respond to health checks.

To create the probe, the Contoso admins:

1. In the load balancer settings in the portal, create a health probe: `SQLAlwaysOnEndPointProbe`.
2. Set the probe to monitor VMs on TCP port 59999.
3. Set an interval of 5 seconds between probes and a threshold of 2. If two probes fail, the VM will be considered unhealthy.



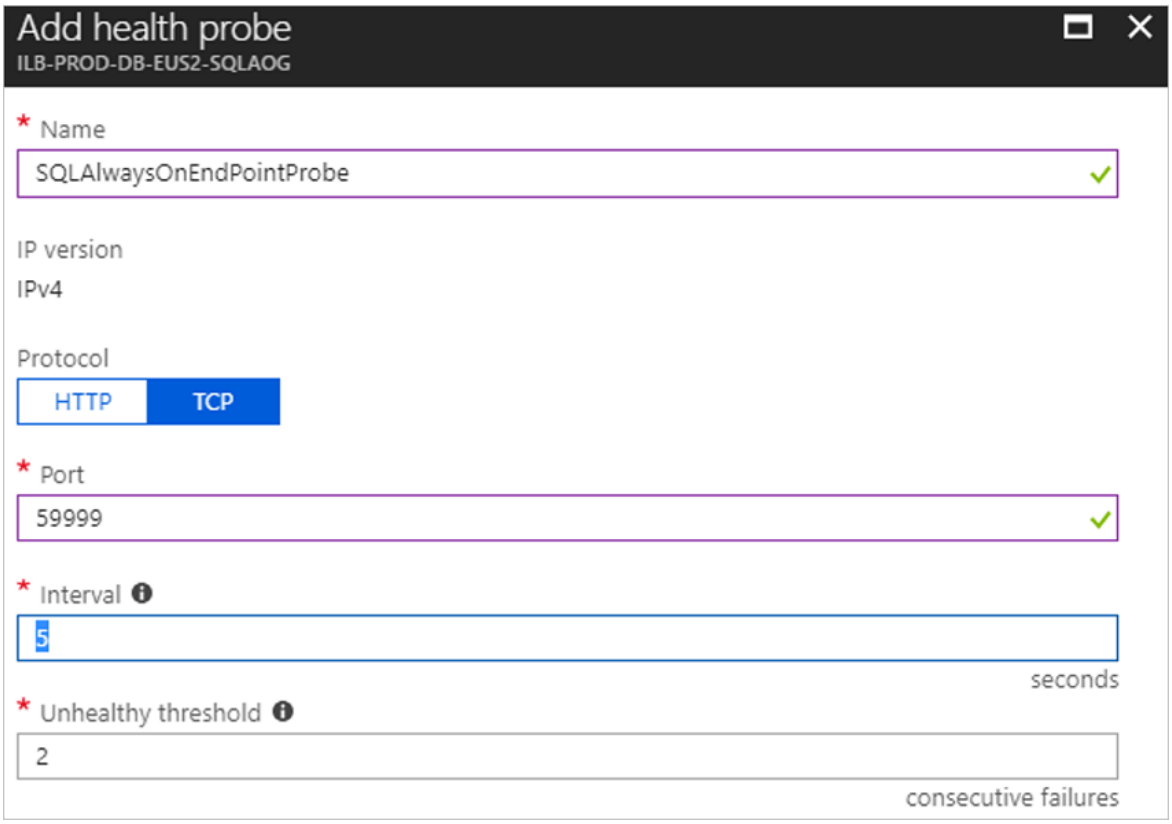
Configure the load balancer to receive traffic

Now, the Contoso admins set up a load balancer rule to define how traffic is distributed to the VMs.

- The front-end IP address handles incoming traffic.
- The back-end IP pool receives the traffic.

To create the rule, the Contoso admins:

1. In the load balancer settings in the portal, add a new rule: `SQLAlwaysOnEndPointListener`.
2. Set a front-end listener to receive incoming SQL client traffic on TCP port 1433.
3. Specify the back-end pool to which traffic will be routed and the port on which VMs listen for traffic.
4. Enable floating IP (direct server return), which is always required for SQL Server Always On.



Need more help?

- Get an overview of Azure Load Balancer.
- Learn about how to create a load balancer.

Step 4: Prepare Azure for Azure Migrate

Here are the Azure components Contoso needs to deploy Azure Migrate:

- A virtual network in which VMs will be located when they're migrated.
- An Azure Storage account to hold replicated data.

The Contoso admins set up these components:

1. Contoso already created a network/subnet it can use for Azure Migrate when it deployed the Azure infrastructure.
    - The SmartHotel360 application is a production application, and `WEBVM` will be migrated to the Azure production network (`VNET-PROD-EUS2`) in the primary region (`East US 2`).
    - `WEBVM` will be placed in the `ContosoRG` resource group, which is used for production resources, and in the production subnet (`PROD-FE-EUS2`).
2. The Contoso admins create an Azure Storage account (`contosovmsacc20180528`) in the primary region.
    - Use a general-purpose account with standard storage and LRS replication.

Step 5: Prepare on-premises VMware for Azure Migrate

Here's what the Contoso admins prepare on-premises:

- An account on the vCenter Server or vSphere ESXi host to automate VM discovery.
- On-premises VM settings so that Contoso can connect to the replicated Azure VM after migration.

Prepare an account for automatic discovery

Azure Migrate needs access to VMware servers to:

- Automatically discover VMs.
- Orchestrate replication and migration.
- At least a read-only account is required. They need an account that can run operations such as creating and removing disks and turning on VMs.

To set up the account, the Contoso admins:

1. Create a role at the vCenter level.
2. Assign that role the required permissions.

Prepare to connect to Azure VMs after migration

After migration, Contoso wants to connect to the Azure VMs and allow Azure to manage the VMs. To do this, the Contoso admins do the following tasks before migration:

1. For access over the internet, they:
   - Enable RDP or SSH on the on-premises VM before migration.
   - Ensure that TCP and UDP rules are added for the **Public** profile.
   - Check that RDP or SSH is allowed in the operating system firewall.
2. For access over Site-to-Site VPN, they:
   - Enable RDP or SSH on the on-premises VM before migration.
   - Check that RDP or SSH is allowed in the operating system firewall.
   - For Windows, set the operating system's SAN policy on the on-premises VM to **OnlineAll**.
3. Install the Azure agent:
   - Azure Linux agent
   - Azure Windows agent
4. Miscellaneous
   - For Windows, there should be no Windows updates pending on the VM when triggering a migration. If there are, the Contoso admins won't be able to sign in to the VM until the update completes.
   - After migration, they can check **Boot diagnostics** to view a screenshot of the VM. If it doesn't work, they should verify that the VM is running and review these troubleshooting tips.

Need more help?

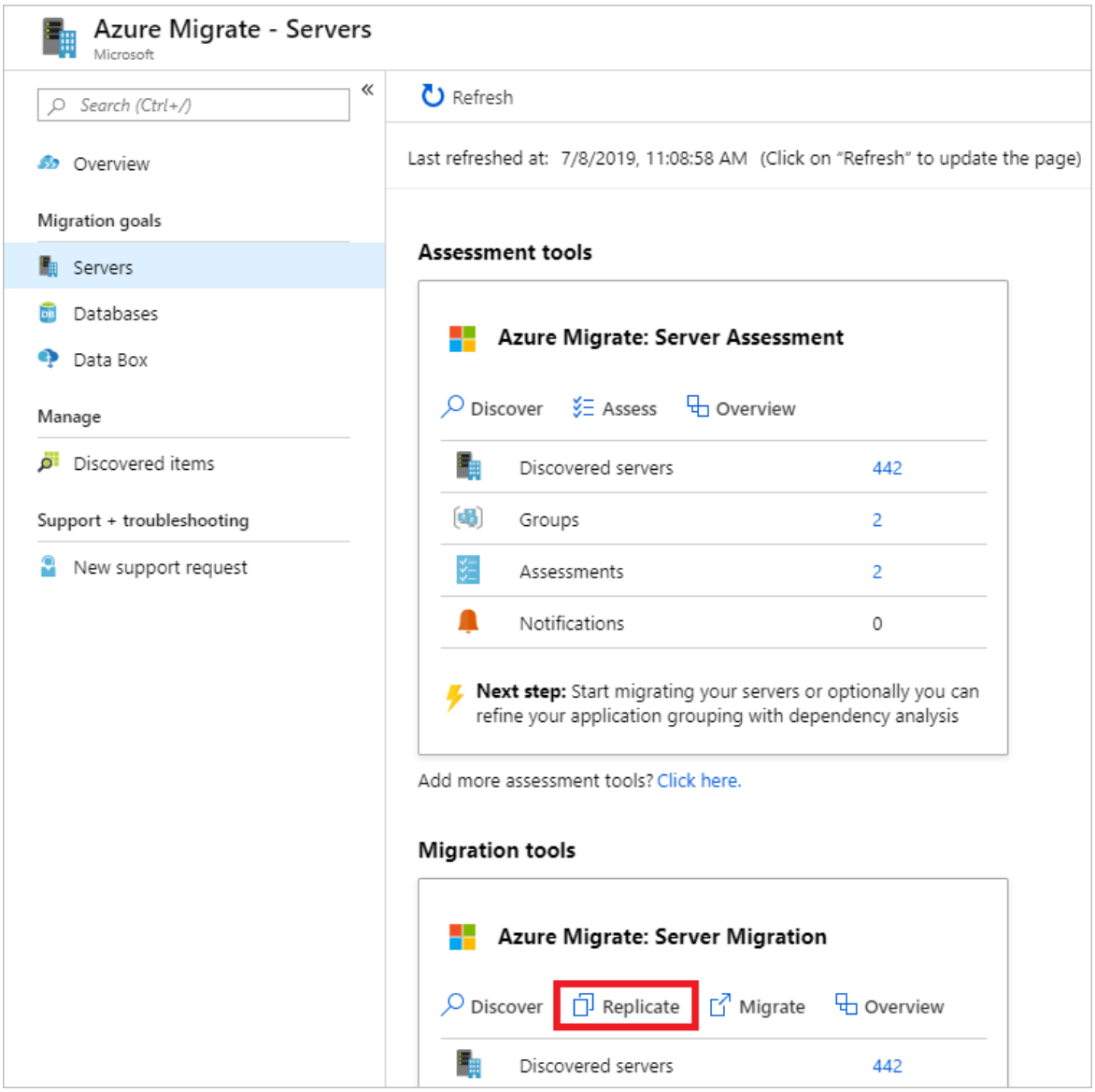
Learn about how to prepare VMs for migration.

Step 6: Replicate the on-premises VMs to Azure

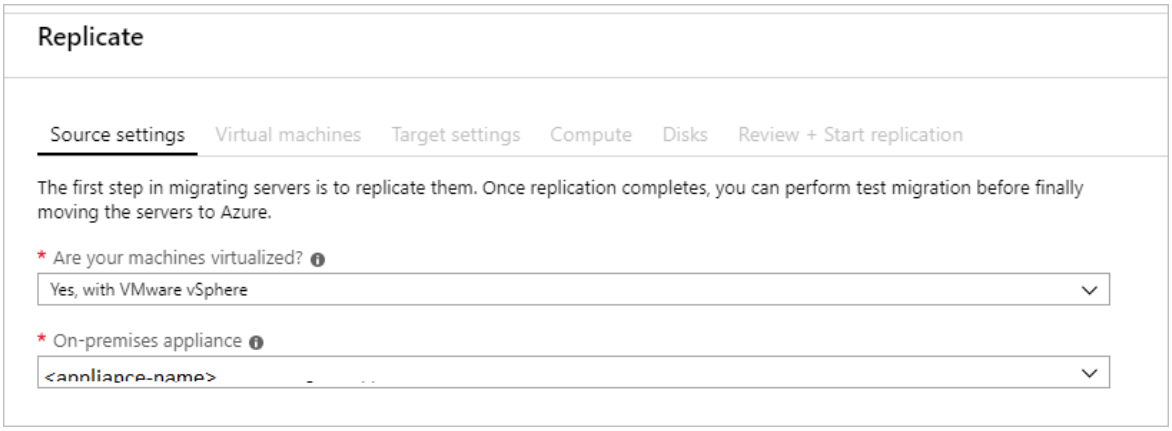
Before the Contoso admins can run a migration to Azure, they need to set up and enable replication.

With discovery finished, they can begin replication of VMware VMs to Azure.

1. In the Azure Migrate project, they go to **Servers** > **Azure Migrate: Server Migration**, and select **Replicate**.



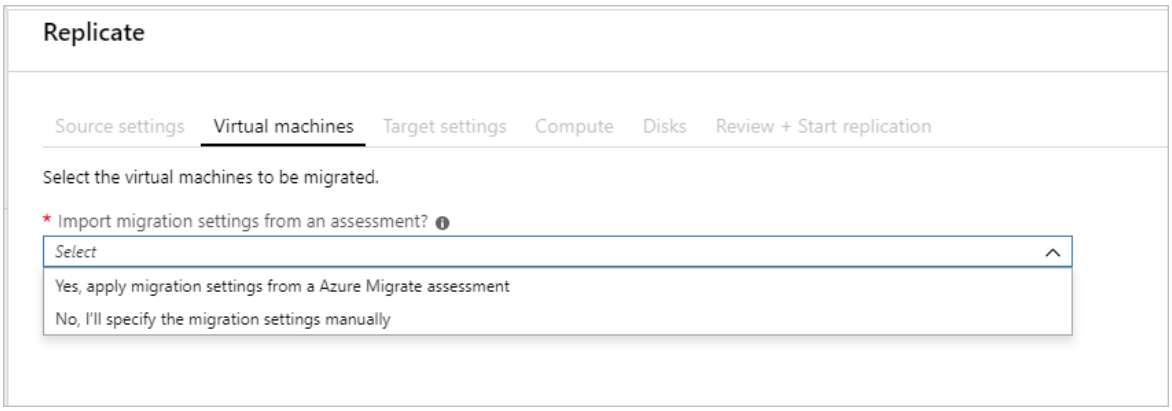
2. In **Replicate** > **Source settings** > **Are your machines virtualized?**, they select **Yes, with VMware vSphere**.
3. In **On-premises appliance**, they select the name of the Azure Migrate appliance that was set up, and then select **OK**.



4. In **Virtual machines**, they select the machines to replicate.
   - If the Contoso admins have run an assessment for the VMs, they can apply VM sizing and disk type (premium/standard) recommendations from the assessment results. In **Import migration settings from an Azure Migrate assessment?**, they select the **Yes** option.
   - If they didn't run an assessment or don't want to use the assessment settings, they select the **No**

option.

- If they selected to use the assessment, they select the VM group and assessment name.



5. In **Virtual machines**, they search for VMs as needed and check each VM to migrate. Then they select **Next: Target settings**.
6. In **Target settings**, they select the subscription, and target region to which they'll migrate, and specify the resource group in which the Azure VMs will reside after migration. In **Virtual Network**, they select the Azure virtual network/subnet to which the Azure VMs will be joined after migration.
7. In **Azure Hybrid Benefit**, the Contoso admins:
   - Select **No** if they don't want to apply Azure Hybrid Benefit. Then they select **Next**.
   - Select **Yes** if they have Windows Server machines that are covered with active Software Assurance or Windows Server subscriptions, and they want to apply the benefit to the machines they're migrating. Then they select **Next**.
8. In **Compute**, they review the VM name, size, OS disk type, and availability set. VMs must conform with Azure requirements.
   - **VM size:** If they're using assessment recommendations, the VM size drop-down list contains the recommended size. Otherwise, Azure Migrate picks a size based on the closest match in the Azure subscription. Alternatively, they can pick a manual size in **Azure VM size.**
   - **OS disk:** They specify the OS (boot) disk for the VM. The OS disk is the disk that has the operating system bootloader and installer.
   - **Availability set:** If the VM should be in an Azure availability set after migration, they specify the set. The set must be in the target resource group specified for the migration.
9. In **Disks**, they specify whether the VM disks should be replicated to Azure. Then they select the disk type (standard SSD/HDD or premium managed disks) in Azure and select **Next**.
   - They can exclude disks from replication.
   - If disks are excluded, they won't be present on the Azure VM after migration.
10. In **Review + Start replication**, they review the settings. Then they select **Replicate** to start the initial replication for the servers.

> **NOTE**
> Replication settings can be updated any time before replication starts in **Manage** > **Replicating machines**. Settings can't be changed after replication starts.

Step 7: Migrate the database via Azure Database Migration Service

The Contoso admins migrate the database via Azure Database Migration Service by following the step-by-step migration tutorial. They can perform online, offline, and hybrid (preview) migrations.

As a summary, they must perform the following tasks:

- Use the Premium pricing tier to create an Azure Database Migration Service instance that connects to the virtual network.
- Ensure that the instance can access the remote SQL Server via the virtual network. Ensure that all incoming ports are allowed from Azure to SQL Server at the virtual network level, the network VPN, and the machine that hosts SQL Server.
- Configure the instance:
  - Create a migration project.
  - Add a source (on-premises database).
  - Select a target.
  - Select the databases to migrate.
  - Configure advanced settings.
  - Start the replication.
  - Resolve any errors.
  - Perform the final cutover.

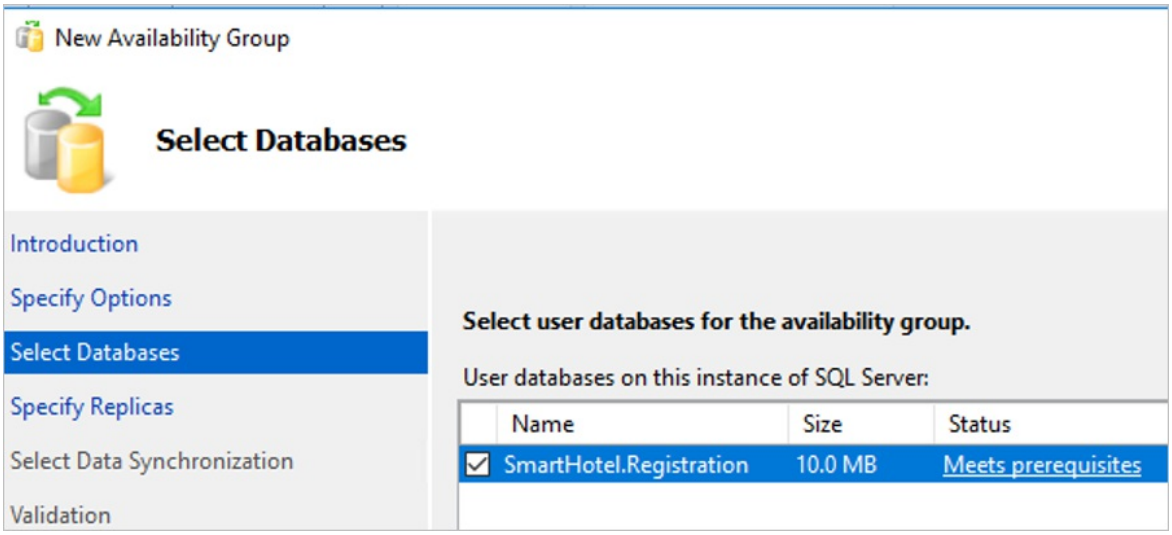
Step 8: Protect the database with SQL Server Always On

With the application database running on `SQLAOG1`, the Contoso admins can now protect it by using Always On availability groups. They configure SQL Server Always On by using SQL Server Management Studio and then assign a listener by using Windows clustering.

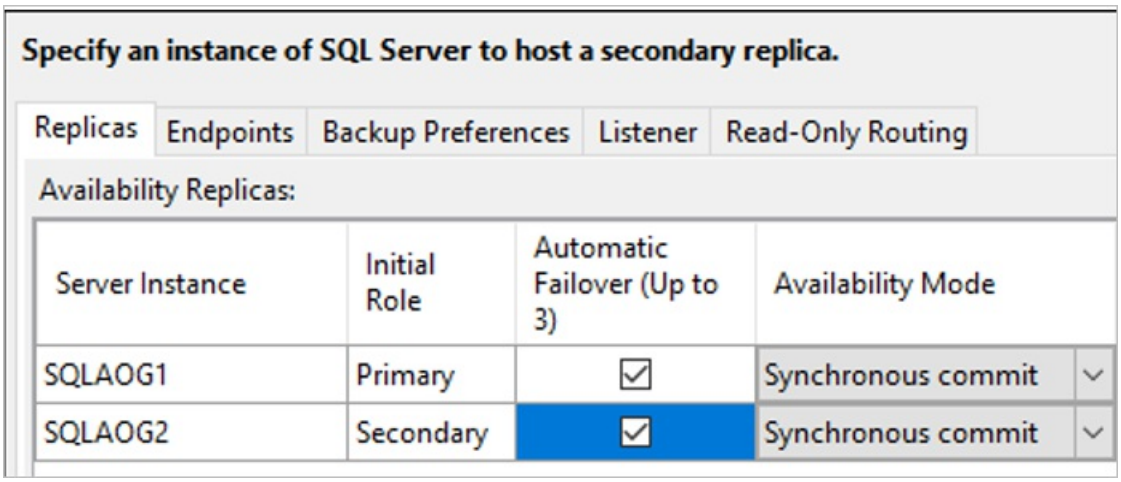
Create an Always On availability group

1. In SQL Server Management Studio, they select and hold (or right-click) **Always On High Availability** to start the **New Availability Group Wizard**.

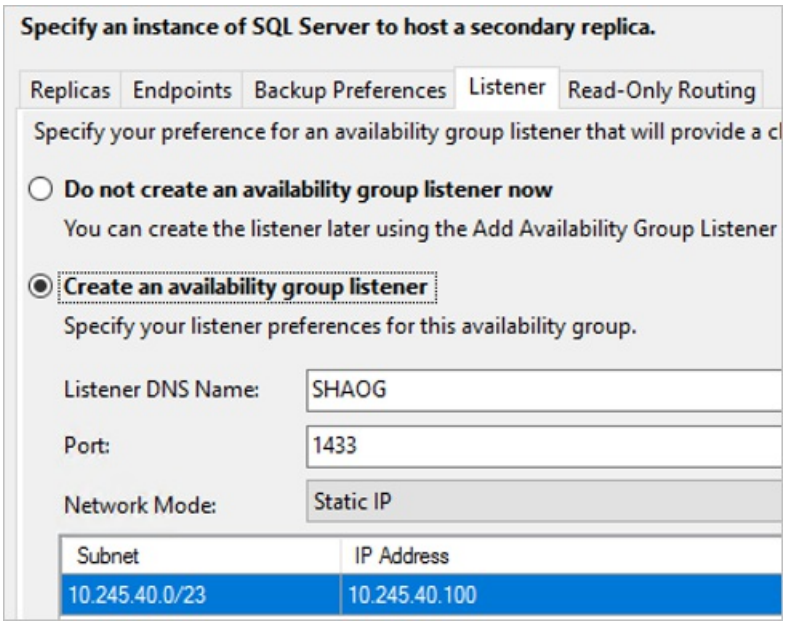
2. In **Specify Options**, they name the availability group `SHAOG`. In **Select Databases**, they select the `SmartHotel360` database.



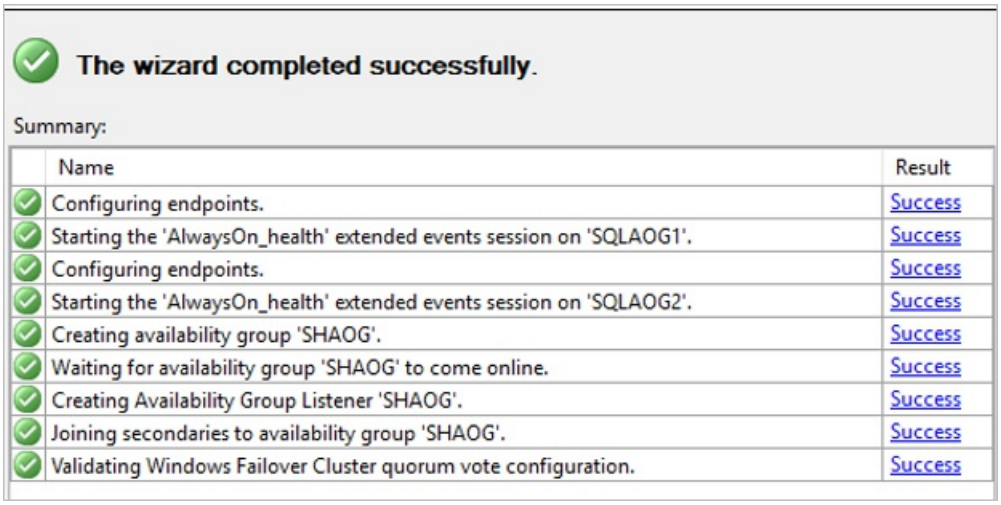
3. In **Specify Replicas**, they add the two SQL nodes as availability replicas and configure them to provide automatic failover with synchronous commit.



4. They configure a listener for the group (`SHAOG`) and port. The IP address of the internal load balancer is added as a static IP address (`10.245.40.100`).



5. In **Select Data Synchronization**, they enable automatic seeding. With this option, SQL Server automatically creates secondary replicas for every database in the group, so Contoso doesn't have to manually back up and restore them. After validation, the availability group is created.



6. Contoso ran into an issue when creating the group. It isn't using Active Directory Windows integrated security and needs to grant permissions to the SQL login to create the Windows failover cluster roles.

```
SQLQuery1.sql - S...contosoadmin (55))*   SHAOG:SQLAOG1
GRANT ALTER ANY AVAILABILITY GROUP TO [NT AUTHORITY\SYSTEM];
GRANT CONNECT SQL TO [NT AUTHORITY\SYSTEM];
GRANT VIEW SERVER STATE TO [NT AUTHORITY\SYSTEM];
```

7. After the group is created, it appears in SQL Server Management Studio.

Configure a listener on the cluster

As a last step in setting up the SQL deployment, the Contoso admins configure the internal load balancer as the listener on the cluster and bring the listener online. They use a script to do this task.

```
ClusterUpdateSQLAOG.ps1
<#
Contoso Cloud Migration
.File Name
 - ClusterUpdateSQLAOG.ps1

.What calls this script?
 - Run this script on SQLAOG1 to update Windows Failover Cluster, but only after
   ClusterCreateSQLAOG.ps1 has be successfully executed and the steps to establish
   the SQL Always On Availability Group has been completed.

.What does this script do?
 - This script creates the Windows Failover Client IP probes that are required to determine
   which nodes are online.
#>

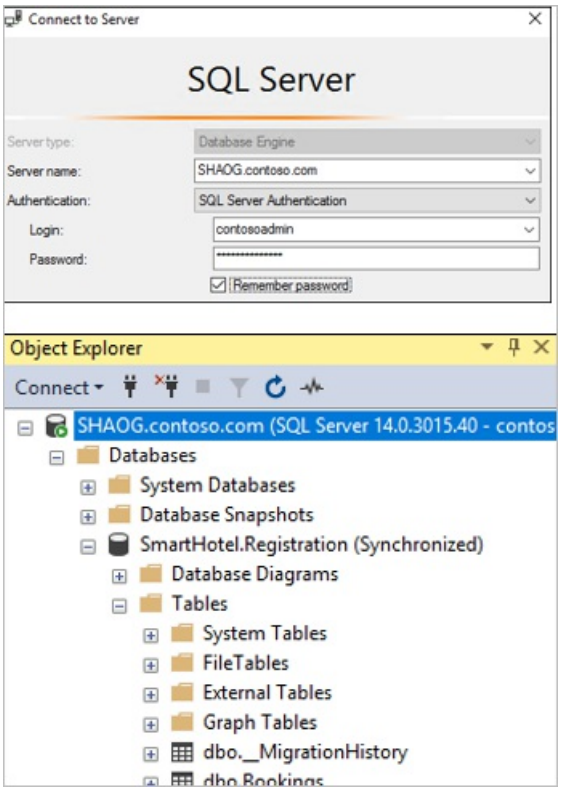
$ClusterNetworkName = "Cluster Network 1"
$IPResourceName = "SHAOG_10.245.40.100"
$ILBIP = "10.245.40.100"
Import-Module FailoverClusters
Get-ClusterResource $IPResourceName | Set-ClusterParameter -Multiple @{"Address"="$ILBIP";"Pr
Stop-ClusterResource -Name $IPResourceName
Start-ClusterResource -Name "SHAOG"
```

```
PS C:\Users\administrator\Desktop> C:\Users\administrator\Desktop\ClusterUpdateSQLAOG.ps1
WARNING: The properties were stored, but not all changes will take effect until SHAOG_10.245.40.100

Name                State   OwnerGroup ResourceType
----                -----   ---------- ------------
SHAOG_10.245.40.100 Offline SHAOG      IP Address
SHAOG               Online  SHAOG      SQL Server Availability Group
```

Verify the configuration

With everything set up, Contoso now has a functional availability group in Azure that uses the migrated database. The admins verify the configuration by connecting to the internal load balancer in SQL Server Management Studio.



Need more help?

- Learn about how to create an availability group and listener.
- Manually set up the cluster to use the load balancer IP address.
- Learn more about how to create and use SAS.

Step 9: Migrate the VM with Azure Migrate

The Contoso admins run a quick test failover and then migrate the VM.

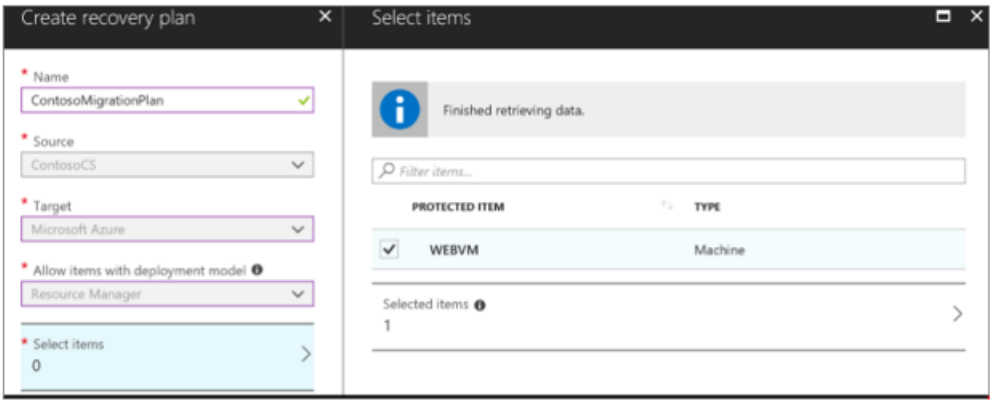
Run a test migration

Running a test migration helps ensure that everything's working as expected before the migration. The Contoso admins:

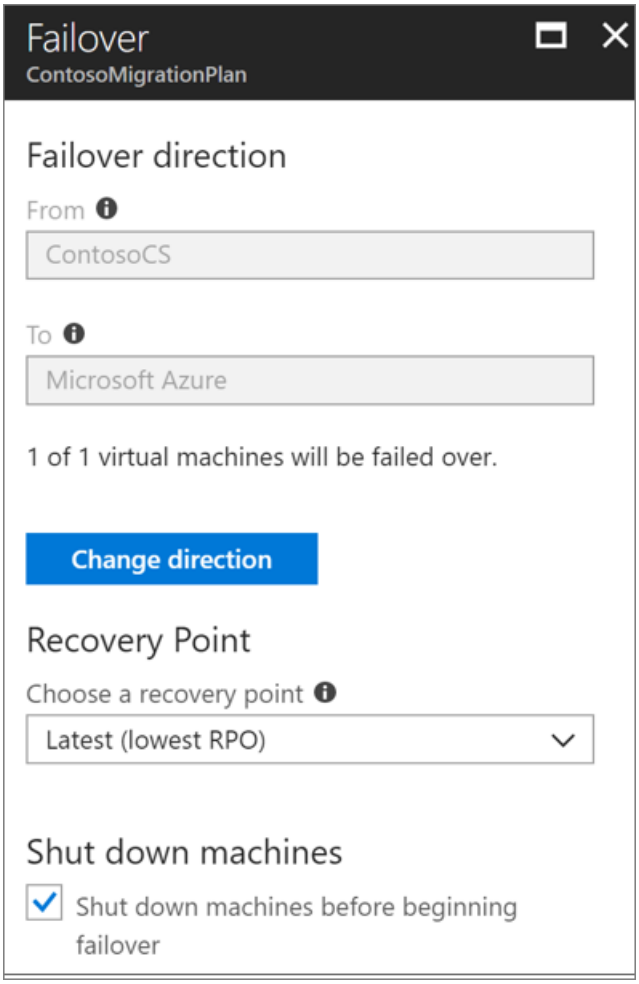
1. Run a test failover to the latest available point in time (`Latest processed`).
2. Select **Shut down machine before beginning failover** so that Azure Migrate attempts to shut down the source VM before triggering the failover. Failover continues even if shutdown fails.
3. A test failover runs:
   - A prerequisites check runs to make sure all of the conditions required for migration are in place.
   - Failover processes the data so that an Azure VM can be created. If the latest recovery point is selected, a recovery point is created from the data.
   - An Azure VM is created by using the data processed in the previous step.
4. After the failover finishes, the replica Azure VM appears in the Azure portal. They check that the VM is the appropriate size, that it's connected to the right network, and that it's running.
5. After verifying, they clean up the failover, and record and save any observations.

Run a failover

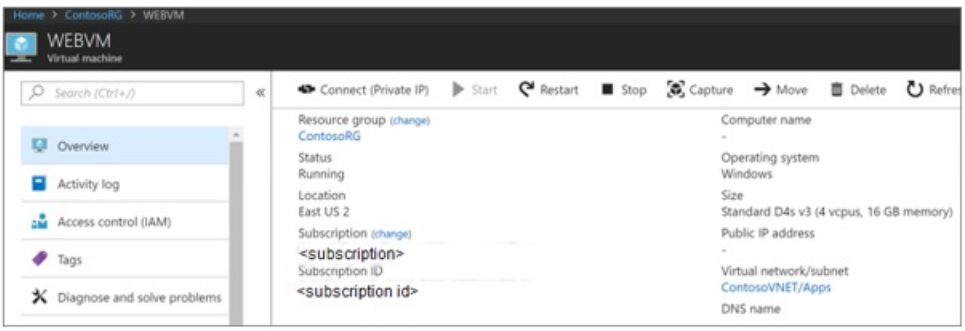
1. After verifying that the test failover worked as expected, they create a recovery plan for migration, and add `WEBVM` to the plan.



2. They run a failover on the plan. They select the latest recovery point. They specify that Azure Migrate should try to shut down the on-premises VM before triggering the failover.

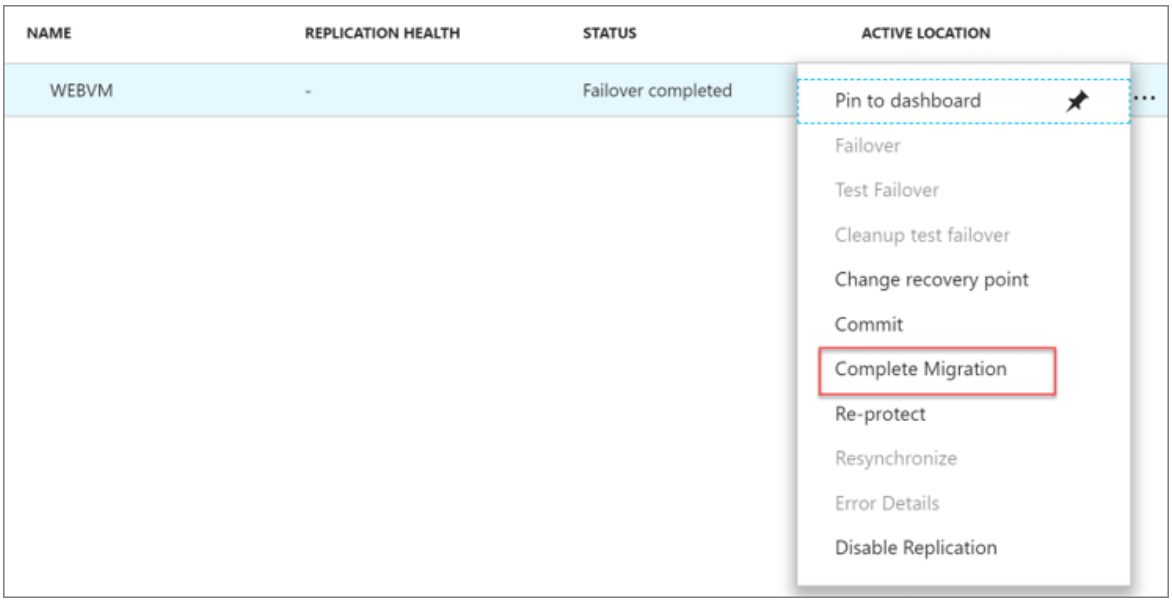


3. After the failover, they verify that the Azure VM appears as expected in the Azure portal.



4. After verifying the VM in Azure, they complete the migration to finish the migration process, stop

replication for the VM, and stop Azure Migrate billing for the VM.



Update the connection string

As the final step in the migration process, the Contoso admins update the connection string of the application to point to the migrated database running on the `SHAOG` listener. This configuration will be changed on the `WEBVM` now running in Azure. This configuration is located in the `web.config` of the ASP.NET application.

1. The Contoso admins locate the file at `C:\inetpub\SmartHotelWeb\web.config` and change the name of the server to reflect the FQDN of the Always On availability group: `shaog.contoso.com`.

```
<connectionStrings>
  <add name="DefaultConnection" connectionString="Data Source=shaog.contoso.com;Database=SmartHotel.Registratio
</connectionStrings>
```

2. After updating the file and saving it, they restart IIS on `WEBVM`. They use `iisreset /restart` from a command prompt.

3. After IIS is restarted, the application now uses the database running on the managed instance.

Need more help?

- Learn about how to run a test failover.
- Learn how to create a recovery plan.
- Learn about failing over to Azure.

Clean up after migration

After migration, the SmartHotel360 application is running on an Azure VM. The `SmartHotel360` database is located in the SQL Server cluster in Azure.

Now, Contoso needs to finish these cleanup steps:

- Remove the on-premises VMs from the vCenter inventory.
- Remove the VMs from local backup jobs.
- Update internal documentation to show the new locations and IP addresses for VMs.
- Review any resources that interact with the decommissioned VMs. Update any relevant settings or documentation to reflect the new configuration.
- Add the two new VMs (`SQLAOG1` and `SQLAOG2`) to production monitoring systems.

Review the deployment

With the migrated resources in Azure, Contoso needs to fully operationalize and secure its new infrastructure.

Security

The Contoso security team reviews the virtual machines `WEBVM`, `SQLAOG1`, and `SQLAOG2` to determine any security issues. They need to:

- Review the network security groups (NSGs) for the VM to control access. NSGs are used to ensure that only traffic allowed to the application can pass.
- Consider securing the data on the disk by using Azure Disk Encryption and Azure Key Vault.
- Evaluate transparent data encryption. Then enable it on the `SmartHotel360` database running on the new Always On availability group. Learn more about transparent data encryption.

For more information, see Security best practices for IaaS workloads in Azure.

Business continuity and disaster recovery

For business continuity and disaster recovery, Contoso takes the following actions:

- To keep data safe, Contoso backs up the data on the `WEBVM`, `SQLAOG1`, and `SQLAOG2` VMs via Azure VM backup.
- Contoso will also learn about how to use Azure Storage to back up SQL Server directly to Azure Blob Storage. Learn more about how to use Azure Storage for SQL Server backup and restore.
- To keep applications up and running, Contoso replicates the application VMs in Azure to a secondary region by using Site Recovery. Learn more about how to set up disaster recovery to a secondary Azure region for an Azure VM.

Licensing and cost optimization

- Contoso has existing licensing for its `WEBVM` and will take advantage of the Azure Hybrid Benefit. Contoso will convert the existing Azure VMs to take advantage of this pricing.
- Contoso will use Azure Cost Management + Billing to ensure the company stays within budgets established by the IT leadership.

Conclusion

In this article, Contoso rehosted the SmartHotel360 application in Azure by migrating the application front-end VM to Azure by using Azure Migrate. Contoso migrated the application database to a SQL Server cluster provisioned in Azure by using Azure Database Migration Service and protected it in a SQL Server Always On availability group.

Migrate open-source databases to Azure

7/15/2021 • 7 minutes to read • Edit Online

This article demonstrates how the fictional company Contoso assessed, planned, and migrated its various on-premises open-source databases to Azure. For step-by-step guidance, you can also watch the Migrate and Modernize Linux VMs and Databases into Azure video.

As Contoso considers migrating to Azure, the company needs a technical and financial assessment to determine whether its on-premises workloads are good candidates for cloud migration. In particular, the Contoso team wants to assess machine and database compatibility for migration. Additionally, it wants to estimate capacity and costs for running Contoso's resources in Azure.

Business drivers

Contoso is having various issues with maintaining the wide array of versions of open-source database workloads that exist on its network. After the latest investor's meeting, the CFO and CTO decided to move all these workloads to Azure. This move will shift them from a structured capital expense model to a fluid operating expense model.

The IT leadership team has worked closely with business partners to understand the business and technical requirements. They want to:

- **Increase security.** Contoso needs to be able to monitor and protect all data resources in a more timely and efficient manner. The company also wants to get a more centralized reporting system set up on database access patterns.
- **Optimize compute resources.** Contoso has deployed a large on-premises server infrastructure. The company has several SQL Server instances that consume but don't really use the underlying CPU, memory, and disk allocated in efficient ways.
- **Increase efficiency.** Contoso needs to remove unnecessary procedures and streamline processes for developers and users. The business needs IT to be fast and not waste time or money to deliver faster on customer requirements. Database administration should be reduced or minimized after the migration.
- **Increase agility.** Contoso IT needs to be more responsive to the needs of the business. It must react faster than the changes in the marketplace to enable success in a global economy. It mustn't get in the way or become a business blocker.
- **Scale.** As the business grows successfully, Contoso IT must provide systems that grow at the same pace.
- **Understand costs.** Business and application owners want to know they won't be stuck with high cloud costs when running the applications on-premises.

Migration goals

The Contoso cloud team has pinned down goals for the various migrations. These goals were used to determine the best migration methods.

REQUIREMENTS	DETAILS
Performance	After migration, applications in Azure should have the same performance capabilities that applications have today in Contoso's on-premises environment. Moving to the cloud doesn't mean that application performance is less critical.

REQUIREMENTS	DETAILS
Compatibility	Contoso needs to understand the compatibility of its applications and databases with Azure. Contoso also needs to understand its Azure hosting options.
Data sources	All databases will be moved to Azure with no exceptions. Based on the database and application analysis of the SQL features being used, they'll move to platform as a service (PaaS) or infrastructure as a service (IaaS). All databases must move.
Application	Applications must be moved to the cloud wherever possible. If they can't move, they'll connect to the migrated database over the Azure network through private connections only.
Costs	Contoso wants to understand not only its migration options but also the costs associated with the infrastructure after it moves to the cloud.
Management	Resource management groups need to be created for the various departments along with resource groups to manage all databases that are migrated. All resources need to be tagged with department information for chargeback requirements.
Limitations	Initially, not all branch offices that run applications will have a direct Azure ExpressRoute link to Azure. These offices will need to connect through virtual network gateways.

Solution design

Contoso has already performed a migration assessment of its digital estate by using Azure Migrate.

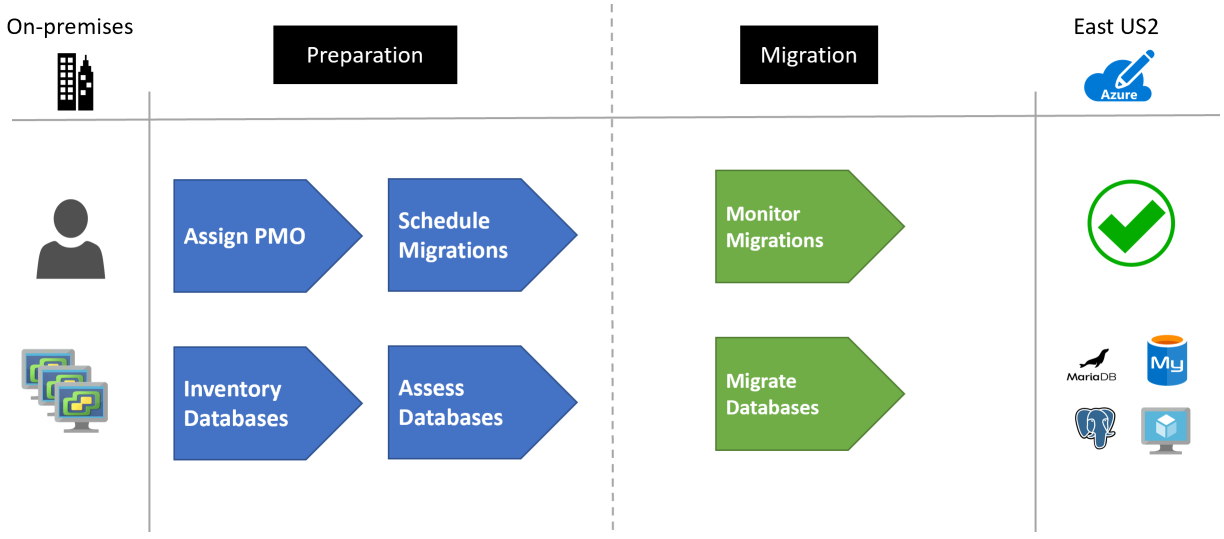


Figure 1: The migration process.

Solution review

Contoso evaluates the proposed design by putting together a pros and cons list.

CONSIDERATION	DETAILS
Pros	Azure will provide a single pane of glass into the database workloads. Costs will be monitored via Azure Cost Management + Billing. Business chargeback billing will be easy to perform with the Azure Billing APIs. Server and software maintenance will be reduced to only the IaaS-based environments.
Cons	Because of the requirement of IaaS-based VMs, there will still be a need to manage the software on those machines.

Budget and management

Before the migration can occur, the necessary Azure structure is required to be in place to support the administration and billing aspects of the solution.

For the management requirements, several management groups were created to support the organizational structure.

For the billing requirements, each of the Azure resources are then tagged with the appropriate billing tags.

Migration process

Data migrations follow a standard and repeatable pattern. This process involves the following steps based on Microsoft best practices:

- Pre-migration:
  - **Discovery:** Inventory database assets and application stack.
  - **Assess:** Assess workloads and fix recommendations.
  - **Convert:** Convert source schema to work in the target.
- Migration:
  - **Migrate:** Migrate the source schema, source data, and objects to target.
  - **Sync data:** Sync data (for minimal downtime).
  - **Cutover:** Cut over the source to target.
- Post-migration:
  - **Remediate applications:** Iteratively make any necessary changes to applications.
  - **Perform tests:** Iteratively run functional and performance tests.
  - **Optimize:** Based on tests, address performance issues and then retest to confirm performance improvements.
  - **Retire assets:** Old VMs and hosting environments are backed up and retired.

Step 1: Discovery

Contoso used Azure Migrate to surface the dependencies across the Contoso environment. Azure Migrate automatically discovered application components on Windows and Linux systems and mapped the communication between services. Azure Migrate also surfaced the connections between Contoso servers, processes, inbound and outbound connection latency, and ports across the TCP-connected architecture. Contoso was only required to install the Microsoft Monitoring Agent and the Microsoft Dependency Agent.

Contoso has identified over 300 database instances that must be migrated. Of these instances, roughly 40 percent can be moved to PaaS-based services. Of the remaining 60 percent, they must be moved to an IaaS-based approach with a VM running the respective database software.

Step 2: Application assessment

The results from the assessment showed Contoso that it uses primarily Java, PHP, and Node.js applications. The company has identified the following applications:

- 100 Java applications
- About 50 Node.js applications
- About 25 PHP applications

Step 3: Database assessment

As the databases were inventoried, each type of database was reviewed to determine the method to migrate it to Azure. The following guidelines were followed on the database migrations.

DATABASE TYPE	DETAILS	TARGET	MIGRATION GUIDE
MySQL	All supported versions upgrade to a supported version before migration	Azure Database for MySQL (PaaS)	Guide
PostgreSQL	All supported versions upgrade to a supported version before migration	Azure Database for PostgreSQL (PaaS)	Guide
MariaDB	All supported versions upgrade to a supported version before migration	Azure Database for MariaDB (PaaS)	Guide

Step 4: Migration planning

Because of the large number of databases, Contoso set up a project management office to keep track of every database migration instance. Accountability and responsibilities were assigned to each business and application team.

Contoso also performed a workload readiness review. This review examined the infrastructure, database, and network components.

Step 5: Test migrations

The first part of the migration preparation involved a test migration of each of the databases to the pre-setup environments. To save time, Contoso scripted all of the operations for the migrations and recorded the timings for each. To speed up the migration, the company identified the migration operations that could run concurrently.

Any rollback procedures were identified for each of the database workloads in case of some unexpected failures.

For the IaaS-based workloads, the company set up all the required third-party software beforehand.

After the test migration, Contoso used the various Azure cost-estimation tools to get a more accurate picture of the future operational costs of the migration.

Step 6: Migration

For the production migration, Contoso identified the time frames for all database migrations and what could be sufficiently executed in a weekend window (midnight Friday through midnight Sunday) with minimal downtime to the business.

Clean up after migration

Contoso identified the archival window for all database workloads. As the window expires, the resources will be deallocated from the on-premises infrastructure. This process includes removing the production data from on-premises servers and retiring the hosting server when the last workload window expires.

Review the deployment

With the migrated resources in Azure, Contoso needs to fully operationalize and secure its new infrastructure.

Security

Contoso needs to:

- Ensure that its new Azure database workloads are secure. For more information, see Azure SQL Database and SQL Managed Instance security capabilities.
- Review the firewall and virtual network configurations.
- Set up Azure Private Link so that all database traffic is kept inside Azure and the on-premises network.
- Enable Microsoft Defender for Identity.

Backups

Ensure that the Azure databases are backed up by using geo-restore. In this way, backups can be used in a paired region if a regional outage occurs.

> **IMPORTANT**
> Make sure that the Azure resource has a resource lock to prevent it from being deleted. Deleted servers can't be restored.

Licensing and cost optimization

- Many Azure database workloads can be scaled up or down. Monitoring server and database performance is important to ensure that you're meeting your needs and keeping costs at a minimum.
- Both CPU and storage have costs associated. There are several pricing tiers to select from. Make sure that the appropriate pricing plan is selected for the data workloads.
- Each read replica is billed based on the compute and storage selected.
- Use reserved capacity to reduce costs.

Conclusion

In this article, Contoso assessed, planned, and migrated its open-source databases to Azure PaaS and IaaS solutions.

Migrate MySQL databases to Azure

5/24/2021 • 7 minutes to read • Edit Online

This article demonstrates how the fictional company Contoso planned and migrated its on-premises MySQL open-source database platform to Azure.

Business drivers

The IT leadership team has worked closely with business partners to understand what they want to achieve with this migration. They want to:

- **Increase availability.** Contoso has had availability issues with its MySQL on-premises environment. The business requires the applications that use this data store to be more reliable.
- **Increase efficiency.** Contoso needs to remove unnecessary procedures and streamline processes for developers and users. The business needs IT to be fast and not waste time or money to deliver faster on customer requirements.
- **Increase agility.** Contoso IT needs to be more responsive to the needs of the business. It must react faster than the changes in the marketplace to enable success in a global economy. It mustn't become a business blocker.
- **Scale.** As the business grows successfully, Contoso IT must provide systems that grow at the same pace.

Migration goals

The Contoso cloud team has pinned down goals for this migration. These goals were used to determine the best migration method.

REQUIREMENTS	DETAILS
Availability	Currently internal staff are having a hard time with the hosting environment for the MySQL instance. Contoso wants to have close to 99.99 percent availability for the database layer.
Scalability	The on-premises database host is quickly running out of capacity. Contoso needs a way to scale its instances past current limitations or scale down if the business environment changes to save on costs.
Performance	The Contoso human resources (HR) department runs various reports daily, weekly, and monthly. When it runs these reports, it experiences significant performance issues with the employee-facing application. It needs to run the reports without affecting application performance.
Security	Contoso needs to know that the database is accessible only to its internal applications and isn't visible or accessible via the internet.
Monitoring	Contoso currently uses tools to monitor the metrics of the MySQL database server and provide notifications when CPU, memory, or storage have issues. The company wants to have this same capability in Azure.

REQUIREMENTS	DETAILS
Business continuity	The HR data store is an important part of Contoso's daily operations. If it became corrupted or needed to be restored, the company wants to minimize downtime as much as possible.
Azure	Contoso wants to move the application to Azure without running it on VMs. Contoso wants to use Azure platform as a service (PaaS) services for the data tier.

Solution design

After pinning down goals and requirements, Contoso designs and reviews a deployment solution and identifies the migration process. The tools and services that it will use for migration are also identified.

Current application

The MySQL database stores employee data that's used for all aspects of the company's HR department. A LAMP-based application is used as the front end to handle employee HR requests. Contoso has 100,000 employees worldwide, so uptime is important.

Proposed solution

Use Azure Database Migration Service to migrate the database to an Azure Database for MySQL instance. Modify all applications and processes to use the new Azure Database for MySQL instance.

Database considerations

As part of the solution design process, Contoso reviewed the features in Azure for hosting its MySQL data. The following considerations helped the company decide to use Azure:

- Similar to Azure SQL Database, Azure Database for MySQL allows for firewall rules.
- Azure Database for MySQL can be used with Azure Virtual Network to prevent the instance from being publicly accessible.
- Azure Database for MySQL has the required compliance and privacy certifications that Contoso must meet for its auditors.
- Report and application processing performance will be enhanced by using read replicas.
- Ability to expose the service to internal network traffic only (no public access) by using Azure Private Link.
- Contoso chose not to move to Azure Database for MySQL because it's considering using the MariaDB ColumnStore and graph database model in the future.
- Aside from MySQL features, Contoso is a proponent of true open-source projects and chose not to use MySQL.
- The bandwidth and latency from the application to the database will be sufficient enough based on the chosen gateway (either Azure ExpressRoute or Site-to-Site VPN).

Solution review

Contoso evaluates the proposed design by putting together a pros and cons list.

CONSIDERATION	DETAILS

CONSIDERATION	DETAILS
Pros	Azure Database for MySQL offers a 99.99 percent financially backed service-level agreement (SLA) for high availability. Azure offers the ability to scale up or down during peak load times each quarter. Contoso can save even more by purchasing reserved capacity. Azure provides point-in-time restore and geo-restore capabilities for Azure Database for MySQL.
Cons	Contoso is limited to the MySQL release versions that are supported in Azure, which are currently 10.2 and 10.3. Azure Database for MySQL has some limitations, such as scaling down storage.

Proposed architecture

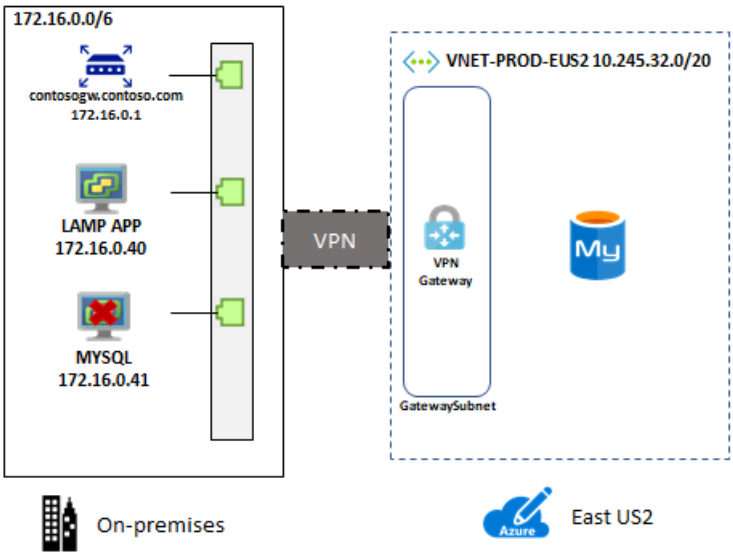


Figure 1: Scenario architecture.

Migration process

Preparation

Before you can migrate your MySQL databases, you need to ensure that those instances meet all the Azure prerequisites for a successful migration.

Supported versions

MySQL uses the `x.y.z` versioning scheme, where `x` is the major version, `y` is the minor version, and `z` is the patch version.

Azure currently supports MySQL versions 10.2.25 and 10.3.16.

Azure automatically manages upgrades for patch updates. Examples are 10.2.21 to 10.2.23. Minor and major version upgrades aren't supported. For example, upgrading from MySQL 10.2 to MySQL 10.3 isn't supported. If you want to upgrade from 10.2 to 10.3, take a dump and restore it to a server created with the new engine version.

Network

Contoso needs to set up a virtual network gateway connection from its on-premises environment to the virtual

network where its MySQL database is located. This connection allows the on-premises application to access the database over the gateway when the connection strings are updated.

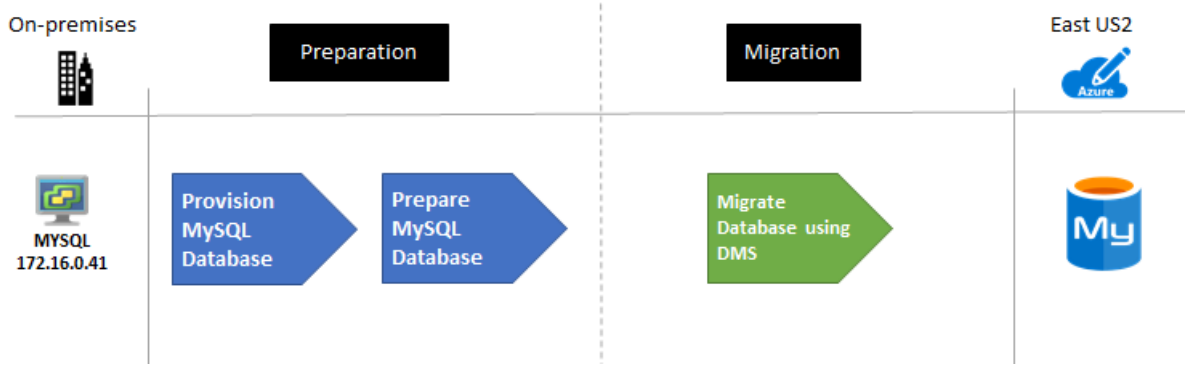


Figure 2: The migration process.

Migration

Contoso admins migrate the database by using Azure Database Migration Service and following the step-by-step migration tutorial. They can perform online, offline, and hybrid (preview) migrations by using MySQL 5.6 or 5.7.

> **NOTE**
> MySQL 8.0 is supported in Azure Database for MySQL. The Database Migration Service tool doesn't yet support that version.

As a summary, they must do the following tasks:

- Ensure all migration prerequisites are met:
  - The MySQL database server source must match the version that Azure Database for MySQL supports. Azure Database for MySQL supports MySQL Community Edition, the InnoDB storage engine, and migration across source and target with the same versions.
  - Enable binary logging in `my.ini` (Windows) or `my.cnf` (Unix). Failure to enable binary logging causes the following error in the Migration Wizard:
    `Error in binary logging. Variable binlog_row_image has value 'minimal'. Please change it to 'full'.`
    For more information, see the MySQL documentation.
  - User must have the `ReplicationAdmin` role.
  - Migrate the database schemas without foreign keys and triggers.
- Create a virtual network that connects via ExpressRoute or a VPN to your on-premises network.
- Create an Azure Database Migration Service instance with a `Premium` SKU that's connected to the virtual network.
- Ensure that the instance can access the MySQL database via the virtual network. Make sure that all incoming ports are allowed from Azure to MySQL at the virtual network level, the network VPN, and the machine that hosts MySQL.
- Create a new Database Migration Service project:

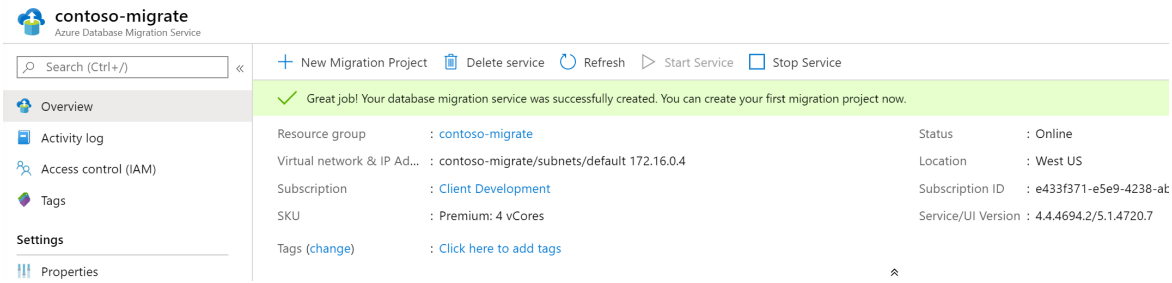


Figure 3: An Azure Database Migration Service project.

Migration by using native tools

As an alternative to using Azure Database Migration Service, Contoso can use common utilities and tools such as MySQL Workbench, mysqldump, Toad, or Navicat to connect to and migrate data to Azure Database for MySQL.

- Dump and restore with mysqldump:
  - Use the exclude-triggers option in mysqldump to prevent triggers from executing during import and improve performance.
  - Use the single-transaction option to set the translation isolation mode to `REPEATABLE READ`, and send a `START TRANSACTION` SQL statement before you dump data.
  - Use the disable-keys option in mysqldump to disable foreign key constraints before load. Removing constraints provides performance gains.
  - Use Azure Blob Storage to store the backup files and perform the restore from there for faster restore.
  - Update application connection strings.
  - After the database is migrated, Contoso must update the connection strings to point to the new Azure Database for MySQL.

Clean up after migration

After migration, Contoso needs to back up the on-premises database for retention purposes and retire the on-premises MySQL database server.

Review the deployment

With the migrated resources in Azure, Contoso needs to fully operationalize and secure its new infrastructure.

Security

Contoso needs to:

- Ensure that its new Azure Database for MySQL instance and databases are secure. For more information, see Security in Azure Database for MySQL.
- Review the firewall and virtual network configurations.
- Set up Private Link so that all database traffic is kept inside Azure and the on-premises network.
- Enable Microsoft Defender for Identity.

Backups

Ensure that the Azure Database for MySQL instances are backed up by using geo-restore, so that backups can be used in a paired region if a regional outage occurs.

> **IMPORTANT**
> Ensure that the Azure Database for MySQL resource has a resource lock to prevent it from being deleted. Deleted servers can't be restored.

Licensing and cost optimization

- Azure Database for MySQL can be scaled up or down. Monitoring the performance of the server and databases is important to ensure your requirements are met while minimizing costs.
- Both CPU and storage have costs associated. Several pricing tiers are available. Be sure the appropriate pricing plan is selected for each data workload.
- Each read replica is billed based on the compute and storage selected.
- Use reserved capacity to save on costs.

Conclusion

In this article, Contoso migrated its MySQL databases to an Azure Database for MySQL instance.

Migrate PostgreSQL databases to Azure

5/26/2021 • 9 minutes to read • Edit Online

This article demonstrates how the fictional company Contoso planned and migrated its on-premises PostgreSQL open-source database platform to Azure.

Business drivers

The IT leadership team has worked closely with business partners to understand what they want to achieve with this migration. They want to:

- **Automate big data.** Contoso uses PostgreSQL for several of its big data and AI initiatives. The company wants to build scalable repeatable pipelines to automate many of these analytical workloads.
- **Increase efficiency.** Contoso needs to remove unnecessary procedures and streamline processes for developers and users. The business needs IT to be fast and not waste time or money to deliver quicker on customer requirements.
- **Increase agility.** Contoso IT needs to be more responsive to the needs of the business. It must react faster than the changes in the marketplace to enable success in a global economy and to not become a business blocker.
- **Scale.** As the business grows successfully, Contoso IT must provide systems that can grow at the same pace.
- **Increase security.** Contoso realizes that regulatory issues will cause the company to adjust its on-premises strategy based on auditing, logging, and compliance requirements.

Migration goals

The Contoso cloud team has pinned down goals for this migration and will use them to determine the best migration method.

REQUIREMENTS	DETAILS
Upgrades	Contoso wants to ensure that it has the latest patches installed when they're available, but the company doesn't want to manage these updates.
Integrations	Contoso wants to integrate the data in the database with data and AI pipelines for machine learning.
Backup and restore	Contoso is looking for the ability to do point-in-time restores when and if data updates fail or are corrupted for any reason.
Azure	Contoso wants to monitor system and fire alerts based on performance and security.
Performance	In some cases, Contoso will have parallel data processing pipelines in different geographic regions and must read data from those regions.

Solution design

After pinning down goals and requirements, Contoso designs and reviews a deployment solution and identifies the migration process. The tools and services it will use for migration are also identified.

Current environment

PostgreSQL 9.6.7 is running on a physical Linux machine (`sql-pg-01.contoso.com`) in the Contoso datacenter. Contoso already has an Azure subscription with a Site-to-Site VPN gateway to an on-premises datacenter network.

Proposed solution

- Use Azure Database Migration Service to migrate the database to an Azure Database for PostgreSQL instance.
- Modify all applications and processes to use the new Azure Database for PostgreSQL instance.
- Build a new data processing pipeline using Azure Data Factory that connects to the Azure Database for PostgreSQL instance.

Database considerations

As part of the solution design process, Contoso reviewed the features in Azure for hosting its PostgreSQL data. The following considerations helped the company decide to use Azure:

- Similar to Azure SQL Database, Azure Database for PostgreSQL supports firewall rules.
- Azure Database for PostgreSQL can be used with virtual networks to prevent the instance from being publicly accessible.
- Azure Database for PostgreSQL has the required compliance certifications that Contoso must meet.
- Integration with DevOps and Azure Data Factory will allow for automated data processing pipelines to be built.
- Processing performance can be enhanced by using read replicas.
- Support for bring your own key (BYOK) for data encryption.
- Ability to expose the service to internal network traffic only (no-public access) by using Azure Private Link.
- The bandwidth and latency from the application to the database will be sufficient enough based on the chosen gateway (either Azure ExpressRoute or Site-to-Site VPN).

Solution review

Contoso evaluates its proposed design by putting together a list of pros and cons.

CONSIDERATION	DETAILS
Pros	All currently required and in-use features are available in Azure Database for PostgreSQL.
Cons	Contoso will still need to do manual migration from a major version of PostgreSQL.

Proposed architecture

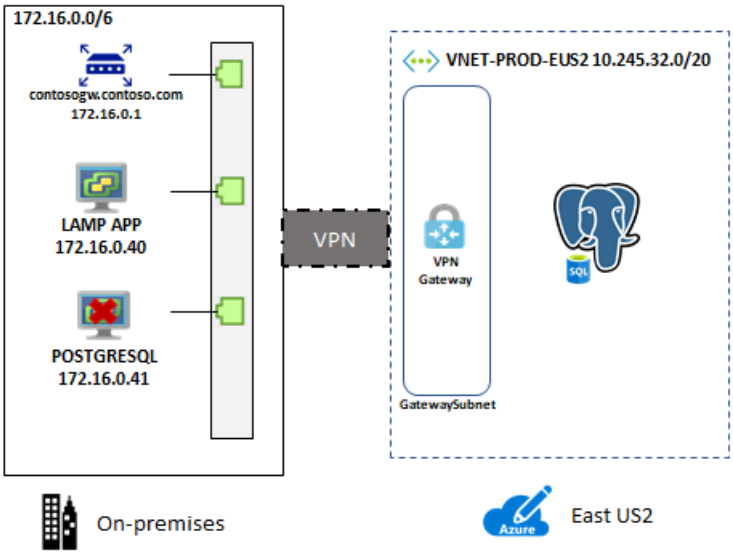


Figure 1: Scenario architecture.

Migration process

Preparation

Before Contoso can migrate its PostgreSQL databases, it ensures that Contoso's instances meet all the Azure prerequisites for a successful migration.

Supported versions

Only migrations to the same or a higher version are supported. Migrating PostgreSQL 9.5 to Azure Database for PostgreSQL 9.6 or 10 is supported, but migrating from PostgreSQL 11 to PostgreSQL 9.6 isn't supported.

Microsoft aims to support *n-2* versions of the PostgreSQL engine in Azure Database for PostgreSQL - Single Server. The versions would be the current major version on Azure (*n*) and the two prior major versions (*-2*).

For the latest updates on supported versions, see Supported PostgreSQL major versions.

> **NOTE**
> Automatic major version upgrade isn't supported. For example, there isn't an automatic upgrade from PostgreSQL 9.5 to PostgreSQL 9.6. To upgrade to the next major version, dump the database and restore it to a server created with the target engine version.

Network

Contoso will need to set up a virtual network gateway connection from its on-premises environment to the virtual network where its Azure Database for PostgreSQL database is located. This connection allows the on-premises application to access the database but not be migrated to the cloud.

Assessment

Contoso will need to assess the current database for replication issues. These issues include:

- The source database version is compatible for migration to the target database version.
- Primary keys must exist on all tables to be replicated.
- Database names can't include a semicolon (`;`).
- Migration of multiple tables with the same name, but a different case might cause unpredictable behavior.

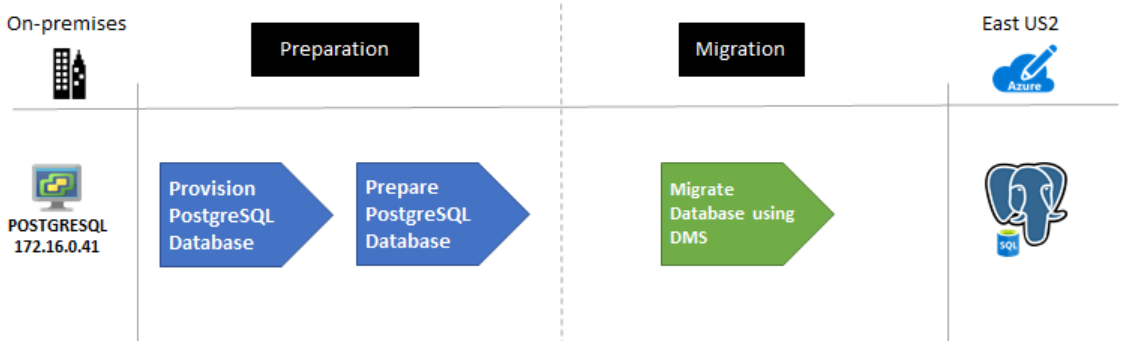


Figure 2: The migration process.

Migration:

Contoso can perform the migration in several ways:

- Dump and restore
- Azure Database Migration Service
- Import/export

Contoso has selected Azure Database Migration Service to allow the company to reuse the migration project whenever it needs to perform major-to-major upgrades. Because a single Database Migration Service activity only accommodates up to four databases, Contoso sets up several jobs by using the following steps.

To prepare, set up a virtual network to access the database. Create a virtual network connection by using VPN gateways in various ways.

Create an Azure Database Migration Service instance

1. In the Azure portal, select **Add a resource**.
2. Search for **Azure Database Migration Service**, and select it.
3. Select **+ Add**.
4. Select the subscription and resource group for the service.
5. Enter a name for the instance.
6. Select the closest location to the Contoso datacenter or VPN gateway.
7. Select **Azure** for the service mode.
8. Select a pricing tier.
9. Select **Review + create**.

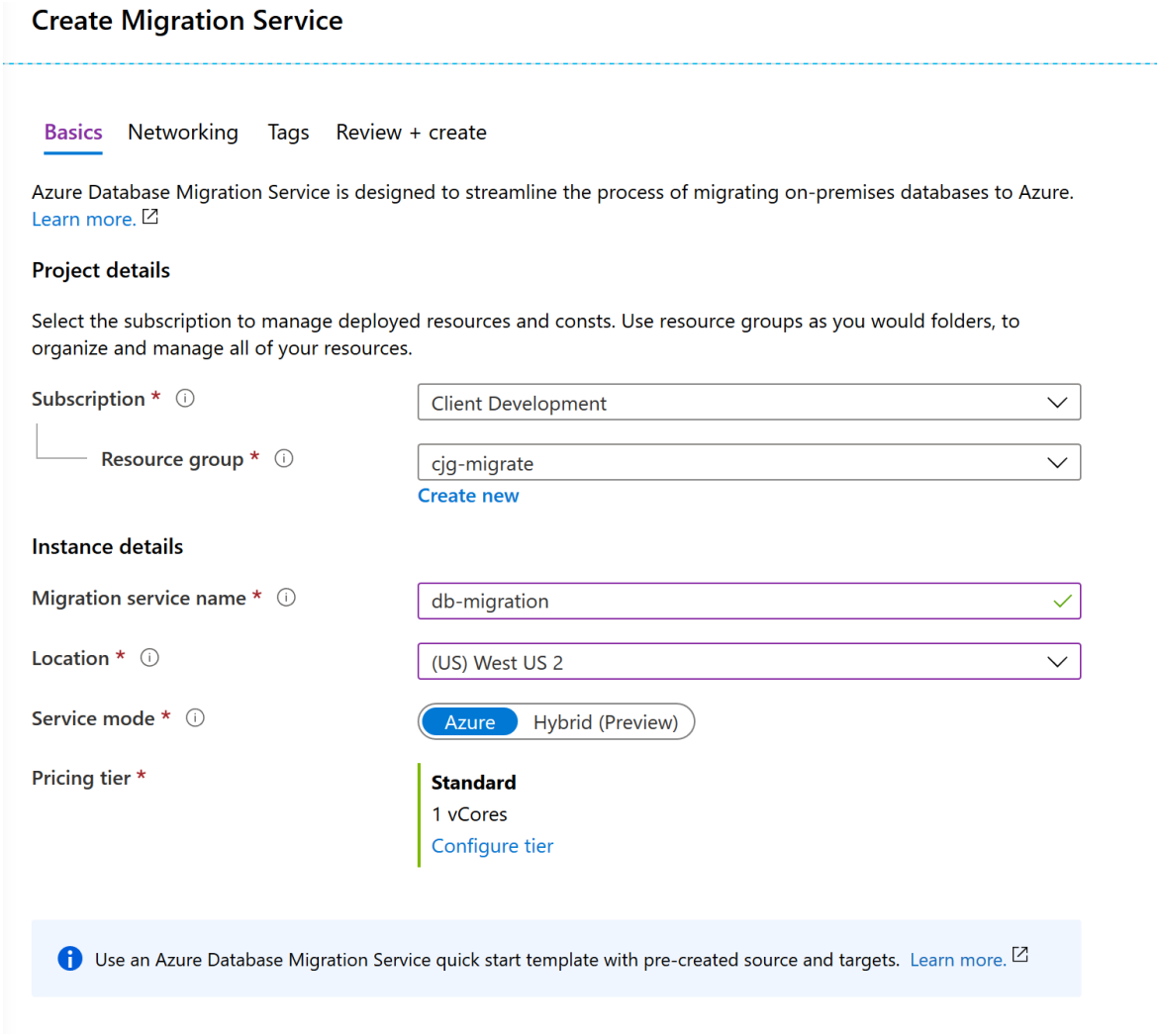


Figure 3: Review and create.

10. Select **Create**.

Create an Azure Database for PostgreSQL instance

1. On the on-premises server, configure the `postgresql.conf` file.
2. Set the server to listen on the proper IP address that Azure Database Migration Service will use to access the server and databases.
   - Set the `listen_addresses` variable.
3. Enable SSL.
   a. Set the `ssl=on` variable.
   b. Verify that Contoso is using a publicly signed SSL certificate for the server that supports TLS 1.2. Otherwise, the Database Migration Service tool will raise an error.
4. Update the `pg_hba.conf` file.
   - Add entries that are specific to the Database Migration Service instance.
5. Logical replication must be enabled on the source server by modifying the values in the `postgresql.conf` file for each server.
   a. `wal_level` = `logical`
   b. `max_replication_slots` = [at least the maximum number of databases for migration]
      - For example, if Contoso wants to migrate four databases, it sets the value to 4.
   c. `max_wal_senders` = [number of databases running concurrently]
      - The recommended value is 10.

6. Migration `User` must have the `REPLICATION` role on the source database.
7. Add the Database Migration Service instance IP address to the `PostgreSQLpg_hba.conf` file.
8. To export the database schemas, run the following commands:

```
pg_dump -U postgres -s dvdrental > dvdrental_schema.sql
```

9. Copy the file, name the copy `dvdrental_schema_foreign.sql`, and remove all non-foreign key and trigger-related items.
10. Remove all foreign key and trigger-related items from the `dvdrental_schema.sql` file.
11. Import the database schema (step 1):

```
psql -h {host}.postgres.database.azure.com -d dvdrental -U username -f dvdrental_schema.sql
```

Migration:

1. In the Azure portal, Contoso goes to its Database Migration Service resource.
2. If the service isn't started, select **Start Service**.
3. Select **New Migration Project**.

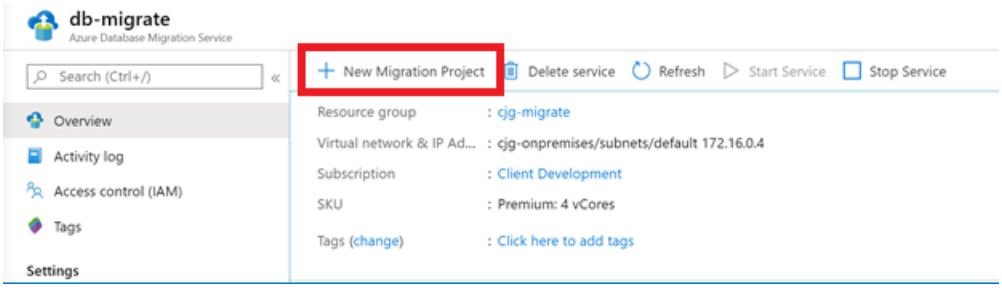


Figure 4: Starting a new migration.

4. Select **New Activity** > **Online data migration**.
5. Enter a name.
6. Select **PostgreSQL** as the source.
7. For the target, select **Azure Database for PostgreSQL** and then select **Save**.

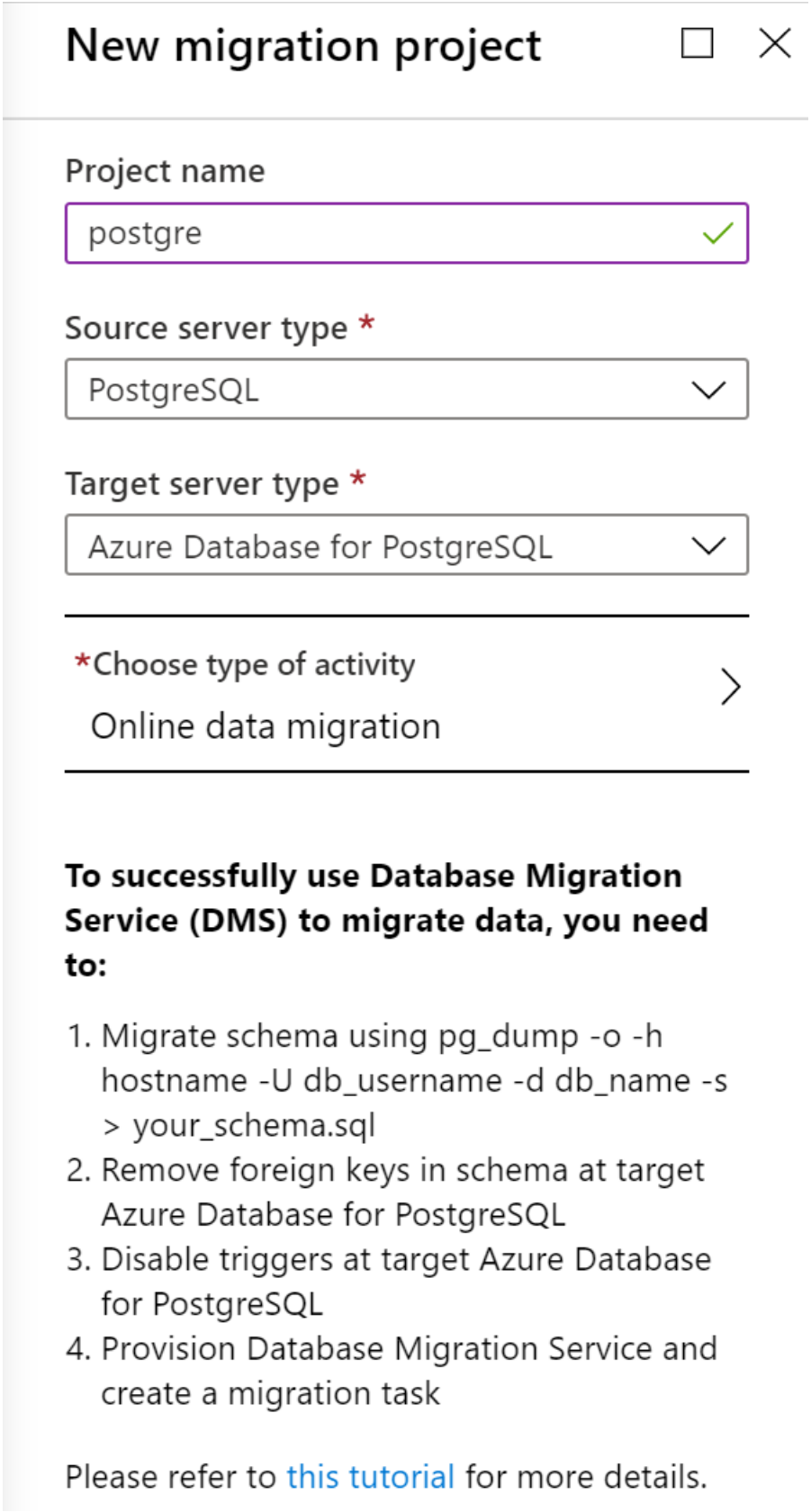


Figure 5: A new migration project is highlighted.

8. Enter the source information, and select **Save**.

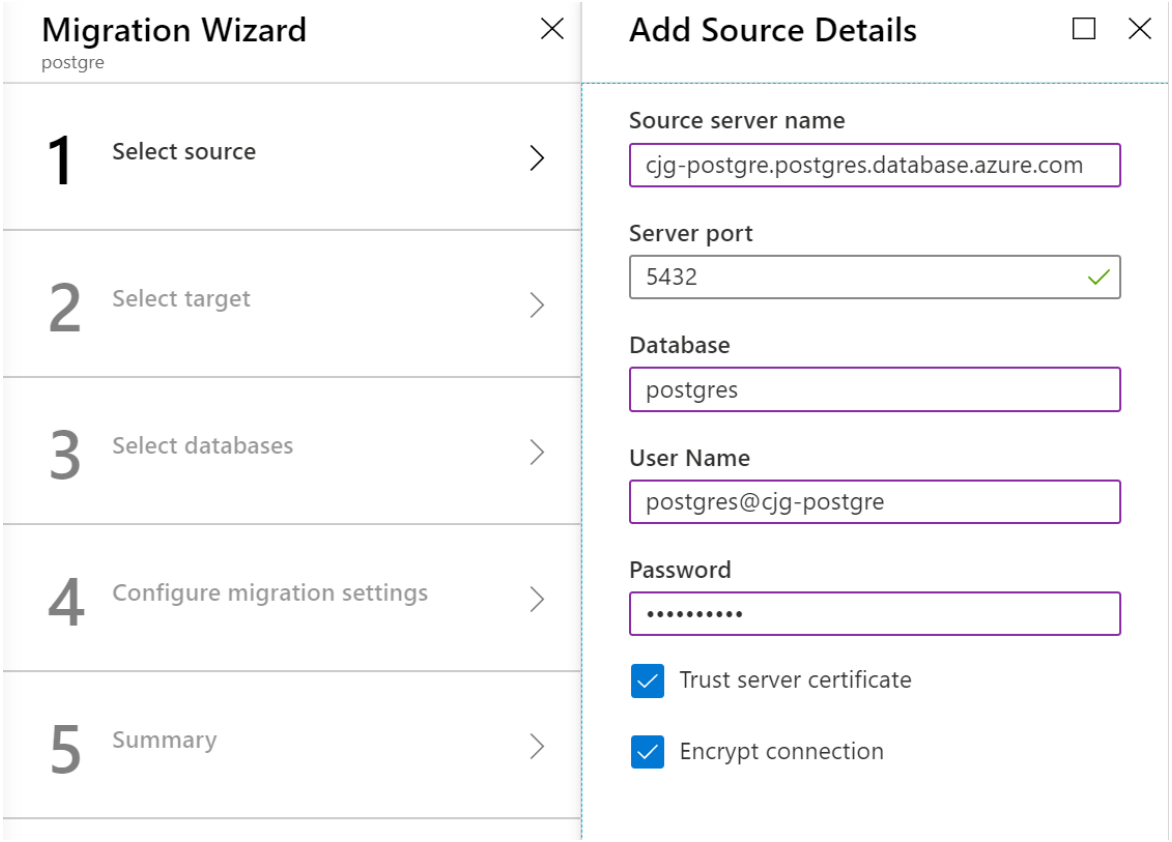


Figure 6: Entering source information.

9. Enter the target information, and select **Save**.

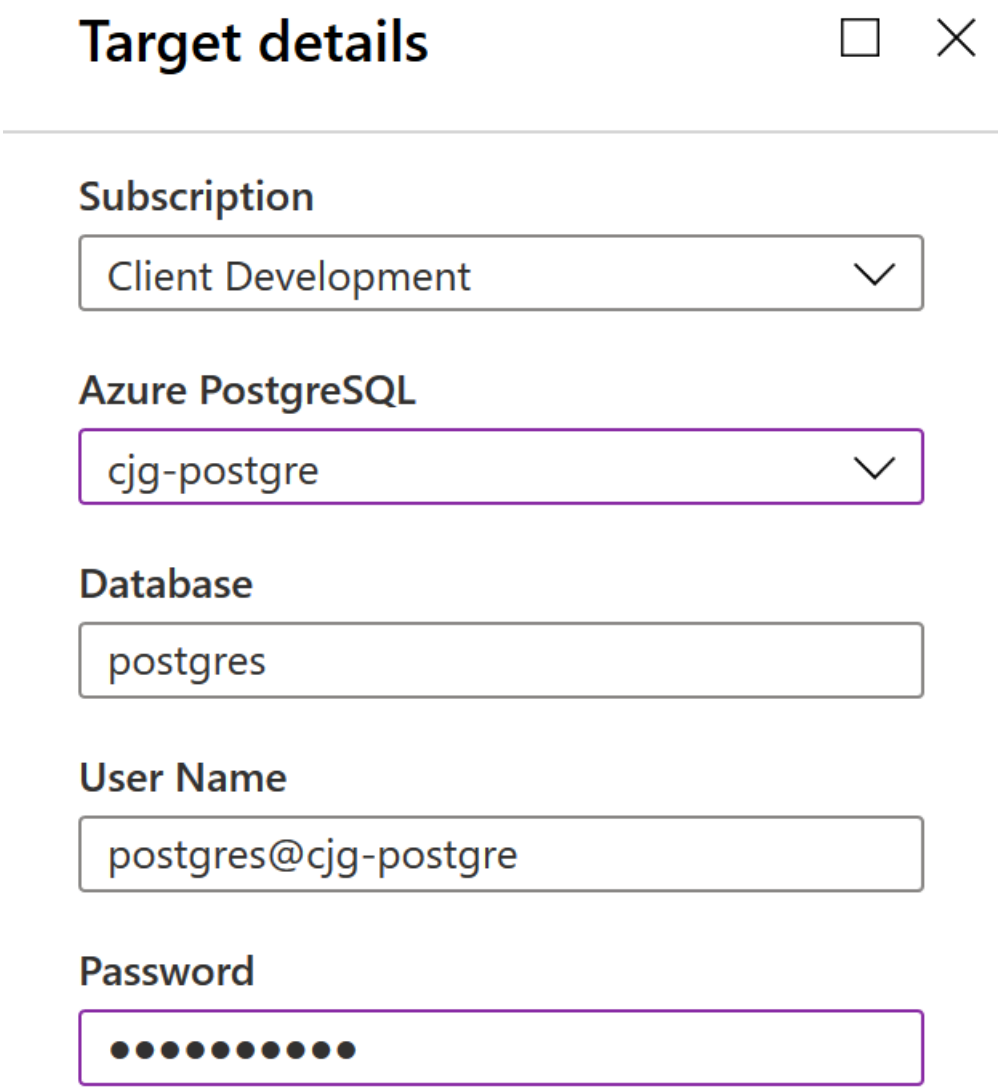


Figure 7: Selecting target information.

10. Select the databases to migrate. The schema for each database should have been migrated previously. Then select **Save**.

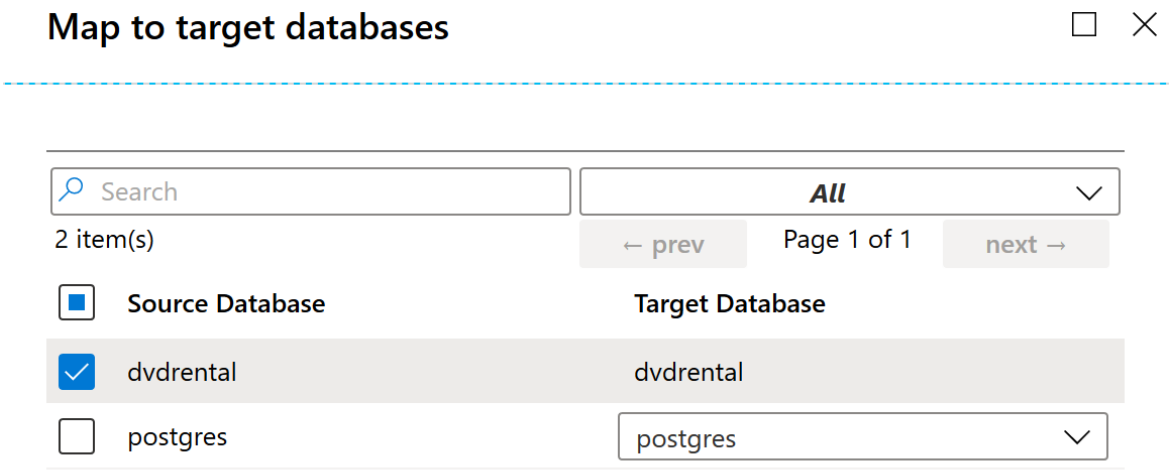


Figure 8: Selecting databases.

11. Configure the advanced settings, and select **Save**.

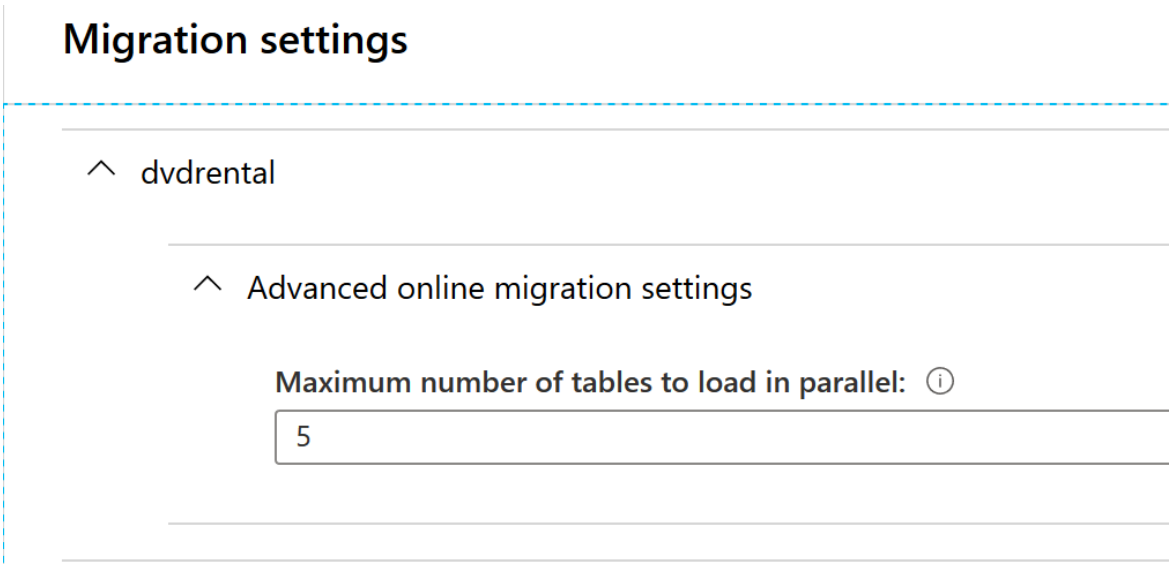


Figure 9: Configuring advanced settings.

12. Give the activity a name, and select **Run**.

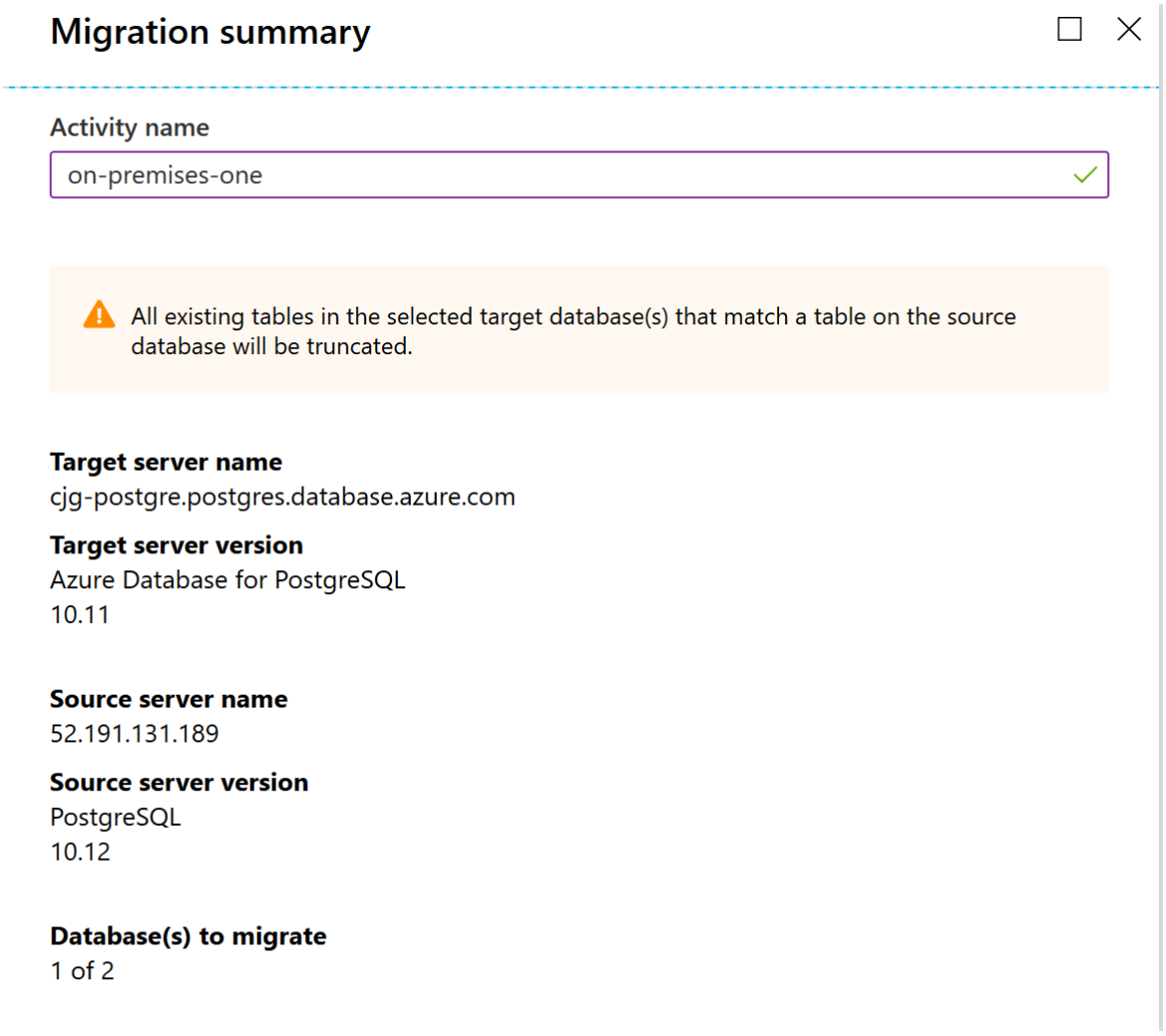


Figure 10: Naming and running the activity.

13. Monitor the migration. Retry it if anything fails. An example is if foreign key references were missing.

14. After `Full load completed` matches the table count, select **Start Cutover**.

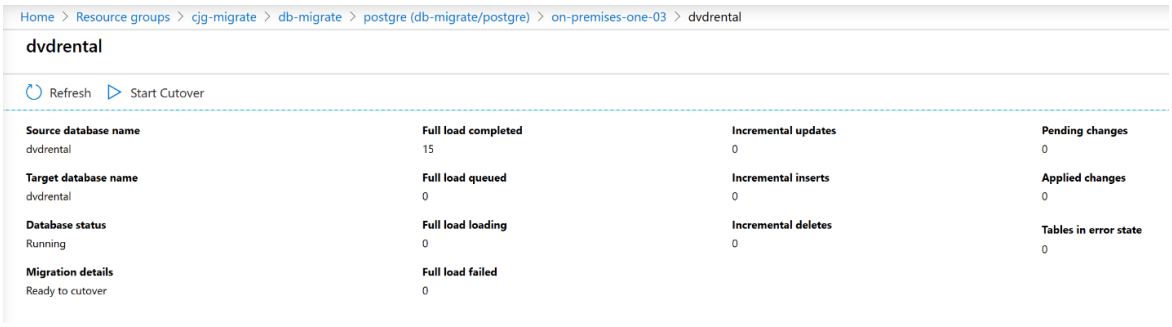


Figure 11: Monitoring the migration to start the cutover.

15. Stop all transactions from the source server.

16. Select the **Confirm** check box, and then select **Apply**.

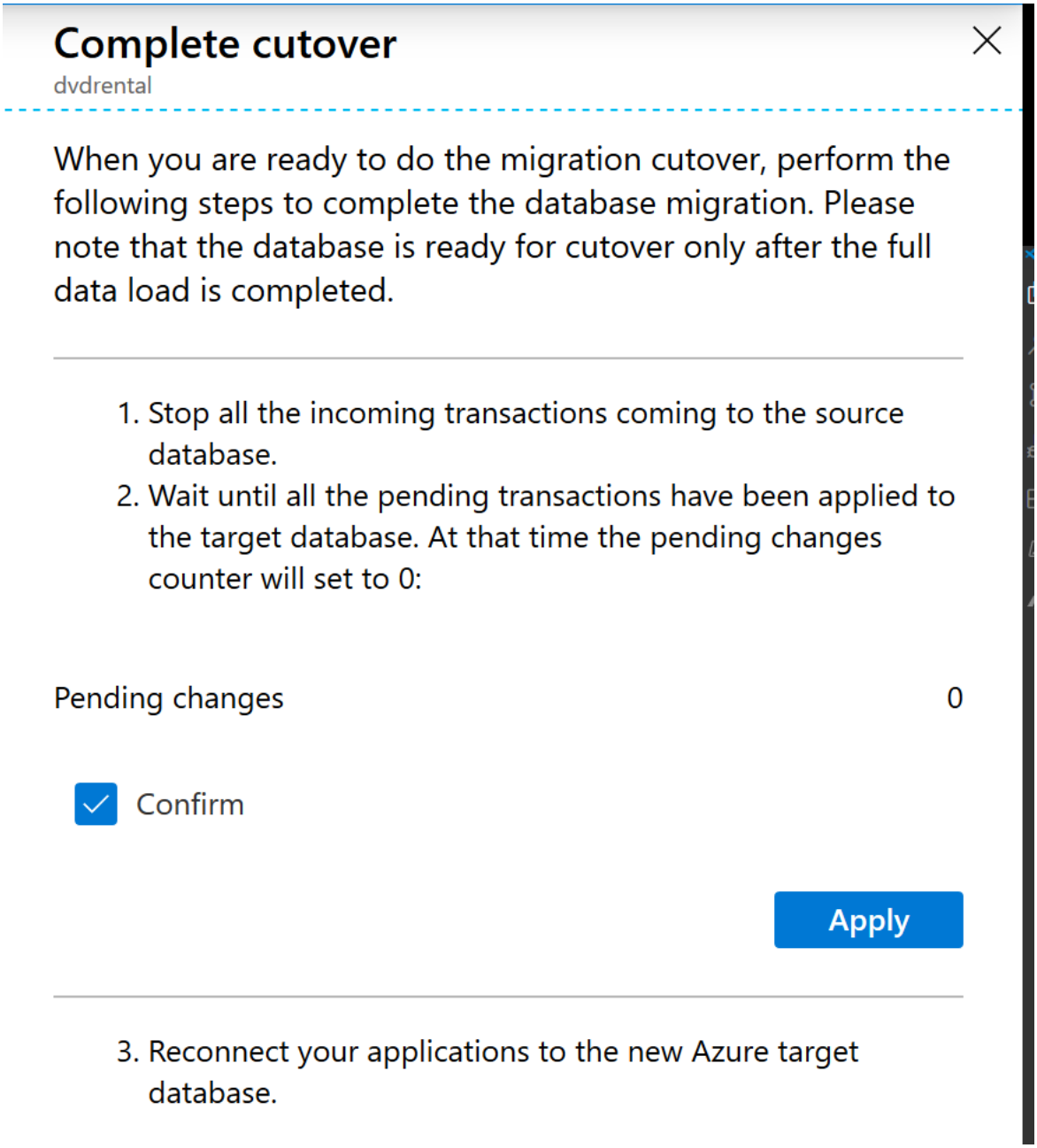


Figure 12: Running the cutover.

17. Wait for the cutover to complete.

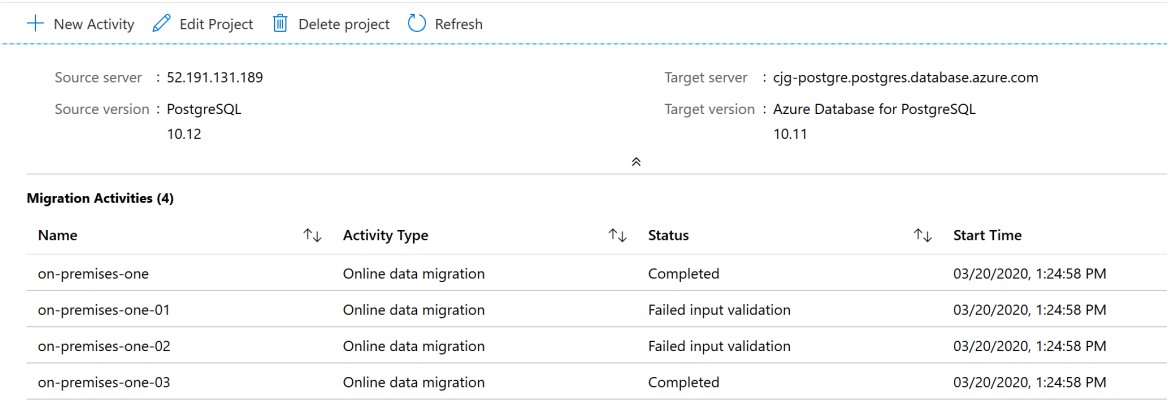


Figure 13: Completing the cutover.

> **NOTE**
> The previous Database Migration Service steps can also be performed via the Azure CLI.

18. Import the database schema (step 2):

```
psql -h {host}.postgres.database.azure.com -d dvdrental -U username -f dvdrental_schema_foreign.sql
```

19. Reconfigure any applications or processes that use the on-premises database to point to the new Azure Database for PostgreSQL database instance.

20. For post-migration, Contoso will ensure that it also set up cross-region read replicas, if necessary, after the migration is finished.

Clean up after migration

After migration, Contoso needs to back up the on-premises database for retention purposes and retire the old PostgreSQL server as part of the cleanup process.

Review the deployment

With the migrated resources in Azure, Contoso needs to fully operationalize and secure its new infrastructure.

Security

Contoso needs to:

- Ensure that the new Azure Database for PostgreSQL instance and databases are secure. For more information, see Security in Azure Database for PostgreSQL: Single server.
- Review the firewall rules and virtual network configurations to verify that connections are limited to only the applications that require it.
- Implement BYOK for data encryption.
- Update all applications to require SSL connections to the databases.
- Set up Private Link so that all database traffic is kept inside Azure and the on-premises network.
- Enable Microsoft Defender for Identity.
- Configure Log Analytics to monitor and alert on security and logs entries of interest.

Backups

Ensure that the Azure Database for PostgreSQL databases is backed up by using geo-restore. In this way, backups can be used in a paired region if a regional outage occurs.

> **IMPORTANT**
> Make sure that the Azure Database for PostgreSQL resource has a resource lock to prevent it from being deleted. Deleted servers can't be restored.

Licensing and cost optimization

- Azure Database for PostgreSQL can be scaled up or down. Performance monitoring of the server and databases is important to ensure that needs are met while keeping costs at a minimum.
- Both CPU and storage have costs associated. There are several pricing tiers to select from. Be sure the appropriate pricing plan is selected for the data workloads.
- Each read replica is billed based on the compute and storage selected.

Conclusion

In this article, Contoso migrated its PostgreSQL databases to an Azure Database for PostgreSQL instance.

Migrate MariaDB databases to Azure

3/6/2021 • 8 minutes to read • Edit Online

This article demonstrates how the fictional company Contoso planned and migrated its on-premises MariaDB open-source database platform to Azure.

Contoso is using MariaDB instead of MySQL because of its:

- Numerous storage engine options.
- Cache and index performance.
- Open-source support with features and extensions.
- ColumnStore storage engine for analytical workloads.

The company's migration goal is to continue to use MariaDB but not worry about managing the environment needed to support it.

Business drivers

The IT leadership team has worked closely with business partners to understand what they want to achieve with this migration. They want to:

- **Increase availability.** Contoso has had availability issues with its MariaDB on-premises environment. The business requires the applications that use this data store to be more reliable.
- **Increase efficiency.** Contoso needs to remove unnecessary procedures and streamline processes for developers and users. The business needs IT to be fast and not waste time or money to deliver faster on customer requirements.
- **Increase agility.** Contoso IT needs to be more responsive to the needs of the business. It must react faster than the changes in the marketplace to enable success in a global economy. It mustn't get in the way or become a business blocker.
- **Scale.** As the business grows successfully, Contoso IT must provide systems that grow at the same pace.

Migration goals

The Contoso cloud team has pinned down goals for this migration. These goals were used to determine the best migration method.

REQUIREMENTS	DETAILS
Availability	Currently internal staff are having a hard time with the hosting environment for the MariaDB instance. Contoso wants to have close to 99.99 percent availability for the database layer.
Scalability	The on-premises database host is quickly running out of capacity. Contoso needs a way to scale its instances past current limitations or scale down if the business environment changes to save on costs.

REQUIREMENTS	DETAILS
Performance	The Contoso human resources (HR) department has several reports it runs on a daily, weekly, and monthly basis. When it runs these reports, it notices considerable performance issues with the employee-facing application. It needs to run the reports without affecting application performance.
Security	Contoso needs to know that the database is accessible only to its internal applications and isn't visible or accessible via the internet.
Monitoring	Contoso currently uses tools to monitor the metrics of the MariaDB database and provide notifications when CPU, memory, or storage have issues. The company wants to have this same capability in Azure.
Business continuity	The HR data store is an important part of Contoso's daily operations. If it became corrupted or needed to be restored, the company wants to minimize downtime.
Azure	Contoso wants to move the application to Azure without running it on VMs. Contoso requirements state to use Azure platform as a service (PaaS) services for the data tier.

Solution design

After pinning down goals and requirements, Contoso designs and reviews a deployment solution and identifies the migration process. The tools and services that it will use for migration are also identified.

Current application

The MariaDB database hosts employee data that's used for all aspects of the company's HR department. A LAMP-based application is used as the front end to handle employee HR requests. Contoso has 100,000 employees worldwide, so uptime is important for its databases.

Proposed solution

- Evaluate the environments for migration compatibility.
- Use common open-source tools to migrate databases to the Azure Database for MariaDB instance.
- Modify all applications and processes to use the new Azure Database for MariaDB instance.

Database considerations

As part of the solution design process, Contoso reviewed the features in Azure for hosting its MariaDB databases. The following considerations helped the company decide to use Azure:

- Similar to Azure SQL Database, Azure Database for MariaDB allows for firewall rules.
- Azure Database for MariaDB can be used with Azure Virtual Network to prevent the instance from being publicly accessible.
- Azure Database for MariaDB has the required compliance and privacy certifications that Contoso must meet for its auditors.
- Report and application processing performance will be enhanced by using read replicas.
- Ability to expose the service to internal network traffic only (no-public access) by using Azure Private Link.
- Contoso chose not to move to Azure Database for MySQL because it's looking at potentially using the MariaDB ColumnStore and graph database model in the future.
- The bandwidth and latency from the application to the database will be sufficient enough based on the

chosen gateway (either Azure ExpressRoute or Site-to-Site VPN).

Solution review

Contoso evaluates the proposed design by putting together a pros and cons list.

CONSIDERATION	DETAILS
Pros	Azure Database for MariaDB offers a 99.99 percent financially backed service-level agreement (SLA) for high availability. Azure offers the ability to scale up or down during peak load times each quarter. Contoso can save even more by purchasing reserved capacity. Azure provides point-in-time restore and geo-restore capabilities for Azure Database for MariaDB.
Cons	Contoso is limited to the MariaDB release versions that are supported in Azure, which are currently 10.2 and 10.3. Azure Database for MariaDB has some limitations, such as scaling down storage.

Proposed architecture

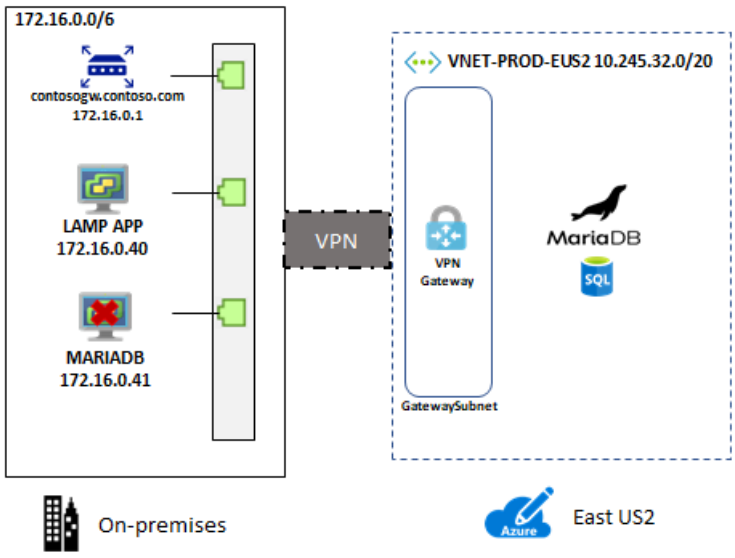


Figure 1: Scenario architecture.

Migration process

Preparation

Before you can migrate your MariaDB databases, you need to ensure that those instances meet all the Azure prerequisites for a successful migration.

Supported versions:

- MariaDB uses the x.y.z naming scheme. For example, x is the major version, y is the minor version, and z is the patch version.
- Azure currently supports 10.2.25 and 10.3.16.
- Azure automatically manages upgrades for patch updates. Examples are 10.2.21 to 10.2.23. Minor and major version upgrades aren't supported. For example, upgrading from MariaDB 10.2 to MariaDB 10.3 isn't supported. If you want to upgrade from 10.2 to 10.3, take a database dump and restore it to a server created

with the target engine version.

The network:

Contoso needs to set up a virtual network gateway connection from its on-premises environment to the virtual network where its MariaDB database is located. This connection allows the on-premises application to access the database over the gateway when the connection strings are updated.

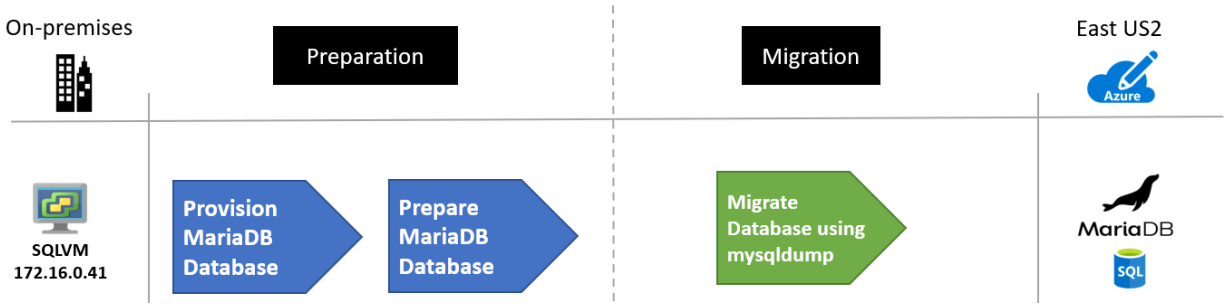


Figure 2: The migration process.

Migration

Because MariaDB is similar to MySQL, Contoso can use the same common utilities and tools such as MySQL Workbench, mysqldump, Toad, or Navicat to connect to and migrate data to Azure Database for MariaDB.

Contoso used the following steps to migrate its databases.

- Determine the on-premises MariaDB version by running the following commands and observing the output. In most cases, your version shouldn't matter much for the schema and data dump. If you're using features at the application level, ensure those applications are compatible with the target version in Azure.

  ```
  mysql -h localhost -u root -P
  ```

  ```
  C:\Users\given>mysql -h localhost -u root -p
  Enter password: **********
  Welcome to the MariaDB monitor.  Commands end with ; or \g.
  Your MariaDB connection id is 13
  Server version: 10.1.44-MariaDB mariadb.org binary distribution

  Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

  Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

  MariaDB [(none)]>
  ```

  Figure 3: Determining the on-premises MariaDB version.

- Create a new MariaDB instance in Azure:
  - Open the Azure portal.
  - Select **Add a resource**.
  - Search for `MariaDB`.

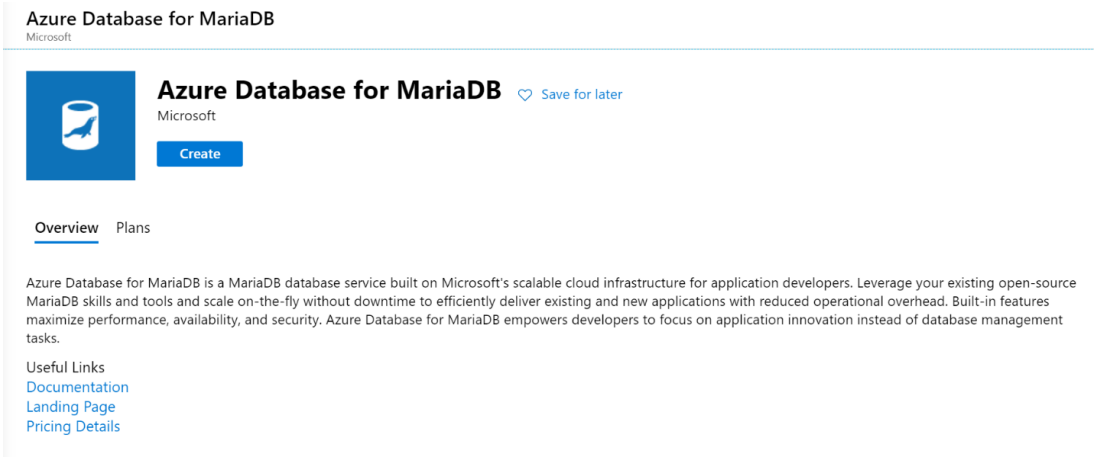


Figure 4: A new MariaDB instance in Azure.

- Select **Create**.
- Select your subscription and resource group.
- Select a server name and location.
- Select your target version, which is 10.2 or 10.3.
- Select your compute and storage.
- Enter an admin username and password.
- Select **Review + create**.

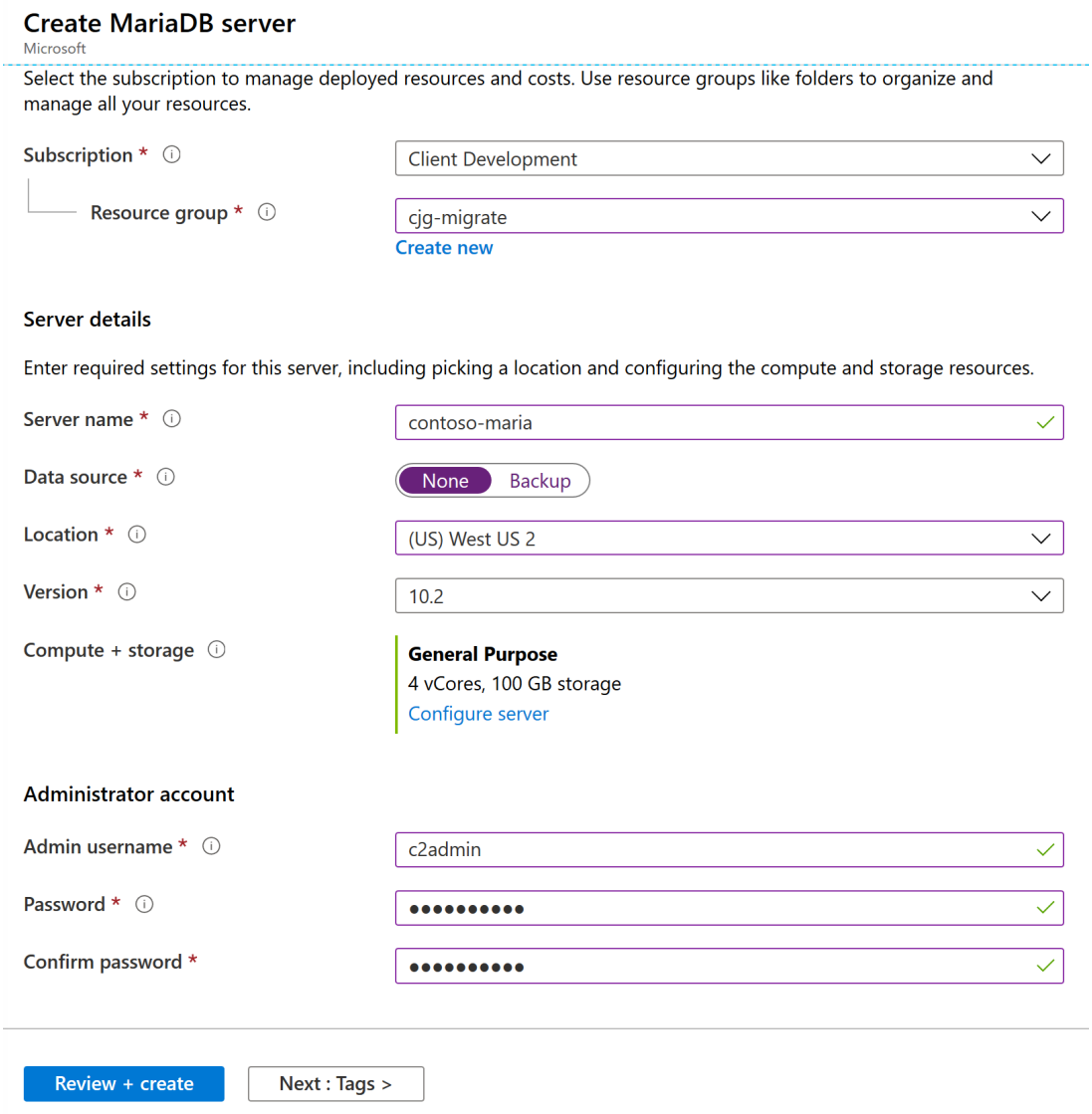


Figure 5: Review and create.

  - Select **Create**.
  - Record the server hostname, username, and password.
  - Select **Connection Security**.
  - Select **Add Client IP** (the IP that you'll be restoring the database from).
  - Select **Save**.

- Run the following commands to export the database called `Employees`. Repeat for each database:

```
mysqldump -h localhost -u root -p --skip-triggers --single-transaction --extended-insert --order-by-primary --disable-keys Employees > Employees.sql
```

- Restore the database. Replace with the endpoint for your Azure Database for MariaDB instance and the username:

```
mysql -h {name}.mariadb.database.azure.com -u user@{name} -p -ssl
create database employees;
use database employees;
source employees.sql;
```

- Use phpMyAdmin or a similar tool, such as MySQL Workbench, Toad, and Navicat, to verify the restore by checking record counts in each table.
- Update all application connection strings to point to the migrated database.
- Test all applications for proper operation.

Clean up after migration

After a verified successful migration, Contoso needs to back up and store the on-premises database backup files for retention purposes. Retire the on-premises MariaDB server.

Review the deployment

With the migrated resources in Azure, Contoso needs to fully operationalize and secure its new infrastructure.

Security

Contoso needs to:

- Ensure that its new Azure Database for MariaDB instance and databases are secure. For more information, see Security in Azure Database for MariaDB.
- Review the firewall rules and virtual network configurations to verify that connections are limited to only the applications that require it.
- Configure any outbound IP requirements to allow connections to the MariaDB gateway IP addresses.
- Update all applications to require SSL connections to the databases.
- Set up Private Link so that all database traffic is kept inside Azure and the on-premises network.
- Enable Microsoft Defender for Identity.
- Configure Log Analytics to monitor and send alerts on security and logs entries of interest.

Backups

Ensure that the Azure Database for MariaDB instances are backed up by using geo-restore. In this way, backups

can be used in a paired region if a regional outage occurs.

> **IMPORTANT**
> Make sure that the Azure Database for MariaDB instance has a resource lock to prevent it from being deleted. Deleted servers can't be restored.

Licensing and cost optimization

- Azure Database for MariaDB can be scaled up or down. Performance monitoring of the server and databases is important to ensure you meet your needs but also keep costs at a minimum.
- Both CPU and storage have costs associated. There are several pricing tiers to select from. Be sure the appropriate pricing plan is selected for the data workloads.
- Each read replica is billed based on the compute and storage selected.
- Use reserved capacity to save on costs.

Conclusion

In this article, Contoso migrated its MariaDB databases to an Azure Database for MariaDB instance.

Rehost an on-premises Linux application to Azure VMs

3/6/2021 • 12 minutes to read • Edit Online

This article shows how the fictional company Contoso rehosts a two-tier LAMP-based application by using Azure infrastructure as a service (IaaS) virtual machines (VMs).

The service desk application used in this example, osTicket, is provided as open-source software. If you want to use it for your own testing purposes, you can download it from GitHub.

Business drivers

The IT leadership team has worked closely with business partners to understand what they want to achieve with this migration:

- **Address business growth.** Contoso is growing, and as a result there's pressure on the on-premises systems and infrastructure.
- **Limit risk.** The service desk application is critical for the Contoso business. Contoso wants to move it to Azure with zero risk.
- **Extend.** Contoso doesn't want to change the application right now. It wants to ensure that the application is stable.

Migration goals

The Contoso cloud team has pinned down goals for this migration to determine the best migration method:

- After migration, the application in Azure should have the same performance capabilities as it does today in the company's on-premises VMware environment. The application will remain as critical in the cloud as it is on-premises.
- Contoso doesn't want to invest in this application. It's important to the business, but in its current form Contoso simply wants to move it safely to the cloud.
- Contoso doesn't want to change the ops model for this application. It wants to interact with the application in the cloud in the same way that it does now.
- Contoso doesn't want to change application functionality. Only the application location will change.
- Having completed a couple of Windows application migrations, Contoso wants to learn how to use a Linux-based infrastructure in Azure.

Solution design

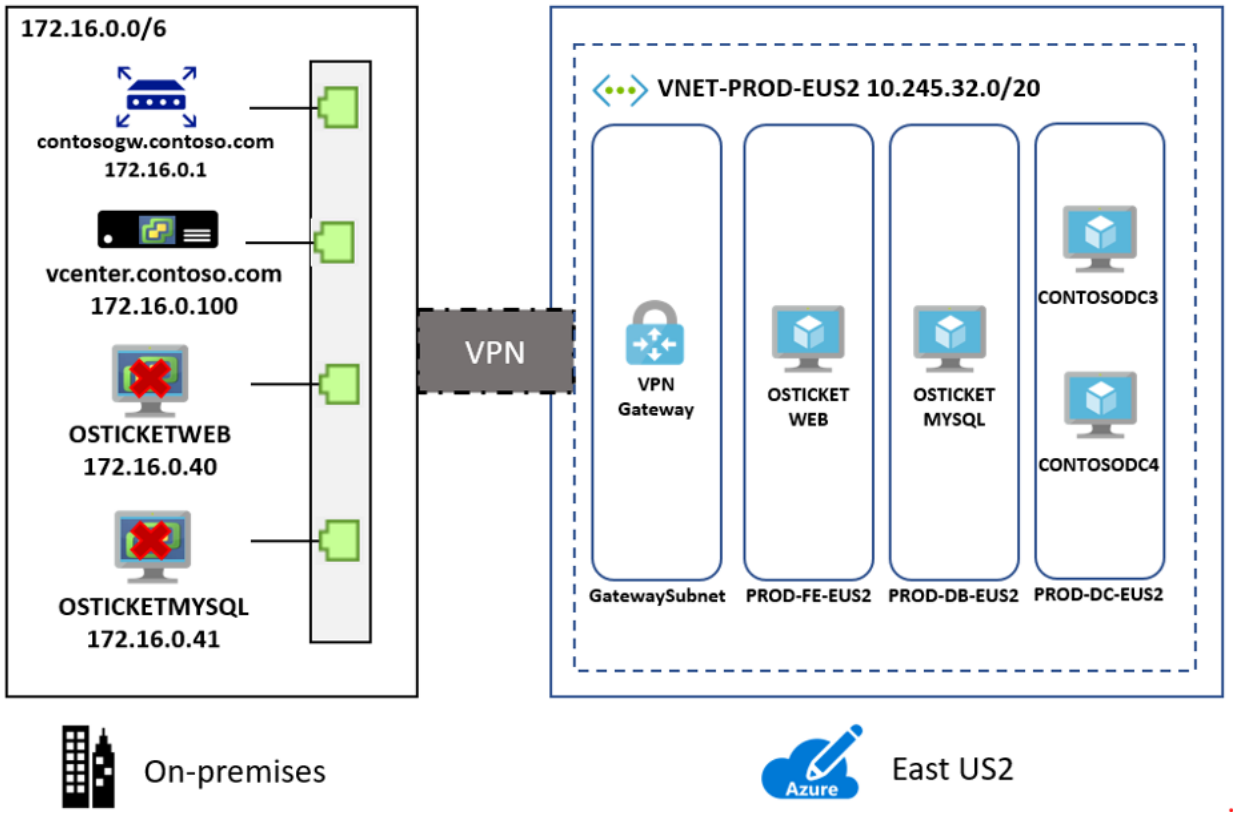
After pinning down goals and requirements, Contoso designs and reviews a deployment solution and identifies the migration process. The Azure services that Contoso will use for the migration also are identified.

Current application

- The osTicket application is tiered across two VMs (`OSTICKETWEB` and `OSTICKETMYSQL`).
- The VMs are located on VMware ESXi host `contosohost1.contoso.com` (version 6.5).
- The VMware environment is managed by vCenter Server 6.5 (`vcenter.contoso.com`) and runs on a VM.
- Contoso has an on-premises datacenter (`contoso-datacenter`) with an on-premises domain controller (`contosodc1`).

Proposed architecture

- Because the application is a production workload, the VMs in Azure will reside in the production resource group `ContosoRG`.
- The VMs will be migrated to the primary region (East US 2) and placed in the production network (`VNET-PROD-EUS2`):
  - The web VM will reside in the front-end subnet (`PROD-FE-EUS2`).
  - The database VM will reside in the database subnet (`PROD-DB-EUS2`).
- The on-premises VMs in the Contoso datacenter will be decommissioned after the migration is done.



Solution review

Contoso evaluates the proposed design by putting together a list of pros and cons.

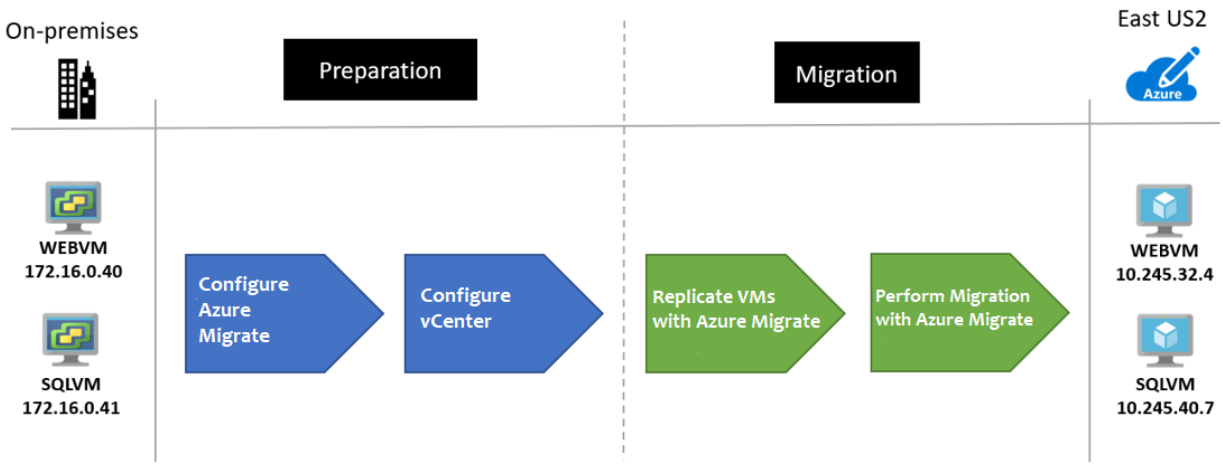
CONSIDERATION	DETAILS
Pros	Both the application VMs will be moved to Azure without changes, which makes the migration simple. Because Contoso is using a lift-and-shift approach for both application VMs, no special configuration or migration tools are needed for the application database. Contoso will retain full control of the application VMs in Azure. The application VMs are running Ubuntu 16.04-TLS, an endorsed Linux distribution. Learn more about endorsed Linux distributions on Azure.

CONSIDERATION	DETAILS
Cons	The web and data tier of the application remain single points of failover. Contoso will need to continue supporting the application as Azure VMs rather than moving to a managed service, such as Azure App Service and Azure Database for MySQL. Contoso realizes that by keeping things simple with a lift-and-shift VM migration, the company isn't taking full advantage of the features provided by Azure Database for MySQL. These features include built-in high availability, predictable performance, simple scaling, automatic backups, and built-in security.

Migration process

Contoso will complete the migration process as follows:

- As a first step, Contoso prepares and sets up Azure components for Azure Migrate: Server Migration and prepares the on-premises VMware infrastructure.
- The company already has the Azure infrastructure in place, so it just needs to configure the replication of the VMs through the Azure Migrate: Server Migration tool.
- With everything prepared, Contoso can start replicating the VMs.
- After replication is enabled and working, Contoso will migrate the VM by failing it over to Azure.



Azure services

SERVICE	DESCRIPTION	COST
Azure Migrate: Server Migration	The service orchestrates and manages migration of your on-premises applications and workloads as well as Amazon Web Services (AWS) and Google Cloud Platform (GCP) VM instances.	During replication to Azure, Azure Storage charges are incurred. Azure VMs are created, and incur charges, when migration occurs. Learn more about charges and pricing.

Prerequisites

Here's what Contoso needs for this scenario.

REQUIREMENTS	DETAILS
Azure subscription	Contoso created subscriptions in an earlier article in this series. If you don't have an Azure subscription, create a free account. If you create a free account, you're the administrator of your subscription and can perform all actions. If you use an existing subscription and you're not the administrator, work with the admin to assign you Owner or Contributor permissions. If you need more granular permissions, see Manage Site Recovery access with Azure role-based access control (Azure RBAC).
Azure infrastructure	Learn how Contoso set up an Azure infrastructure. Learn more about specific prerequisites for Azure Migrate: Server Migration.
On-premises servers	The on-premises vCenter Server should be running version 5.5, 6.0, or 6.5. An ESXi host running version 5.5, 6.0, or 6.5. One or more VMware VMs running on the ESXi host.
On-premises VMs	Review Linux distros that are endorsed to run on Azure.

Scenario steps

Here's how Contoso will complete the migration:

- **Step 1: Prepare Azure for Azure Migrate: Server Migration.** Add the Azure Migrate: Server Migration tool to the Azure Migrate project.
- **Step 2: Prepare on-premises VMware for Azure Migrate: Server Migration.** Prepare accounts for VM discovery, and prepare to connect to Azure VMs after migration.
- **Step 3: Replicate VMs.** Set up replication, and start replicating VMs to Azure Storage.
- **Step 4: Migrate the VMs with Azure Migrate: Server Migration.** Run a test migration to make sure everything's working, and then run a migration to move the VMs to Azure.

Step 1: Prepare Azure for the Azure Migrate: Server Migration tool

Here are the Azure components Contoso needs to migrate the VMs to Azure:

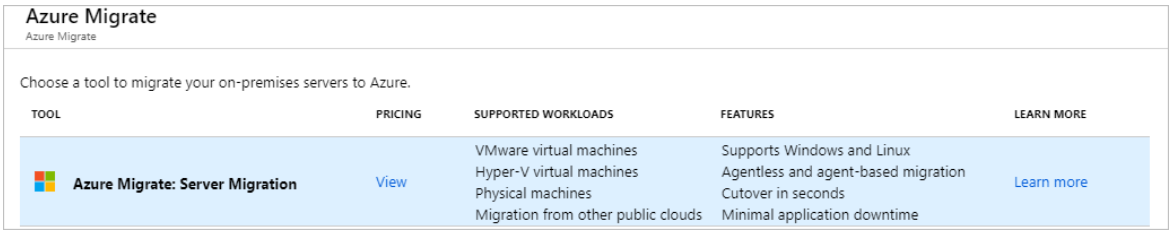
- A virtual network in which Azure VMs will be located when they're created during migration.
- The Azure Migrate: Server Migration tool provisioned.

They set up these components as follows:

1. Set up a network. Contoso already set up a network that can be used for Azure Migrate: Server Migration when the company deployed the Azure infrastructure
    - The SmartHotel360 application is a production application. The VMs will be migrated to the Azure production network (`VNET-PROD-EUS2`) in the primary region (`East US 2`).

- Both VMs will be placed in the `ContosoRG` resource group, which is used for production resources.
- The application front-end VM (`OSTICKETWEB`) will migrate to the front-end subnet (`PROD-FE-EUS2`) in the production network.
- The application database VM (`OSTICKETMYSQL`) will migrate to the database subnet (`PROD-DB-EUS2`) in the production network.

2. Provision the Azure Migrate: Server Migration tool. With the network and storage account in place, Contoso now creates a Recovery Services vault (`ContosoMigrationVault`) and places it in the `ContosoFailoverRG` resource group in the primary region (`East US 2`).



Need more help?

Learn about how to set up the Azure Migrate: Server Migration tool.

Step 2: Prepare on-premises VMware for Azure Migrate: Server Migration

After migration to Azure, Contoso wants to be able to connect to the replicated VMs in Azure. There are a couple of things that the Contoso admins need to do:

- To access Azure VMs over the internet, they enable SSH on the on-premises Linux VM before migration. For Ubuntu, this step can be completed by using the following command: `sudo apt-get ssh install -y`.
- Install the Azure Linux agent
- After they run the migration, they can check **Boot diagnostics** to view a screenshot of the VM.
- If it doesn't work, they'll need to check that the VM is running and review these troubleshooting tips.

Need more help?

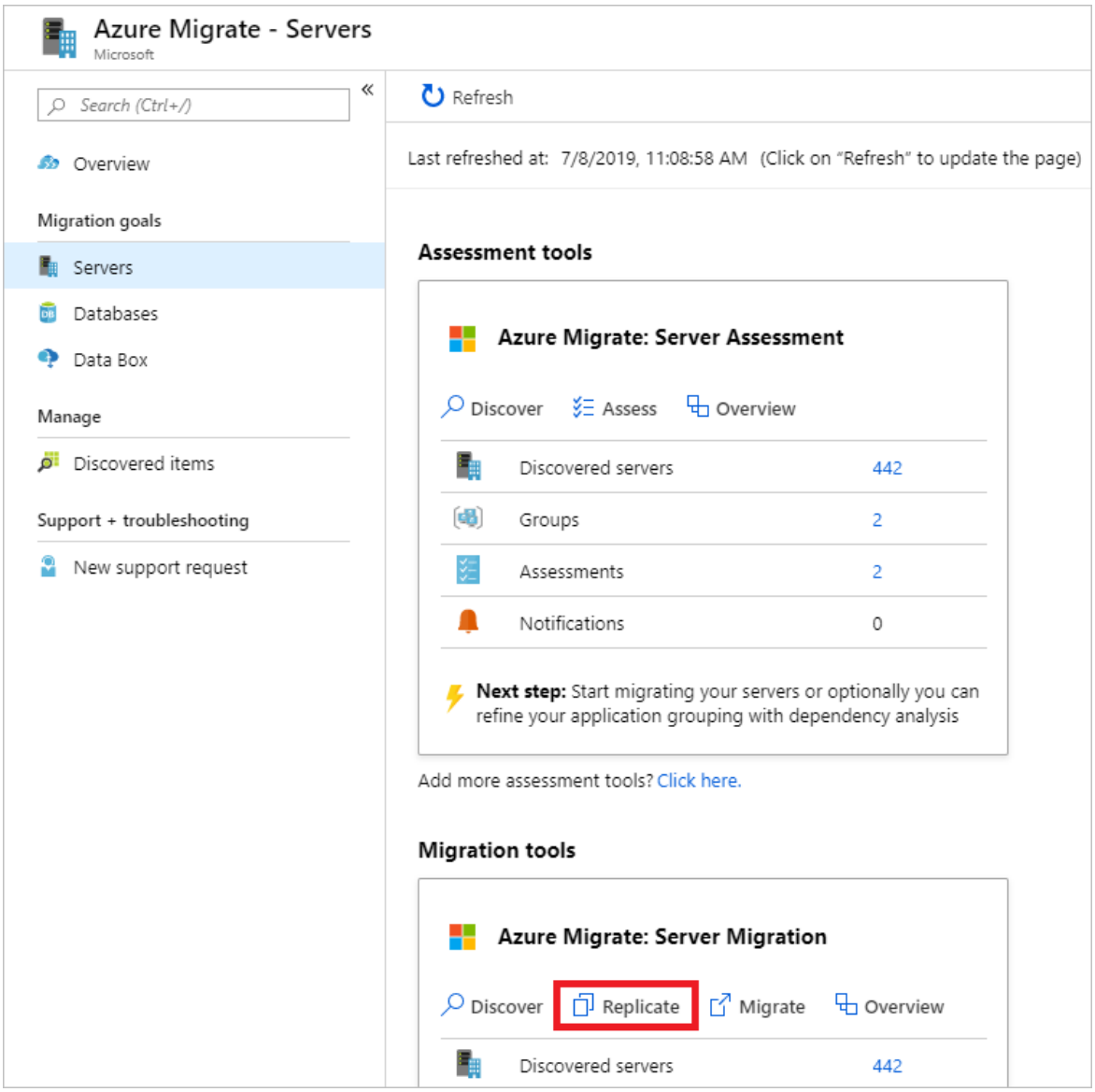
Learn about how to prepare VMs for migration.

Step 3: Replicate the on-premises VMs

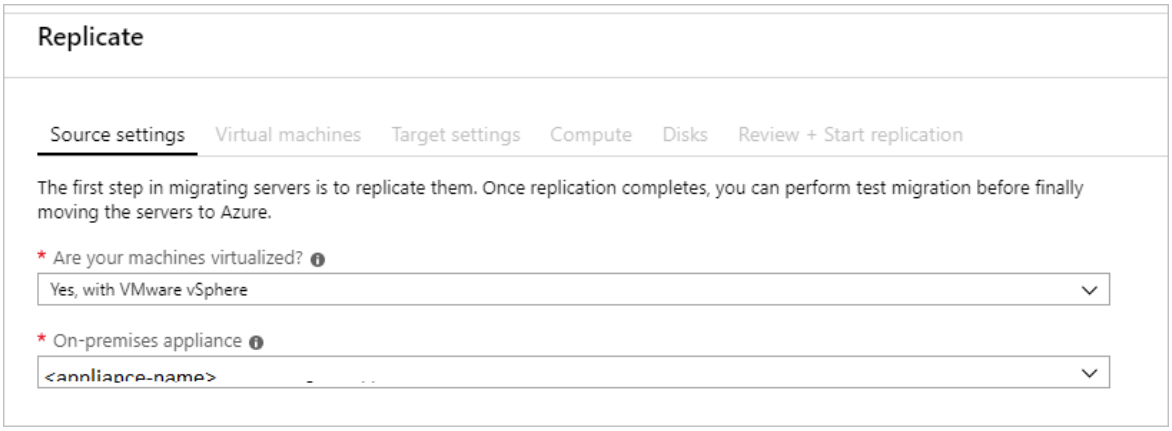
Before Contoso admins can run a migration to Azure, they need to set up and enable replication.

With discovery finished, begin replication of VMware VMs to Azure.

1. In the Azure Migrate project, go to **Servers** > **Azure Migrate: Server Migration**, and select **Replicate**.



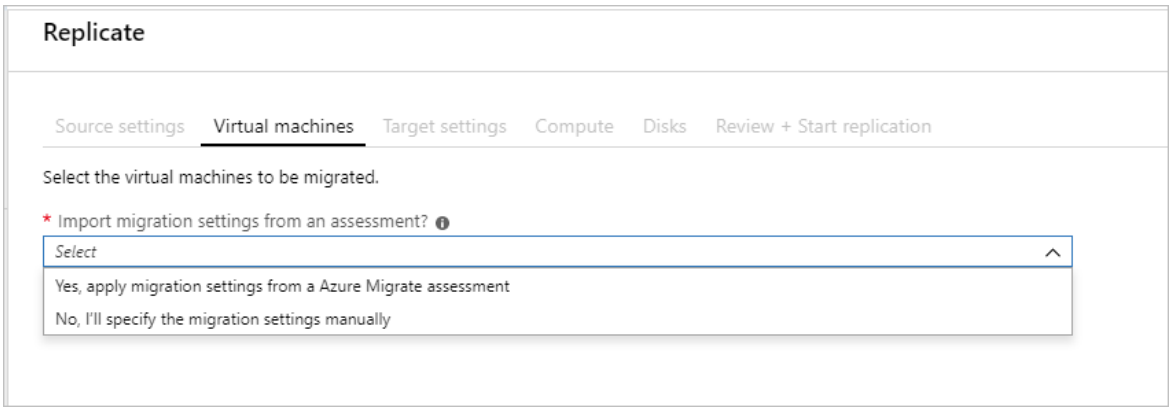
2. In **Replicate** > **Source settings** > **Are your machines virtualized?**, select **Yes, with VMware vSphere**.
3. In **On-premises appliance**, select the name of the Azure Migrate appliance that you set up, and then select **OK**.



4. In **Virtual machines**, select the machines you want to replicate.
   - If you've run an assessment for the VMs, you can apply VM sizing and disk type (premium/standard) recommendations from the assessment results. In **Import migration settings from an Azure Migrate assessment?**, select the **Yes** option.
   - If you didn't run an assessment, or you don't want to use the assessment settings, select the **No**

option.

- If you selected to use the assessment, select the VM group and assessment name.



5. In **Virtual machines**, search for VMs as needed, and select each VM you want to migrate. Then select **Next: Target settings**.
6. In **Target settings**, select the subscription and target region to which you'll migrate. Specify the resource group in which the Azure VMs will reside after migration. In **Virtual Network**, select the Azure virtual network/subnet to which the Azure VMs will be joined after migration.
7. In **Azure Hybrid Benefit**:
   - Select **No** if you don't want to apply Azure Hybrid Benefit. Then select **Next**.
   - Select **Yes** if you have Windows Server machines that are covered with active Software Assurance or Windows Server subscriptions and you want to apply the benefit to the machines you're migrating. Then select **Next**.
8. In **Compute**, review the VM name, size, OS disk type, and availability set. VMs must conform with Azure requirements.
   - **VM size:** If you're using assessment recommendations, the VM size drop-down list will contain the recommended size. Otherwise, Azure Migrate picks a size based on the closest match in the Azure subscription. Alternatively, pick a manual size in **Azure VM size**.
   - **OS disk:** Specify the OS (boot) disk for the VM. The OS disk is the disk that has the operating system bootloader and installer.
   - **Availability set:** If the VM should be in an Azure availability set after migration, specify the set. The set must be in the target resource group you specify for the migration.
9. In **Disks**, specify whether the VM disks should be replicated to Azure. Select the disk type (standard SSD/HDD or premium-managed disks) in Azure. Then select **Next**.
   - You can exclude disks from replication.
   - If you exclude disks, they won't be present on the Azure VM after migration.
10. In **Review + Start replication**, review the settings. Then select **Replicate** to start the initial replication for the servers.

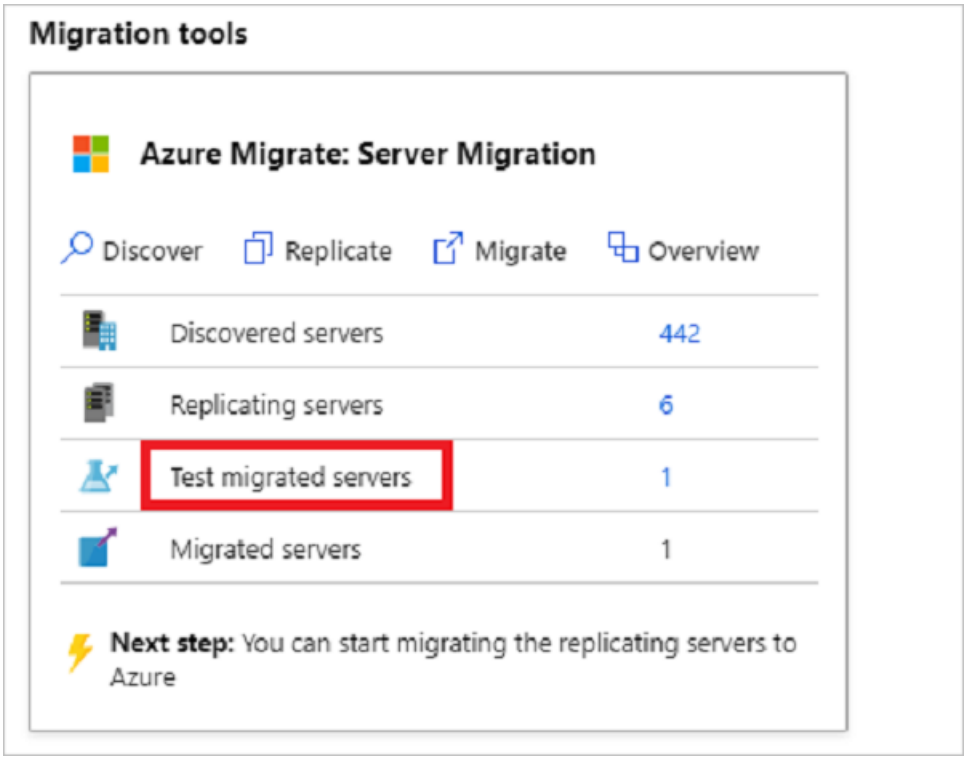
> **NOTE**
>
> You can update replication settings any time before replication starts in **Manage** > **Replicating machines**. Settings can't be changed after replication starts.

Step 4: Migrate the VMs

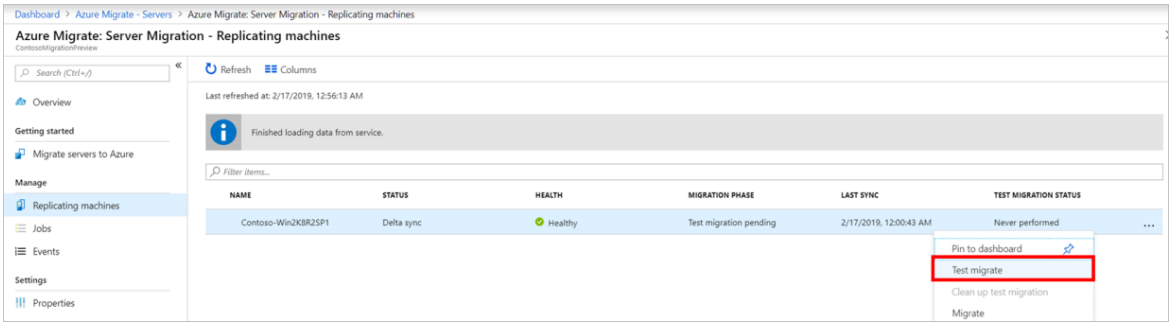
Contoso admins run a quick test migration and then a migration to move the VMs.

Run a test migration

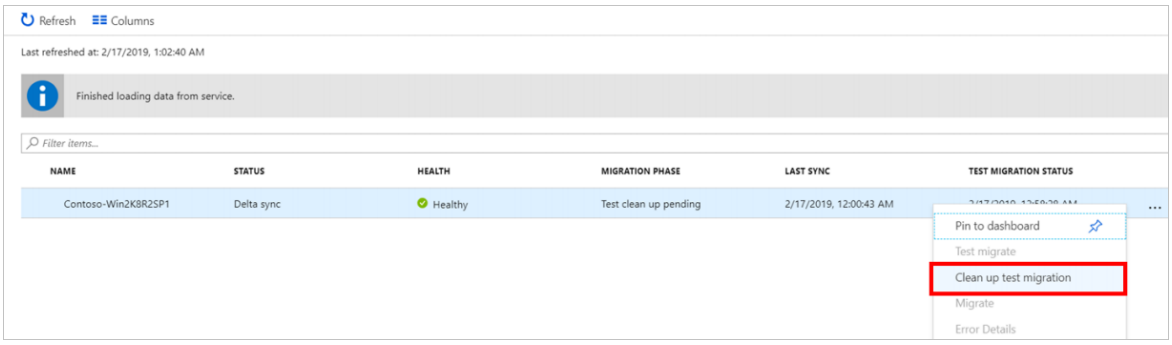
1. In **Migration goals** > **Servers** > **Azure Migrate: Server Migration**, select **Test migrated servers**.



2. Select and hold (or right-click) the VM to test. Then select **Test migrate**.



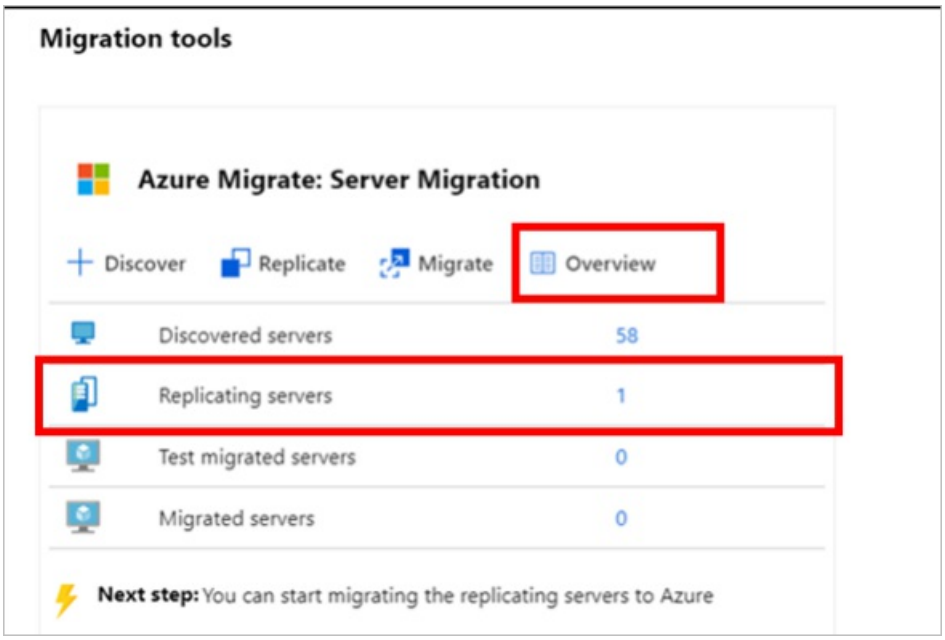
3. In **Test Migration**, select the Azure virtual network in which the Azure VM will be located after the migration. We recommend you use a nonproduction virtual network.
4. The **Test migration** job starts. Monitor the job in the portal notifications.
5. After the migration finishes, view the migrated Azure VM in **Virtual Machines** in the Azure portal. The machine name has a suffix **-Test**.
6. After the test is done, select and hold (or right-click) the Azure VM in **Replicating machines**. Then select **Clean up test migration**.



Migrate the VMs

Now Contoso admins run a full migration to complete the move.

1. In the Azure Migrate project, go to **Servers** > **Azure Migrate: Server Migration**, and select **Replicating servers**.

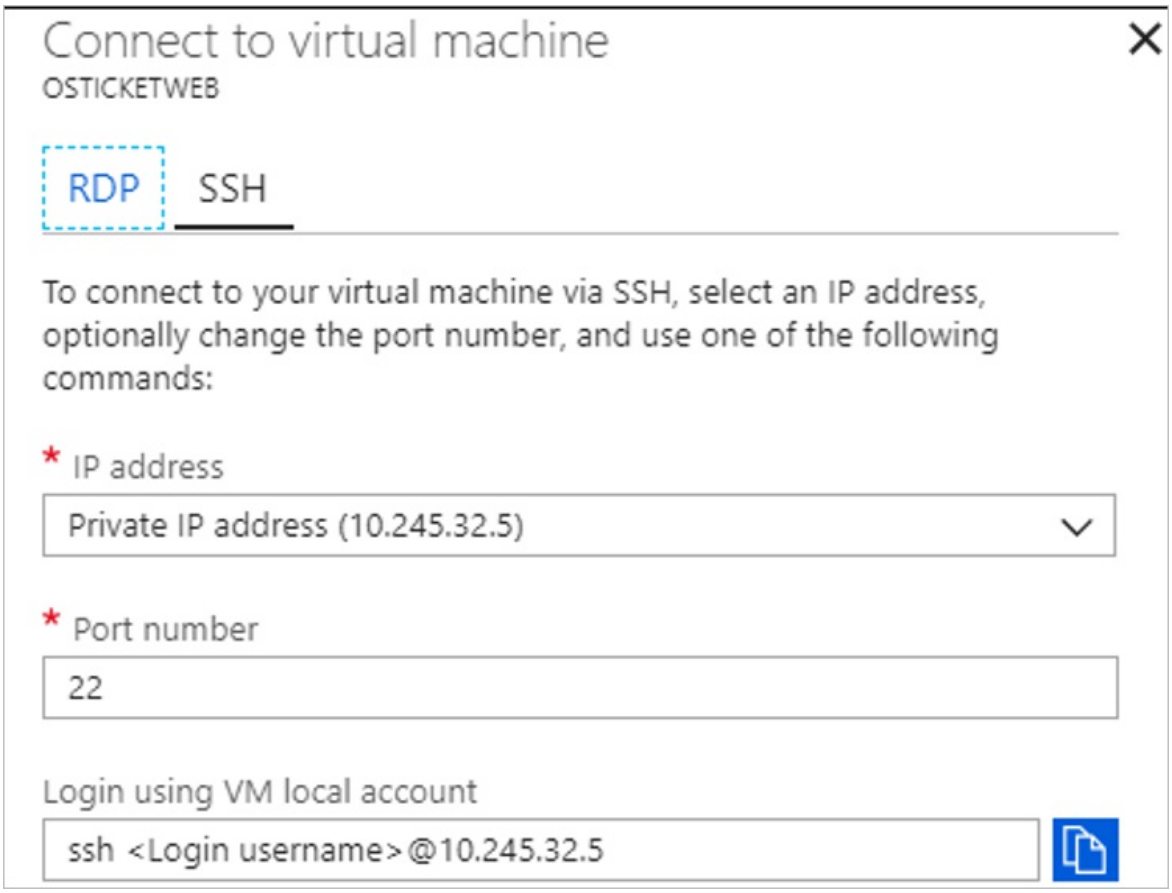


2. In **Replicating machines**, select and hold (or right-click) the VM and select **Migrate**.
3. In **Migrate** > **Shut down virtual machines and perform a planned migration with no data loss**, select **Yes** > **OK**.
   - By default, Azure Migrate shuts down the on-premises VM and runs an on-demand replication to synchronize any VM changes that occurred since the last replication occurred. This action ensures no data loss.
   - If you don't want to shut down the VM, select **No**.
4. A migration job starts for the VM. Track the job in Azure notifications.
5. After the job finishes, you can view and manage the VM from the **Virtual Machines** page.

Connect the VM to the database

As the final step in the migration process, Contoso admins update the connection string of the application to point to the application database running on the `OSTICKETMYSQL` VM.

1. Make an SSH connection to the `OSTICKETWEB` VM by using PuTTY or another SSH client. The VM is private, so connect by using the private IP address.




```
$ ssh contosoadmin@10.245.32.5
The authenticity of host '10.245.32.5 (10.245.32.5)' can't be established.
ECDSA key fingerprint is SHA256:aJNt+VbYqoZb+mrBlCCj5[redacted].
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '10.245.32.5' (ECDSA) to the list of known hosts.
contosoadmin@10.245.32.5's password:
Welcome to Ubuntu 16.04.3 LTS (GNU/Linux 4.4.0-127-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

136 packages can be updated.
59 updates are security updates.


Last login: [redacted] from 172.16.0.200
contosoadmin@osticketweb:~$
```

2. Make sure that the `OSTICKETWEB` VM can communicate with the `OSTICKETMYSQL` VM. Currently, the configuration is hardcoded with the on-premises IP address `172.16.0.43`.

   Before the update:

```
# Database Options
# ------------------------------
# Mysql Login info
define('DBTYPE','mysql');
define('DBHOST','172.16.0.43');
define('DBNAME','osticket');
define('DBUSER','osticket');
```

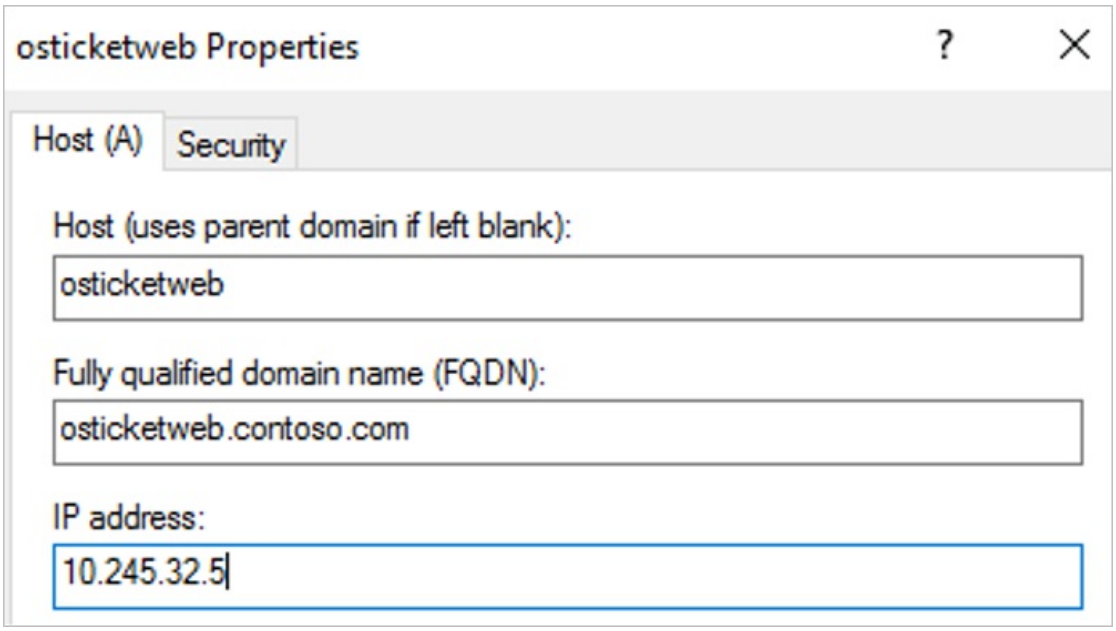
After the update:

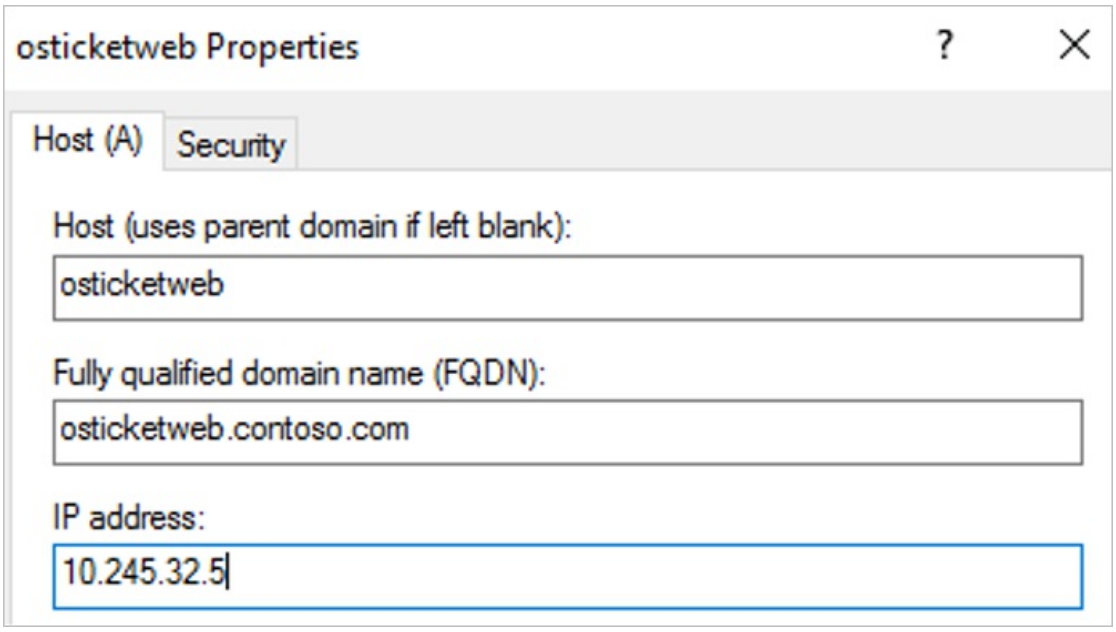
```
# Database Options
# ----------------------------------------
# Mysql Login info
define('DBTYPE','mysql');
define('DBHOST','osticketmysql.contoso.com');
define('DBNAME','osticket');
define('DBUSER','osticket');
-- INSERT --
```

3. Restart the service with **systemctl restart apache2**.

```
contosoadmin@osticketweb:~$ systemctl restart apache2
==== AUTHENTICATING FOR org.freedesktop.systemd1.manage-units
Authentication is required to restart 'apache2.service'.
Authenticating as: contosoadmin,,, (contosoadmin)
Password:
==== AUTHENTICATION COMPLETE ===
contosoadmin@osticketweb:~$
```

4. Finally, update the DNS records for `OSTICKETWEB` and `OSTICKETMYSQL` on one of the Contoso domain controllers.





Need more help?

- Learn about how to run a test migration.
- Learn about how to migrate VMs to Azure.

Clean up after migration

With migration complete, the osTicket application tiers are now running on Azure VMs.

Now, Contoso needs to do the following tasks:

- Remove the on-premises VMs from the vCenter inventory.
- Remove the on-premises VMs from local backup jobs.
- Update the internal documentation to show the new location and IP addresses for `OSTICKETWEB` and `OSTICKETMYSQL`.
- Review any resources that interact with the VMs. Update any relevant settings or documentation to reflect the new configuration.
- Contoso used the Azure Migrate service with management VM to assess the VMs for migration. Admins should remove the migration VM and web VMs from VMware ESXi server.

Review the deployment

With the application now running, Contoso needs to fully operationalize and secure its new infrastructure.

Security

The Contoso security team reviews the `OSTICKETWEB` and `OSTICKETMYSQL` VMs to determine any security issues.

- The team reviews the network security groups (NSGs) for the VMs to control access. NSGs are used to ensure that only traffic allowed to the application can pass.
- The team also considers securing the data on the VM disks by using Azure Disk Encryption and Azure Key Vault.

For more information, see Security best practices for IaaS workloads in Azure.

Business continuity and disaster recovery

For business continuity and disaster recovery, Contoso takes the following actions:

- **Keep data safe.** Contoso backs up the data on the VMs by using Azure VM backup.

- **Keep applications up and running.** Contoso replicates the application VMs in Azure to a secondary region by using Site Recovery. For more information, see Quickstart: Set up disaster recovery to a secondary Azure region for an Azure VM.

Licensing and cost optimization

- After deploying resources, Contoso assigns Azure tags as defined during the Azure infrastructure deployment.
- Contoso has no licensing issues with the Ubuntu servers.
- Contoso will use Azure Cost Management + Billing to ensure the company stays within budgets established by the IT leadership.

Rehost an on-premises Linux application to Azure VMs and Azure Database for MySQL

3/6/2021 • 15 minutes to read • Edit Online

This article shows how the fictional company Contoso rehosts a two-tier LAMP-based application and migrates it from on-premises to Azure by using Azure Virtual Machines (VMs) and Azure Database for MySQL.

The service desk application used in this example, osTicket, is provided as open-source software. If you want to use it for your own testing, you can download it from GitHub.

Business drivers

The IT leadership team has worked closely with business partners to understand what they want to achieve:

- **Address business growth.** Contoso is growing, and as a result there's pressure on the on-premises systems and infrastructure.
- **Limit risk.** The service desk application is critical for the business. Contoso wants to move it to Azure with zero risk.
- **Extend.** Contoso doesn't want to change the application right now. The company wants to keep the application stable.

Migration goals

The Contoso cloud team has pinned down goals for this migration to determine the best migration method:

- After migration, the application in Azure should have the same performance capabilities as it does today in the company's on-premises VMware environment. The application will remain as critical in the cloud as it is on-premises.
- Contoso doesn't want to invest in this application. It's important to the business, but in its current form Contoso simply wants to move it safely to the cloud.
- Having completed a couple of Windows application migrations, Contoso wants to learn how to use a Linux-based infrastructure in Azure.
- Contoso wants to minimize database admin tasks after the application is moved to the cloud.

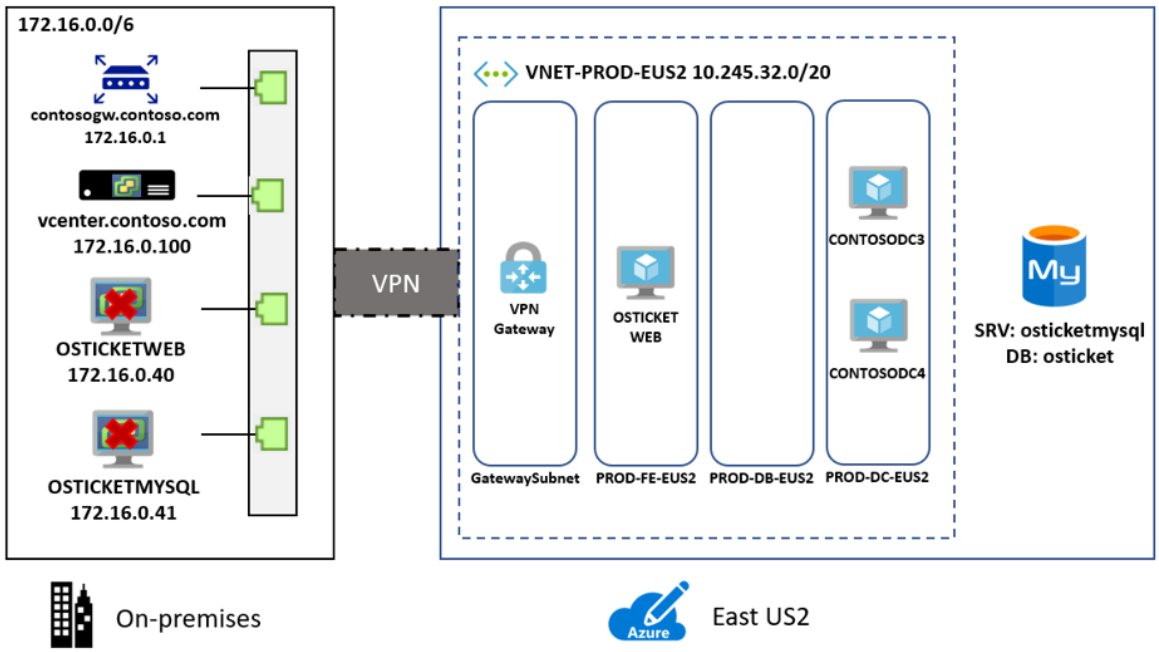
Proposed architecture

In this scenario:

- Currently the application is tiered across two VMs (`OSTICKETWEB` and `OSTICKETMYSQL`).
- The VMs are located on VMware ESXi host `contosohost1.contoso.com` (version 6.5).
- The VMware environment is managed by vCenter Server 6.5 (`vcenter.contoso.com`) and runs on a VM.
- Contoso has an on-premises datacenter (`contoso-datacenter`), with an on-premises domain controller (`contosodc1`).
- The web application on `OSTICKETWEB` will be migrated to an Azure infrastructure as a service (IaaS) VM.
- The application database will be migrated to the Azure Database for MySQL platform as a service.
- Because Contoso is migrating a production workload, the resources will reside in the production resource

group `ContosoRG`.

- The `OSTICKETWEB` resource will be replicated to the primary region (East US 2) and placed in the production network (`VNET-PROD-EUS2`):
  - The web VM will reside in the front-end subnet (`PROD-FE-EUS2`).
- The application database will be migrated to Azure Database for MySQL by using Azure Database Migration Service.
- The on-premises VMs in the Contoso datacenter will be decommissioned after the migration is done.



Migration process

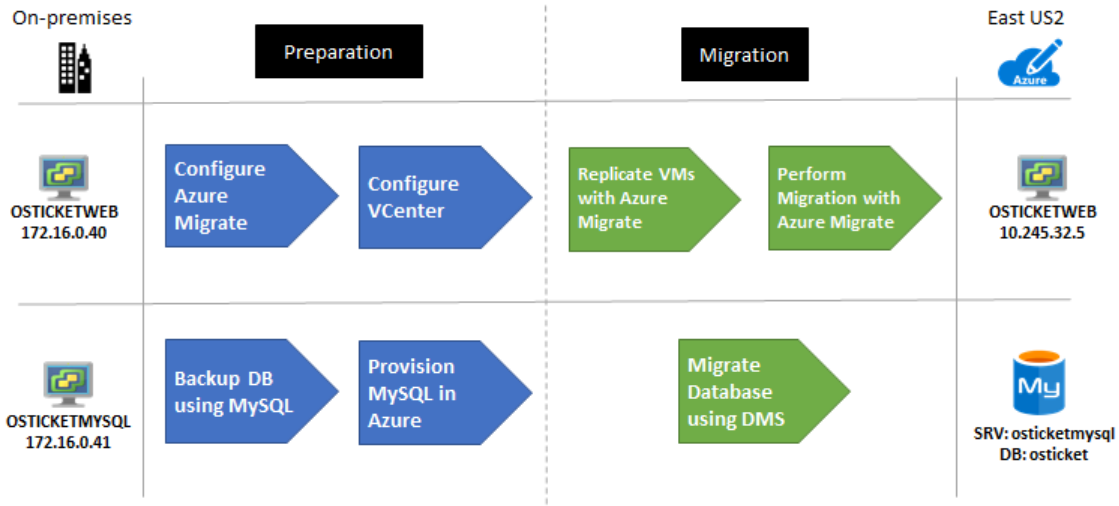
Contoso will complete the migration process as follows:

To migrate the web VM:

- As a first step, Contoso sets up the Azure and on-premises infrastructure needed to deploy Azure Migrate.
- The company already has the Azure infrastructure in place, so it just needs to add and configure the replication of the VMs through the Azure Migrate: Server Migration tool.
- With everything prepared, Contoso can start replicating the VM.
- After replication is enabled and working, Contoso will complete the move by using Azure Migrate.

To migrate the database:

1. Contoso provisions a MySQL instance in Azure.
2. Contoso sets up Database Migration Service, ensuring access to the on-premises database server.
3. Contoso migrates the database to Azure Database for MySQL.



Azure services

SERVICE	DESCRIPTION	COST
Azure Migrate	Contoso uses Azure Migrate to assess its VMware VMs. Azure Migrate assesses the migration suitability of the machines. It provides sizing and cost estimates for running in Azure.	Azure Migrate is available at no additional charge. You might incur charges depending on the tools (first-party or ISV) you decide to use for assessment and migration.
Azure Database Migration Service	Database Migration Service enables seamless migration from multiple database sources to Azure data platforms with minimal downtime.	Learn about supported regions and Database Migration Service pricing.
Azure Database for MySQL	The database is based on the open-source MySQL database engine. It provides a fully managed enterprise-ready community MySQL database for application development and deployment.	Learn more about Azure Database for MySQL pricing and scalability options.

Prerequisites

Here's what Contoso needs for this scenario.

REQUIREMENTS	DETAILS
Azure subscription	Contoso created subscriptions during an earlier article. If you don't have an Azure subscription, create a free account. If you create a free account, you're the administrator of your subscription and can perform all actions. If you use an existing subscription and you're not the administrator, work with the admin to assign you Owner or Contributor permissions. If you need more granular permissions, see Manage Azure Site Recovery access with role-based access control (RBAC).

REQUIREMENTS	DETAILS
Azure infrastructure	Contoso set up the Azure infrastructure as described in Azure infrastructure for migration.
On-premises servers	The on-premises vCenter Server should be running version 5.5, 6.0, 6.5, or 6.7. An ESXi host running version 5.5, 6.0, 6.5, or 6.7. One or more VMware VMs running on the ESXi host.
On-premises VMs	Review Linux machines that are endorsed to run on Azure.

Scenario steps

Here's how Contoso admins will complete the migration:

- **Step 1: Prepare Azure for Azure Migrate: Server Migration.** Add the server migration tool to the Azure Migrate project.
- **Step 2: Prepare on-premises VMware for Azure Migrate: Server Migration.** Prepare accounts for VM discovery and prepare to connect to Azure Virtual Machines after migrated.
- **Step 3: Replicate VMs.** Set up replication and start replicating VMs to Azure Storage.
- **Step 4: Migrate the application VM with Azure Migrate: Server Migration.** Run a test migration to make sure everything's working, and then run a full migration to move the VM to Azure.
- **Step 5: Migrate the database.** Set up migration by using Azure Database Migration Service.

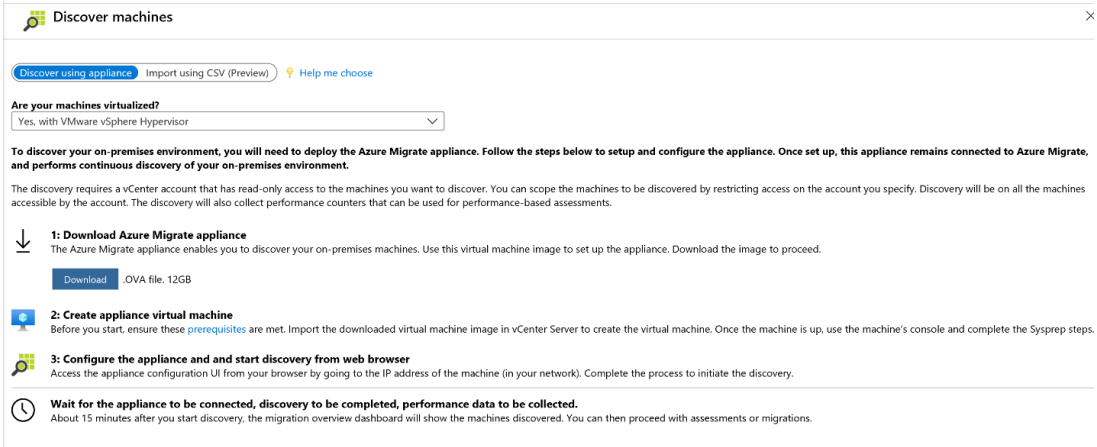
Step 1: Prepare Azure for the Azure Migrate: Server Migration tool

Here are the Azure components Contoso needs to migrate the VMs to Azure:

- A virtual network in which Azure VMs will be located when they're created during migration.
- The Azure Migrate: Server Migration tool (OVA) provisioned and configured.

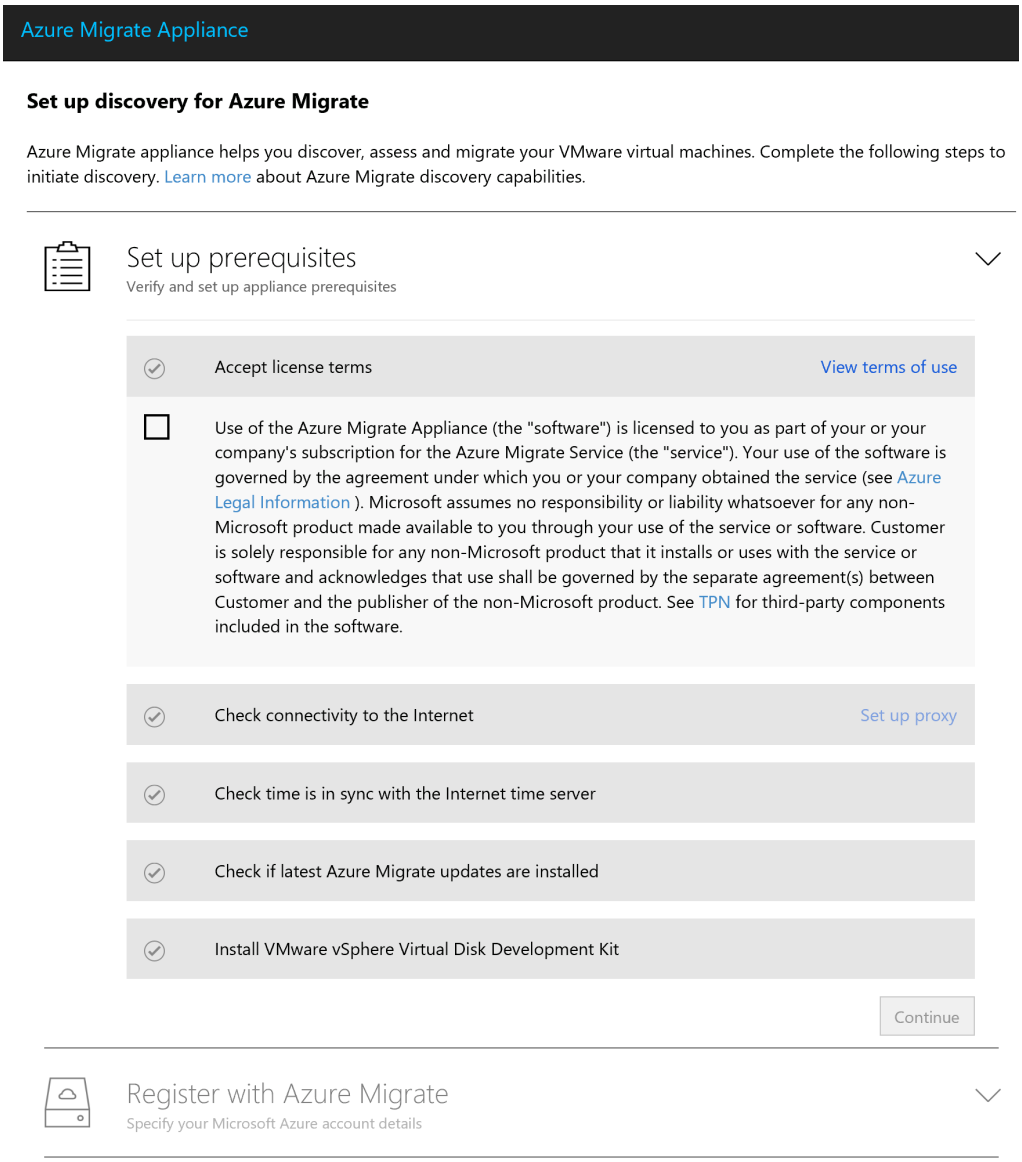
To set up the components, Contoso admins follow these steps:

1. Set up a network. Contoso already set up a network that can be used for Azure Migrate: Server Migration when it deployed the Azure infrastructure.
2. Provision the Azure Migrate: Server Migration tool.
    a. From Azure Migrate, download the OVA image, and import it into VMware.

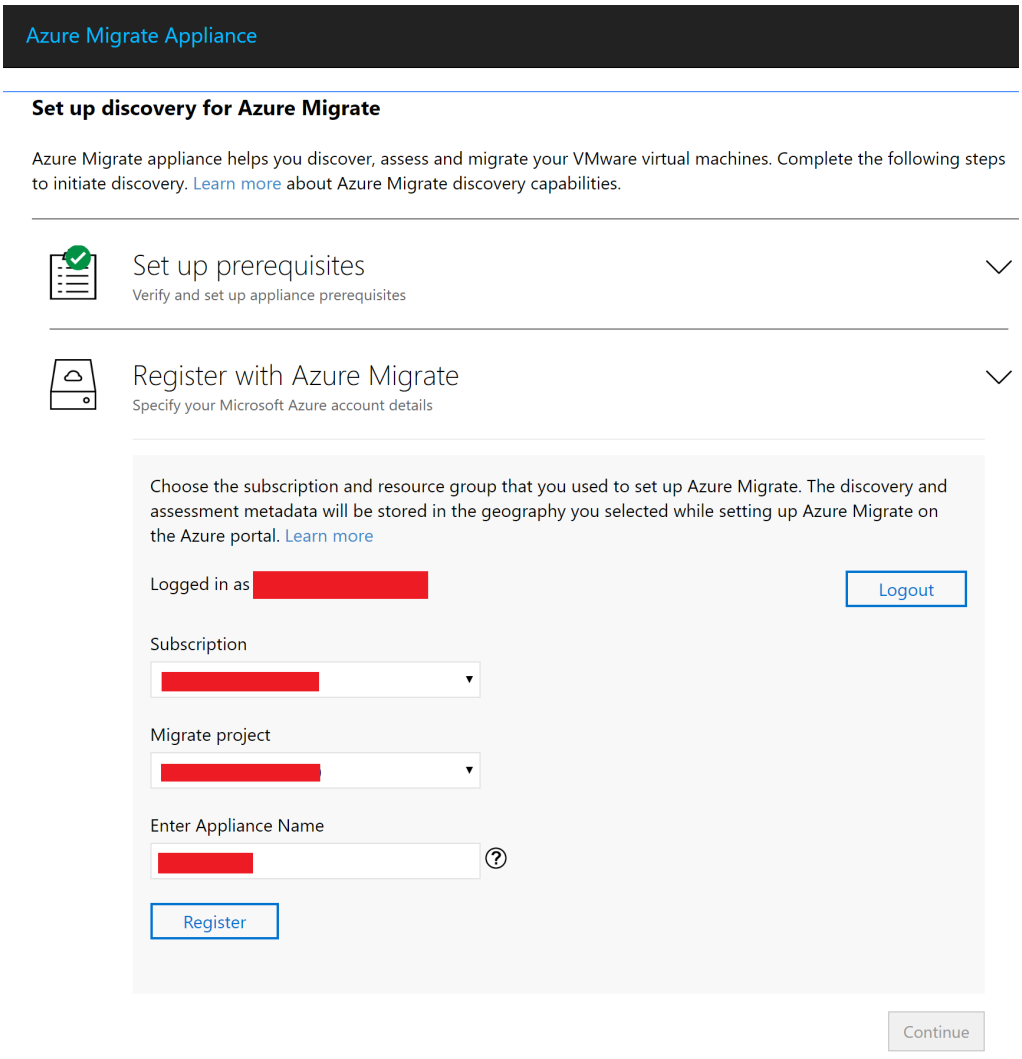


b. Start the imported image, and configure the tool by using the following steps:

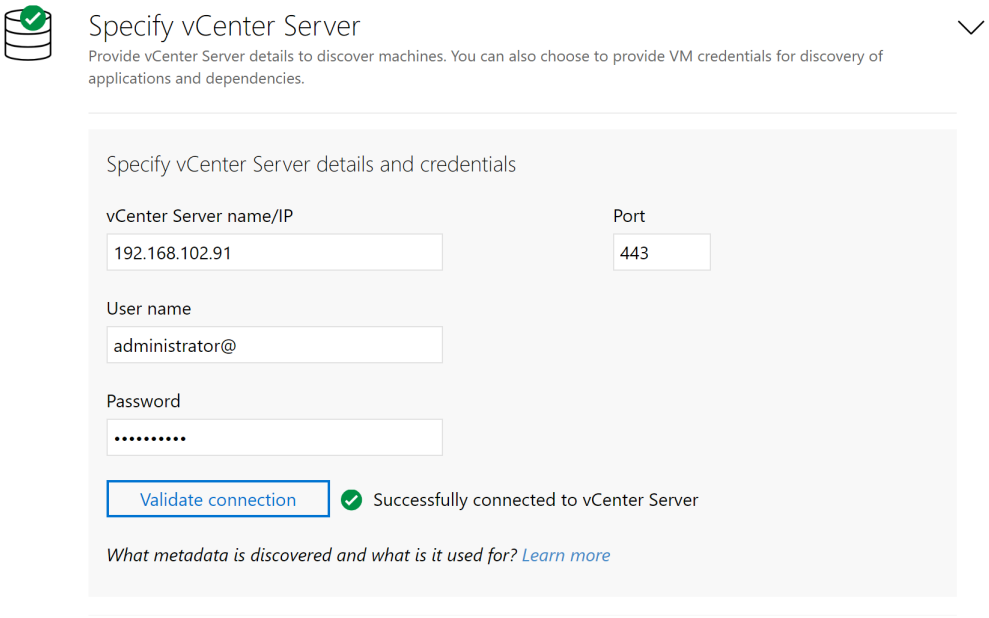
a. Set up the prerequisites.



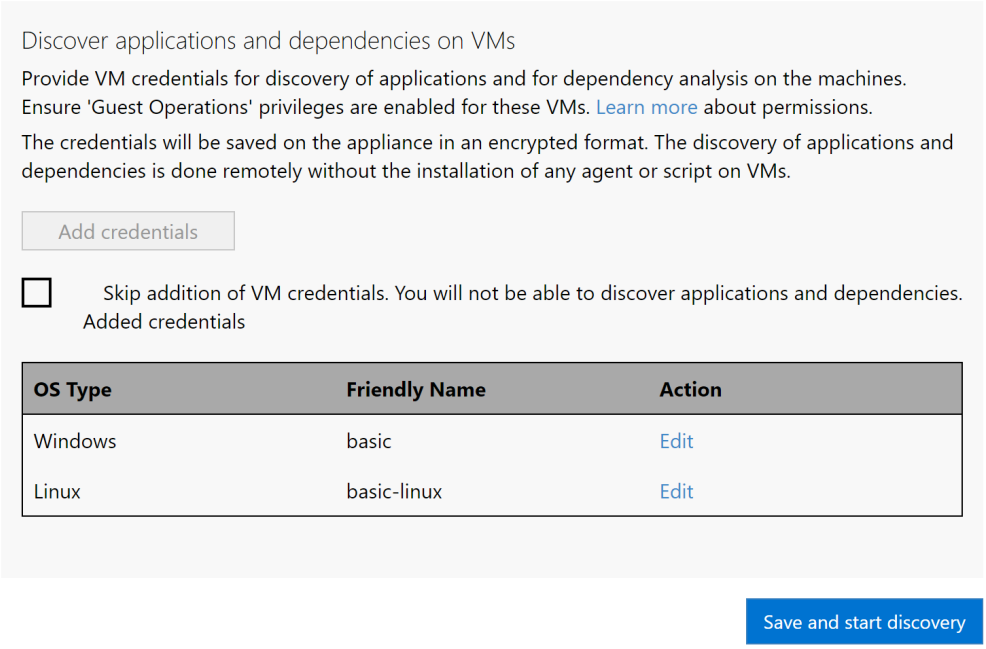
b. Point the tool to the Azure subscription.



c. Set the VMware vCenter credentials.



d. Add any Linux-based credentials for discovery.



3. After the tool is configured, it takes some time for the tool to enumerate all the virtual machines. After the process is finished, the VMs populate in the Azure Migrate tool in Azure.

Need more help?

Learn about how to set up the Azure Migrate: Server Migration tool.

Step 2: Prepare on-premises VMware for Azure Migrate: Server Migration

After migrating to Azure, Contoso wants to be able to connect to the replicated VMs in Azure. There are a couple of things that the Contoso admins need to do:

- To access Azure VMs, they enable SSH on the on-premises Linux VM before migration. For Ubuntu, this step can be completed by using the following command: `sudo apt-get ssh install -y`.
- After the admins run the migration, they can check **boot diagnostics** to view a screenshot of the VM.
- If it doesn't work, they'll need to check that the VM is running, and review these troubleshooting tips.
- Install the Azure Linux agent.

Need more help?

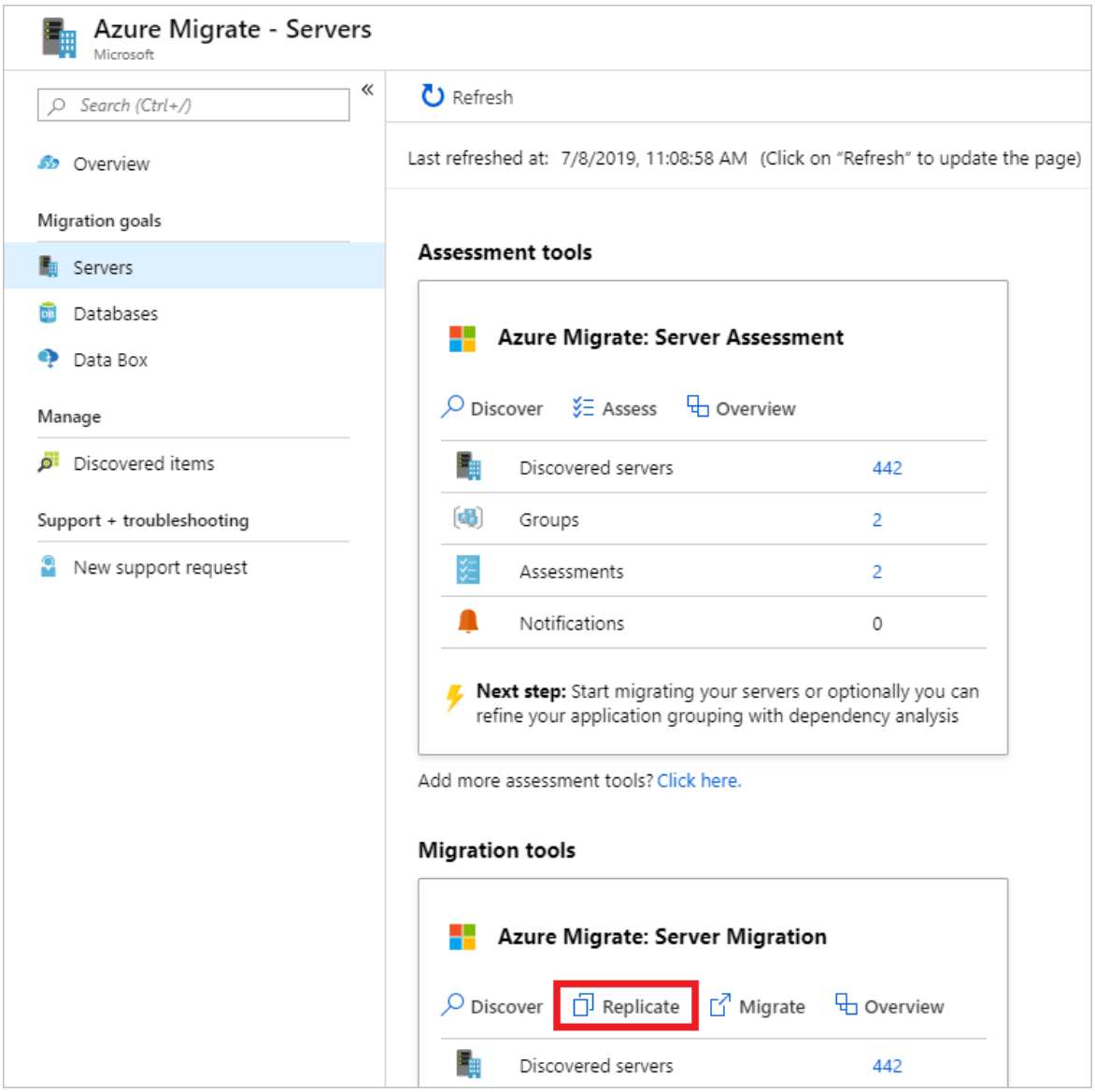
Learn about how to prepare VMs for migration.

Step 3: Replicate VMs

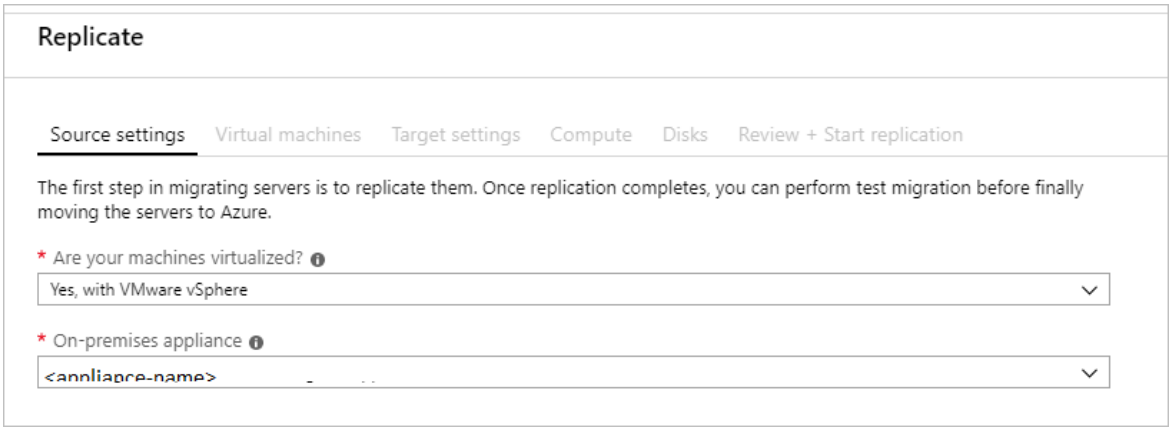
Before Contoso admins can run a migration to Azure, they need to set up and enable replication.

With discovery finished, they can begin replication of the application VM to Azure.

1. In the Azure Migrate project, go to **Servers** > **Azure Migrate: Server Migration**, and select **Replicate**.



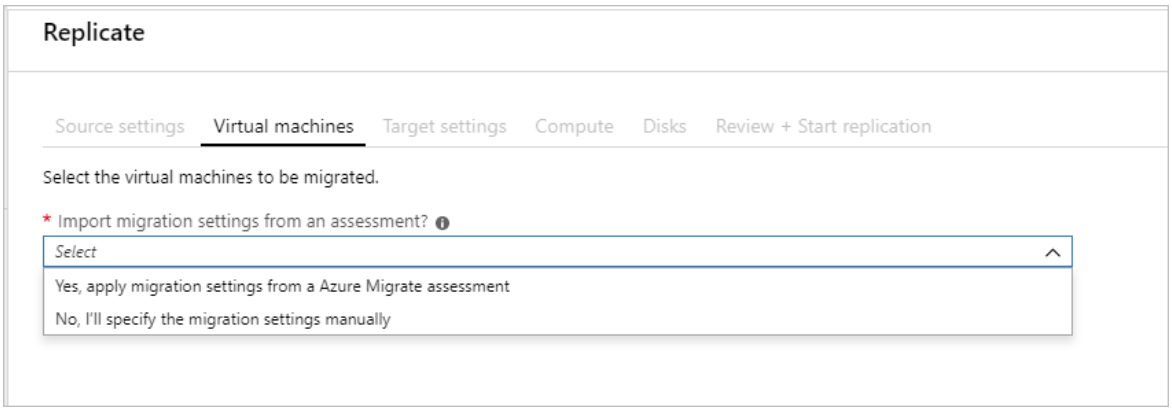
2. In **Replicate** > **Source settings** > **Are your machines virtualized?**, select **Yes, with VMware vSphere**.
3. In **On-premises appliance**, select the name of the Azure Migrate appliance that you set up, and then select **OK**.



4. In **Virtual machines**, select the machines you want to replicate:
   - If you've run an assessment for the VMs, you can apply VM sizing and disk type (premium/standard) recommendations from the assessment results. In **Import migration settings from an Azure Migrate assessment?**, select the **Yes** option.
   - If you didn't run an assessment, or you don't want to use the assessment settings, select the **No**

option.

- If you selected to use the assessment, select the VM group and assessment name.



5. In **Virtual machines**, search for VMs as needed, and select each VM you want to migrate. Then select **Next: Target settings**.
6. In **Target settings**, select the subscription and target region to which you'll migrate. Specify the resource group in which the Azure VMs will reside after migration. In **Virtual Network**, select the Azure virtual network/subnet to which the Azure VMs will be joined after migration.
7. In **Azure Hybrid Benefit**:
   - Select **No** if you don't want to apply Azure Hybrid Benefit. Then select **Next**.
8. In **Compute**, review the VM name, size, OS disk type, and availability set. VMs must conform with Azure requirements.
   - **VM size:** If you use assessment recommendations, the VM size drop-down list contains the recommended size. Otherwise, Azure Migrate picks a size based on the closest match in the Azure subscription. Alternatively, pick a manual size in **Azure VM size**.
   - **OS disk:** Specify the OS (boot) disk for the VM. The OS disk is the disk that has the operating system bootloader and installer.
   - **Availability set:** If the VM should be in an Azure availability set after migration, specify the set. The set must be in the target resource group you specify for the migration.
9. In **Disks**, specify whether the VM disks should be replicated to Azure. Then select the disk type (standard SSD/HDD or premium-managed disks) in Azure, and select **Next**.
   - You can exclude disks from replication.
   - If you exclude disks, they won't be present on the Azure VM after migration.
10. In **Review + Start replication**, review the settings. Then select **Replicate** to start the initial replication for the servers.

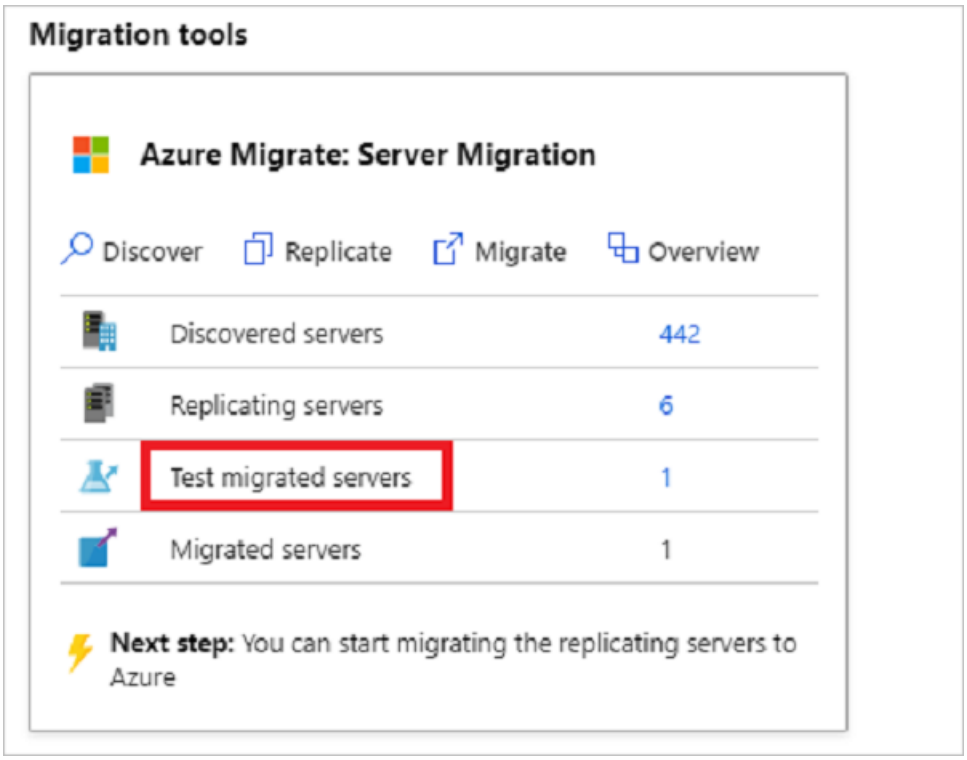
> **NOTE**
> You can update replication settings any time before replication starts in **Manage** > **Replicating machines**. Settings can't be changed after replication starts.

Step 4: Migrate the VM with Azure Migrate: Server Migration

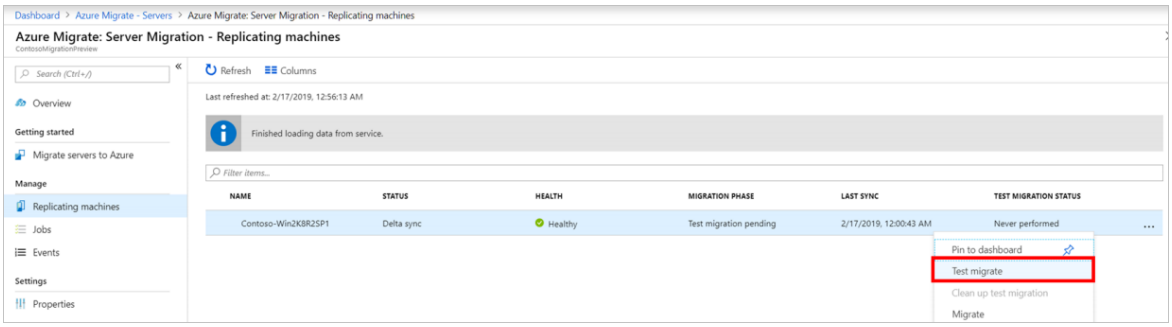
Contoso admins run a quick test migration and then a full migration to move the web VM.

Run a test migration

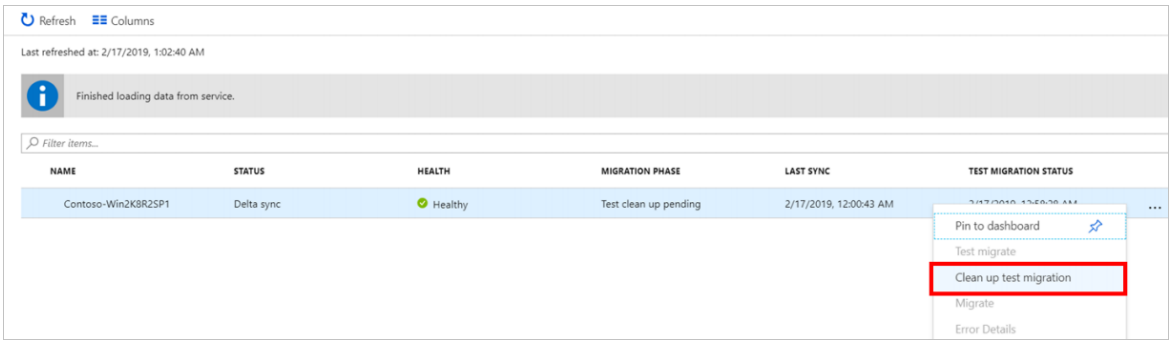
1. In **Migration goals** > **Servers** > **Azure Migrate: Server Migration**, select **Test migrated servers**.



2. Select and hold (or right-click) the VM to test, and then select **Test migrate**.



3. In **Test Migration**, select the Azure virtual network in which the Azure VM will be located after the migration. We recommend you use a nonproduction virtual network.
4. The **Test migration** job starts. Monitor the job in the portal notifications.
5. After the migration finishes, view the migrated Azure VM in **Virtual Machines** in the Azure portal. The machine name has a suffix **-Test**.
6. After the test is done, select and hold (or right-click) the Azure VM in **Replicating machines**. Then select **Clean up test migration**.

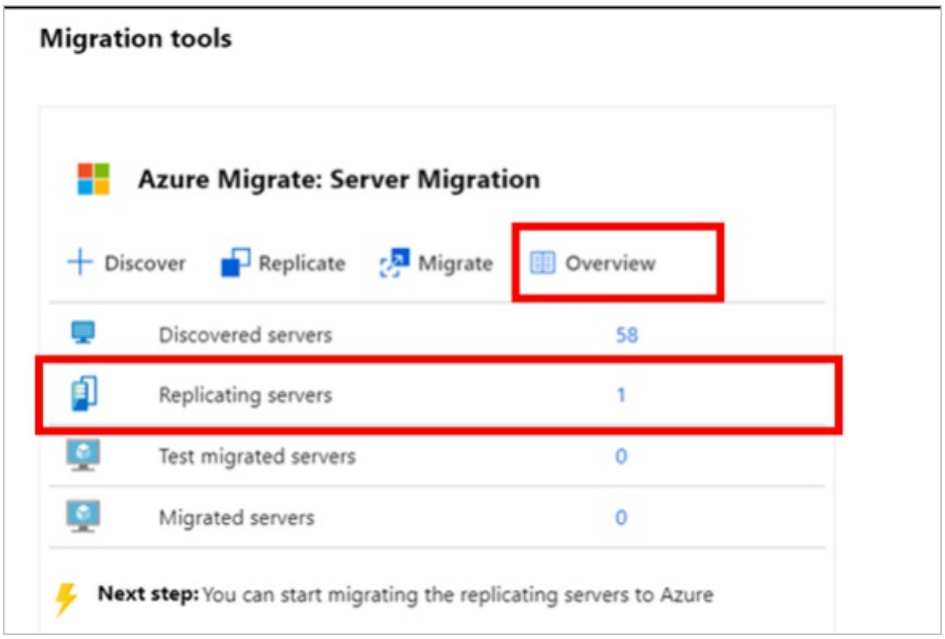


Migrate the VM

Now Contoso admins run a full migration to complete the move.

1. In the Azure Migrate project, go to **Servers** > **Azure Migrate: Server Migration**, and select

Replicating servers.

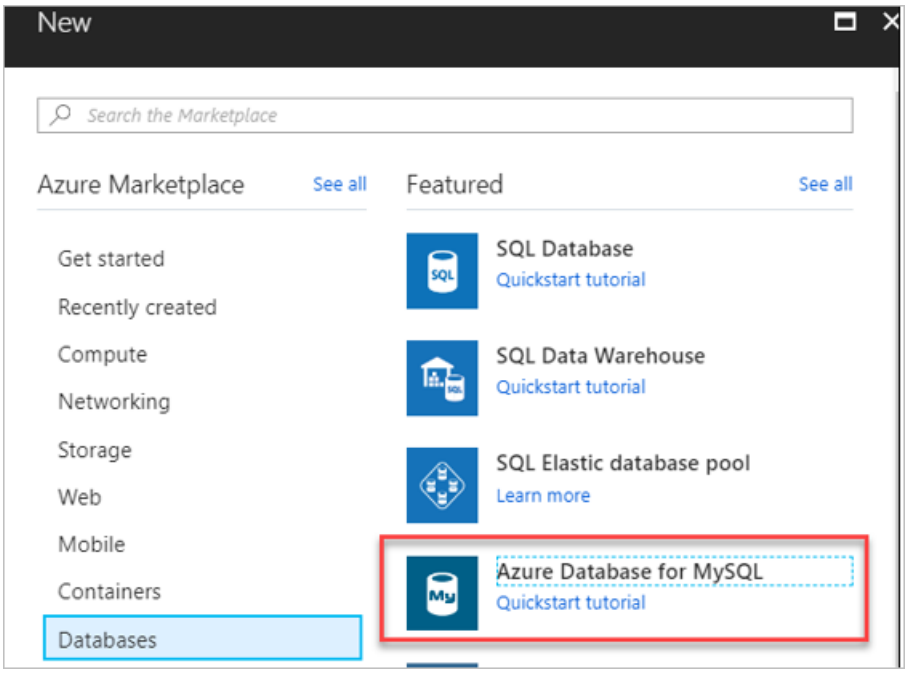


2. In **Replicating machines**, select and hold (or right-click) the VM, and then select **Migrate**.
3. In **Migrate** > **Shut down virtual machines and perform a planned migration with no data loss**, select **Yes** > **OK**.
   - By default, Azure Migrate shuts down the on-premises VM and runs an on-demand replication to synchronize any VM changes that occurred since the last replication occurred. This action ensures no data loss.
   - If you don't want to shut down the VM, select **No**.
4. A migration job starts for the VM. Track the job in Azure notifications.
5. After the job finishes, you can view and manage the VM from the **Virtual Machines** page.

Step 5: Provision Azure Database for MySQL

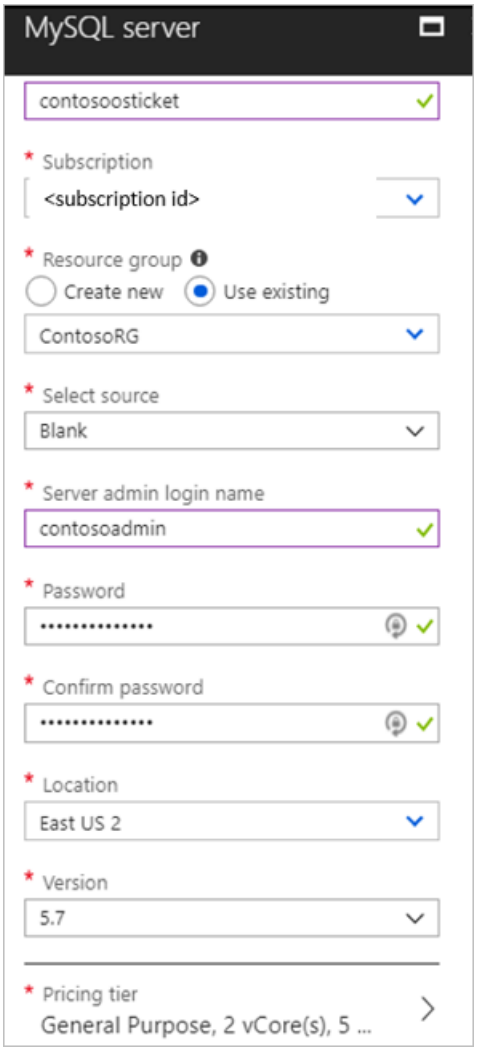
Contoso admins provision a MySQL database instance in the primary region (`East US 2`).

1. In the Azure portal, create an Azure Database for MySQL resource.

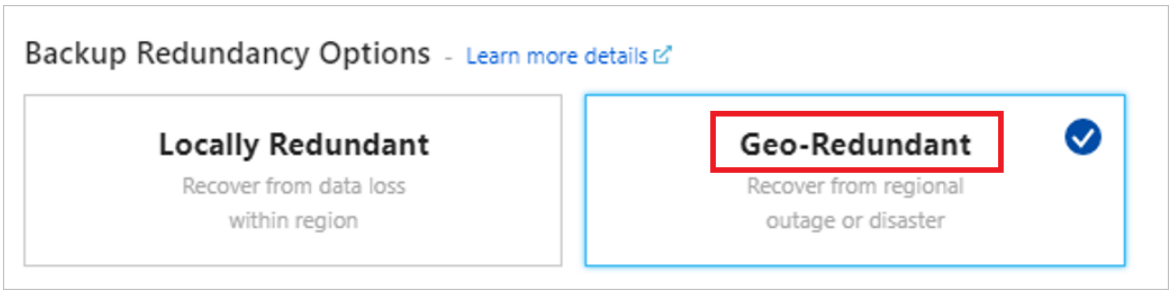


2. Add the name `contosoosticket` for the Azure database. Add the database to the production resource group `ContosoRG`, and specify credentials for it.

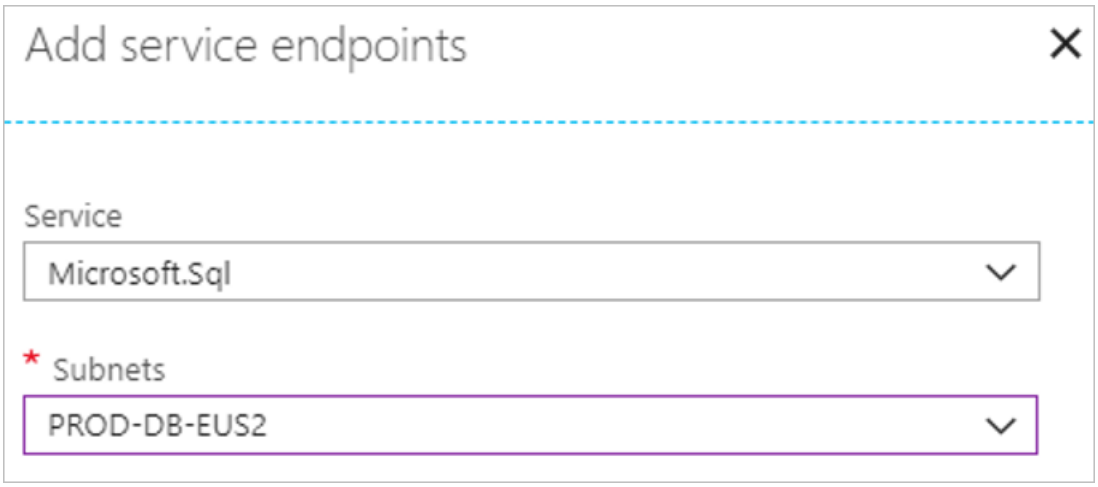
3. The on-premises MySQL database is version 5.7, so select this version for compatibility. Use the default sizes, which match database requirements.



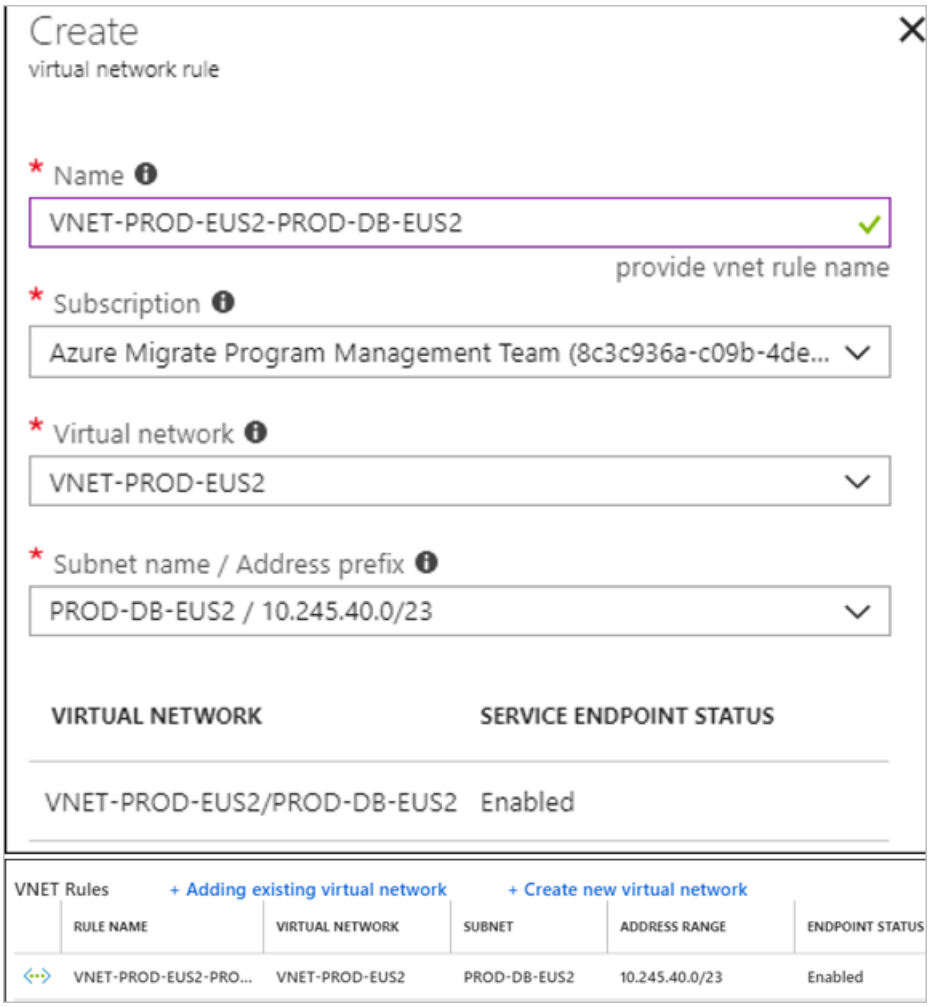
4. For **Backup Redundancy Options**, select **Geo-Redundant**. This option allows you to restore the database in the secondary region (`Central US`) if an outage occurs. You can configure this option only when you provision the database.



5. In the `VNET-PROD-EUS2` network, go to **Service endpoints**, and add a service endpoint (a database subnet) for the SQL service.



6. After adding the subnet, create a virtual network rule that allows access from the database subnet in the production network.



Step 6: Migrate the database

There are several ways to move the MySQL database. Each option requires the Contoso admins to create an Azure Database for MySQL instance for the target. After it's created, they can perform the migration by using two paths that are described in the following steps:

- 6a: Database Migration Service
- 6b: MySQL Workbench backup and restore

Step 6a: Migrate the database via Database Migration Service

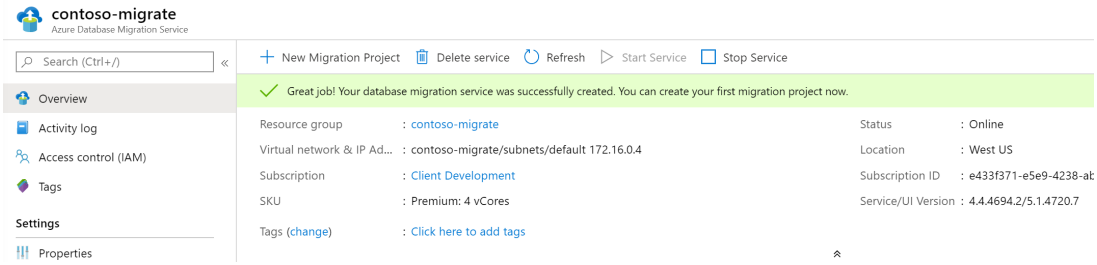
Contoso admins migrate the database via Database Migration Service by following the step-by-step migration

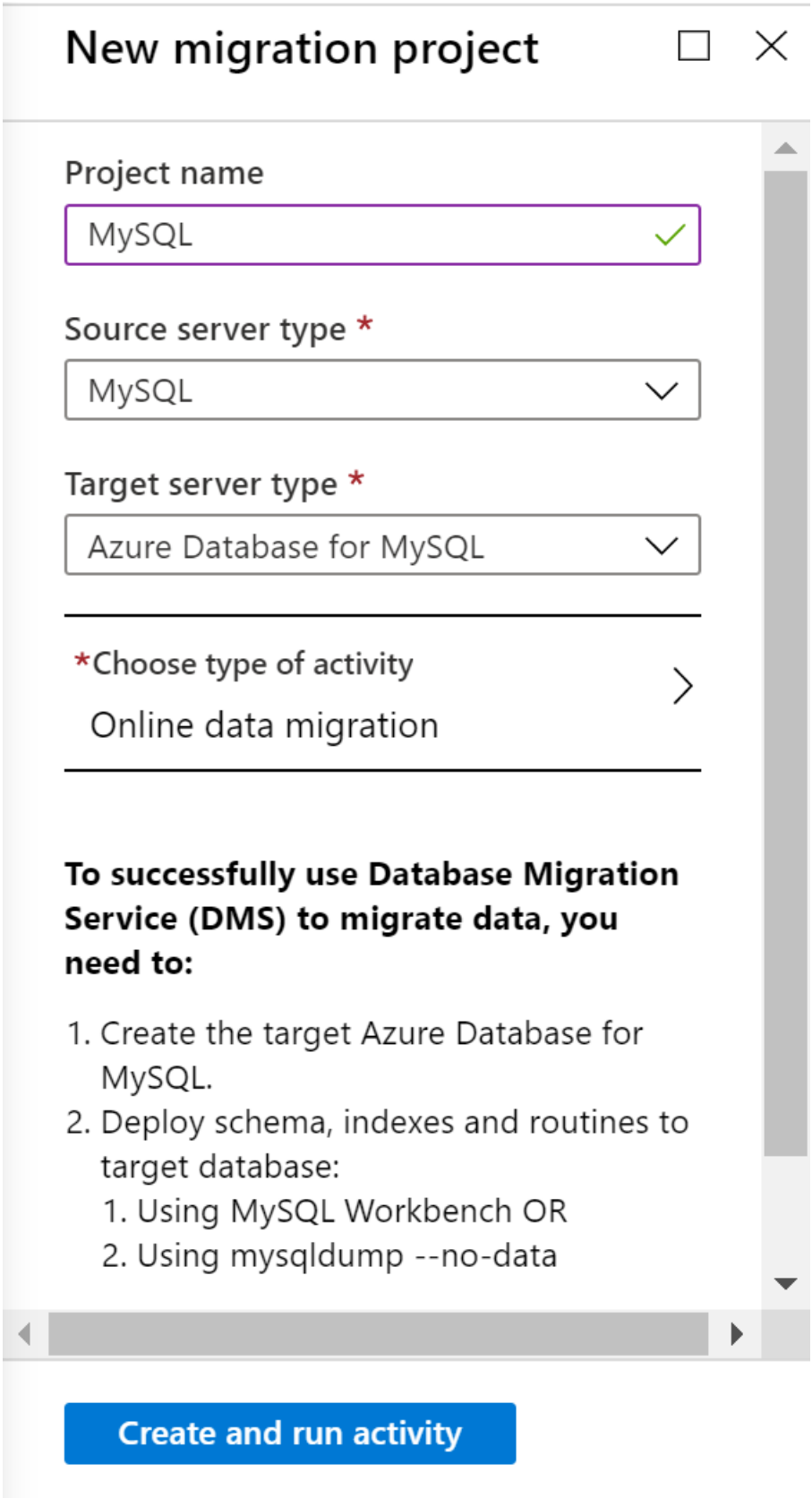
tutorial. They can perform online, offline, and hybrid (preview) migrations by using MySQL 5.6 or 5.7.

> **NOTE**
> MySQL 8.0 is supported in Azure Database for MySQL, but the Database Migration Service tool doesn't yet support that version.

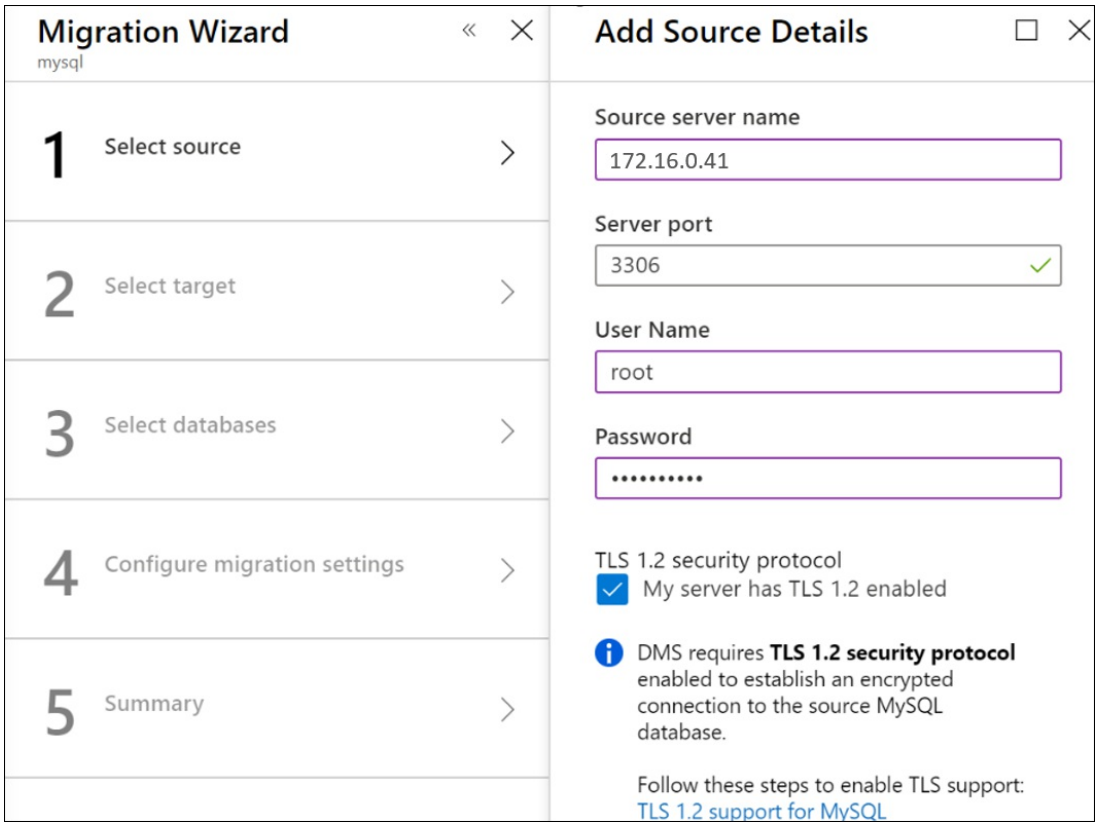
As a summary, Contoso admins must perform the following tasks:

- Ensure all migration prerequisites are met:
  - The MySQL server database source must match the version that Azure Database for MySQL supports. Azure Database for MySQL supports MySQL Community Edition, the InnoDB storage engine, and migration across the source and target with the same versions.
  - Enable binary logging in `my.ini` (Windows) or `my.cnf` (Unix). Failure to do so will cause the following error in the Migration Wizard:
    `Error in binary logging. Variable binlog_row_image has value 'minimal.' Please change it to 'full.'`
    For more information, see the MySQL documentation.
  - User must have `ReplicationAdmin` role.
  - Migrate the database schemas without foreign keys and triggers.
- Create a virtual network that connects via Azure ExpressRoute or a VPN to your on-premises network.
- Create a Database Migration Service instance by using a `Premium` SKU that's connected to the virtual network.
- Ensure that the instance can access the MySQL database via the virtual network. Ensure that all incoming ports are allowed from Azure to MySQL at the virtual network level, the network VPN, and the machine that hosts MySQL.
- Run the Database Migration Service tool:
  - Create a migration project.

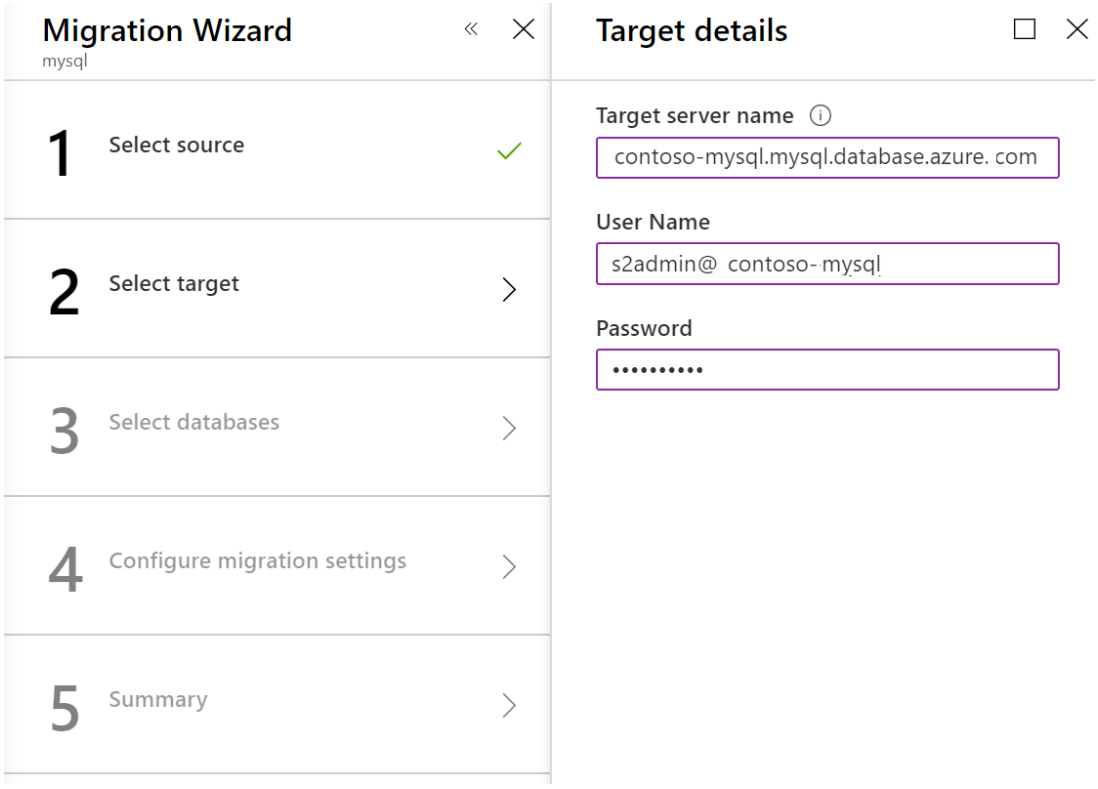




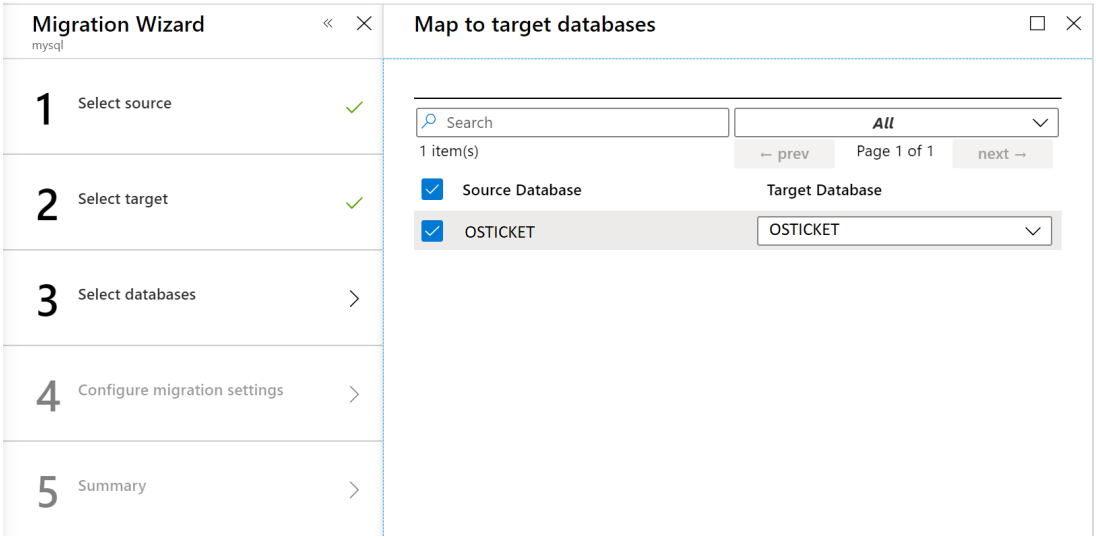
- Add a source (on-premises database).



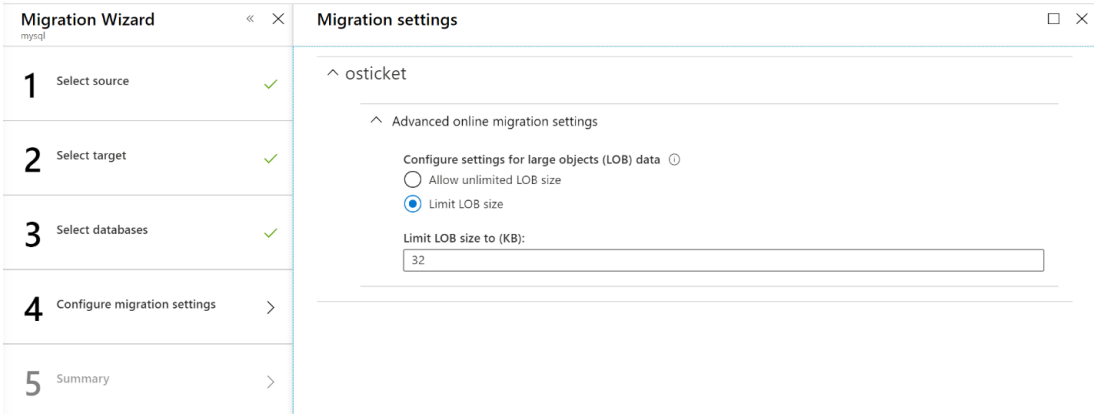
- Select a target.



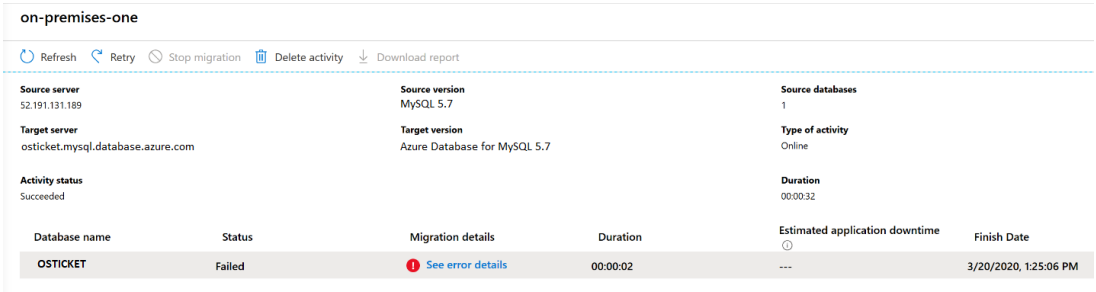
- Select the databases to migrate.



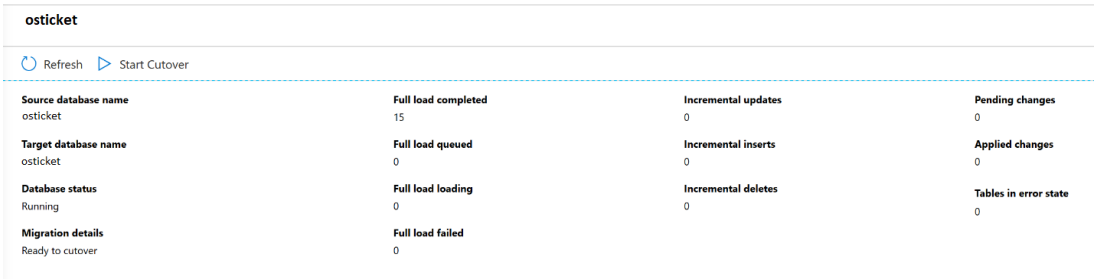
- Configure advanced settings.

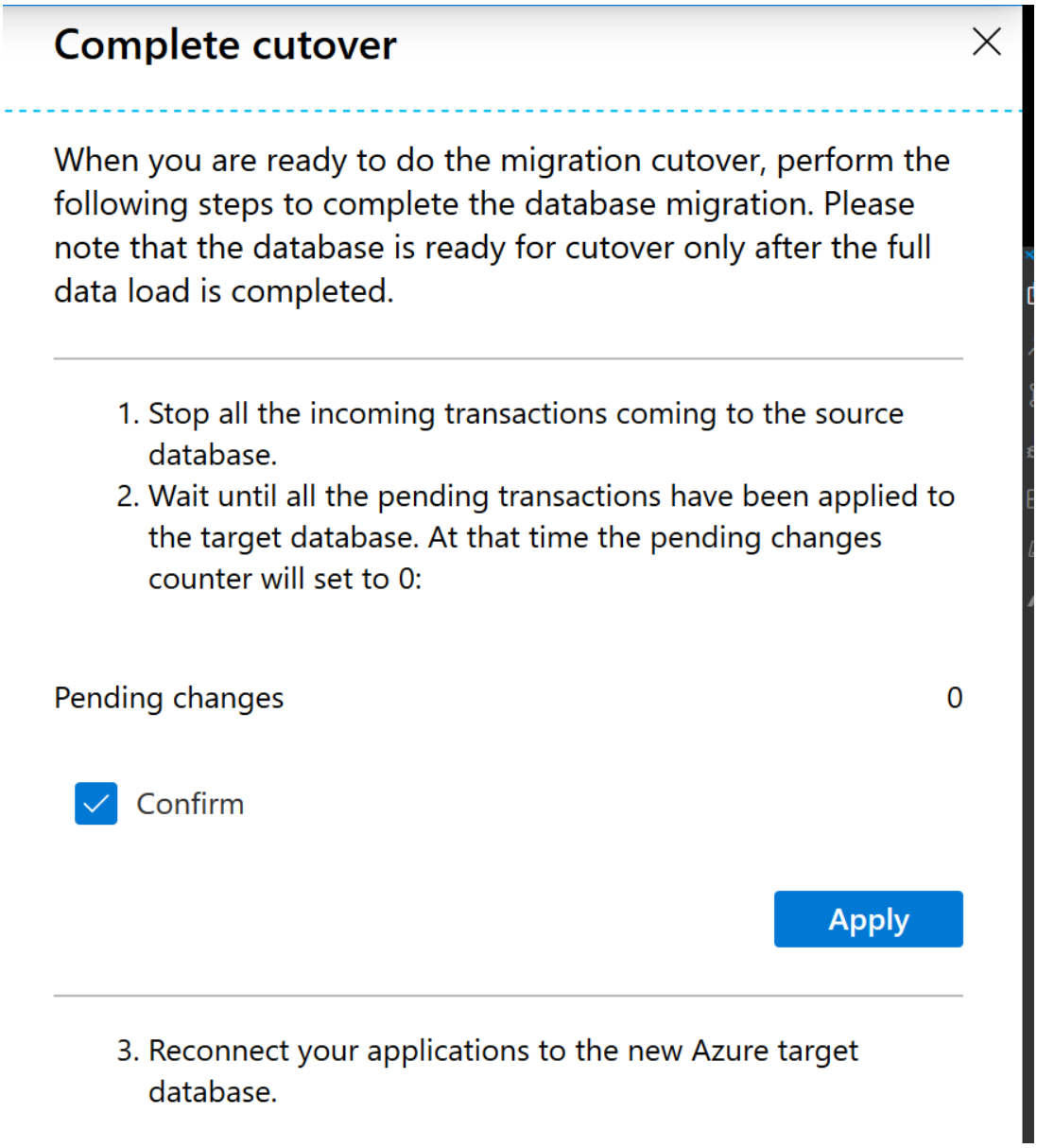


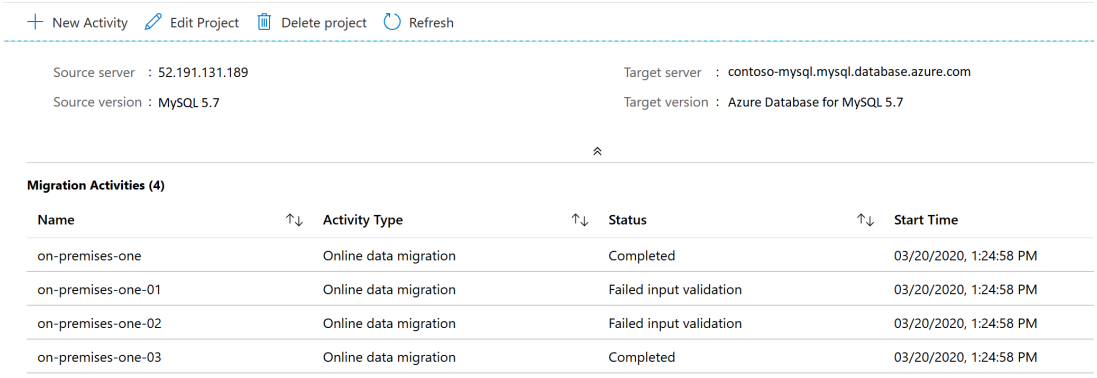
- Start the replication and resolve any errors.



- Perform the final cutover.







- Reinstate any foreign keys and triggers.
- Modify applications to use the new database.

3. Reconnect your applications to the new Azure target database.

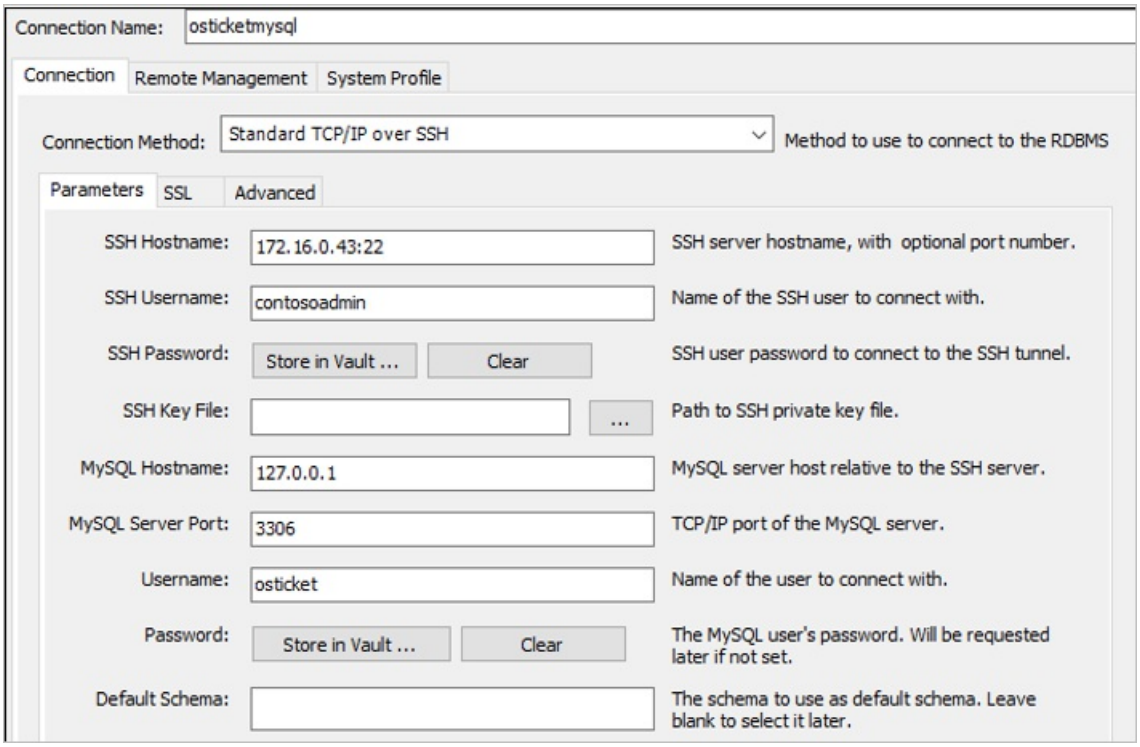
Completed

Step 6b: Migrate the database (MySQL Workbench)

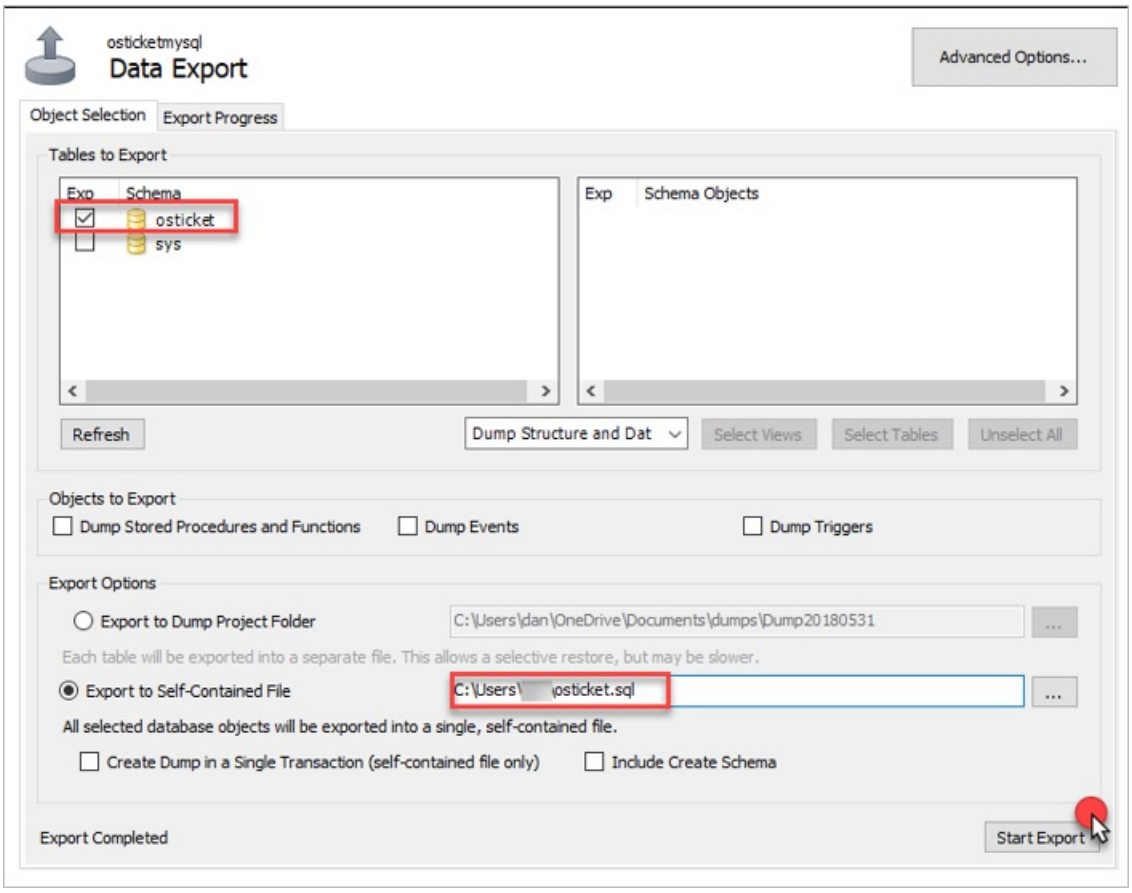
Contoso admins migrate the database by using backup and restore with MySQL tools. They install MySQL Workbench, back up the database from `OSTICKETMYSQL`, and then restore it to Azure Database for MySQL.

Install MySQL Workbench

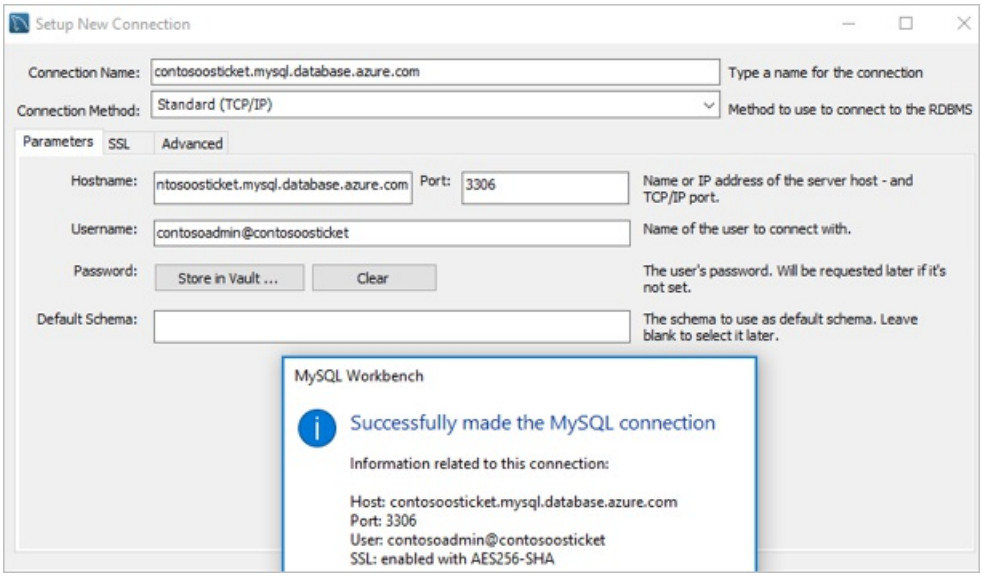
1. Check the prerequisites, and download MySQL Workbench.
2. Install MySQL Workbench for Windows by following the installation instructions.
3. In MySQL Workbench, create a MySQL connection to `OSTICKETMYSQL`.



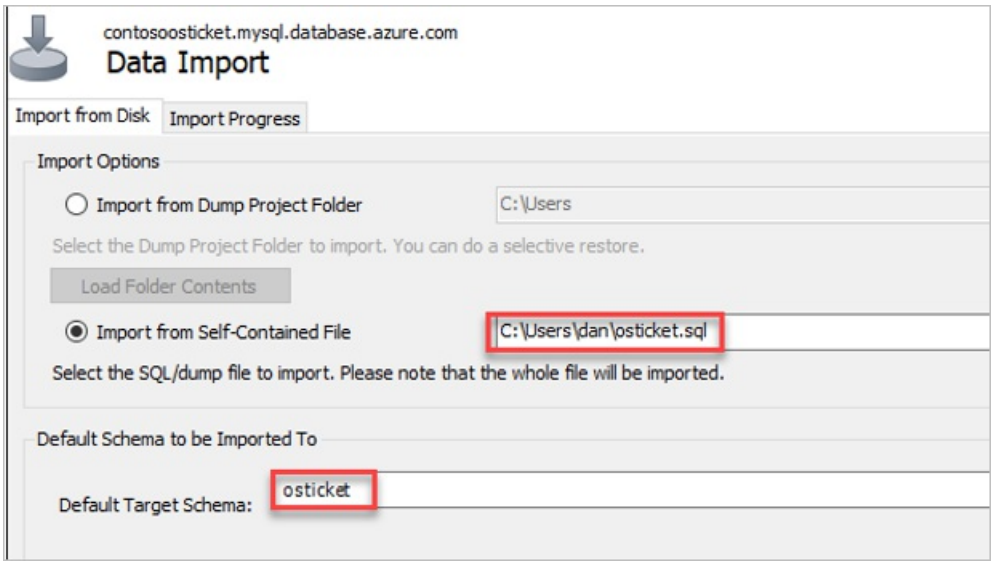
4. Export the database as `osticket` to a local self-contained file.



5. After the database is backed up locally, create a connection to the Azure Database for MySQL instance.



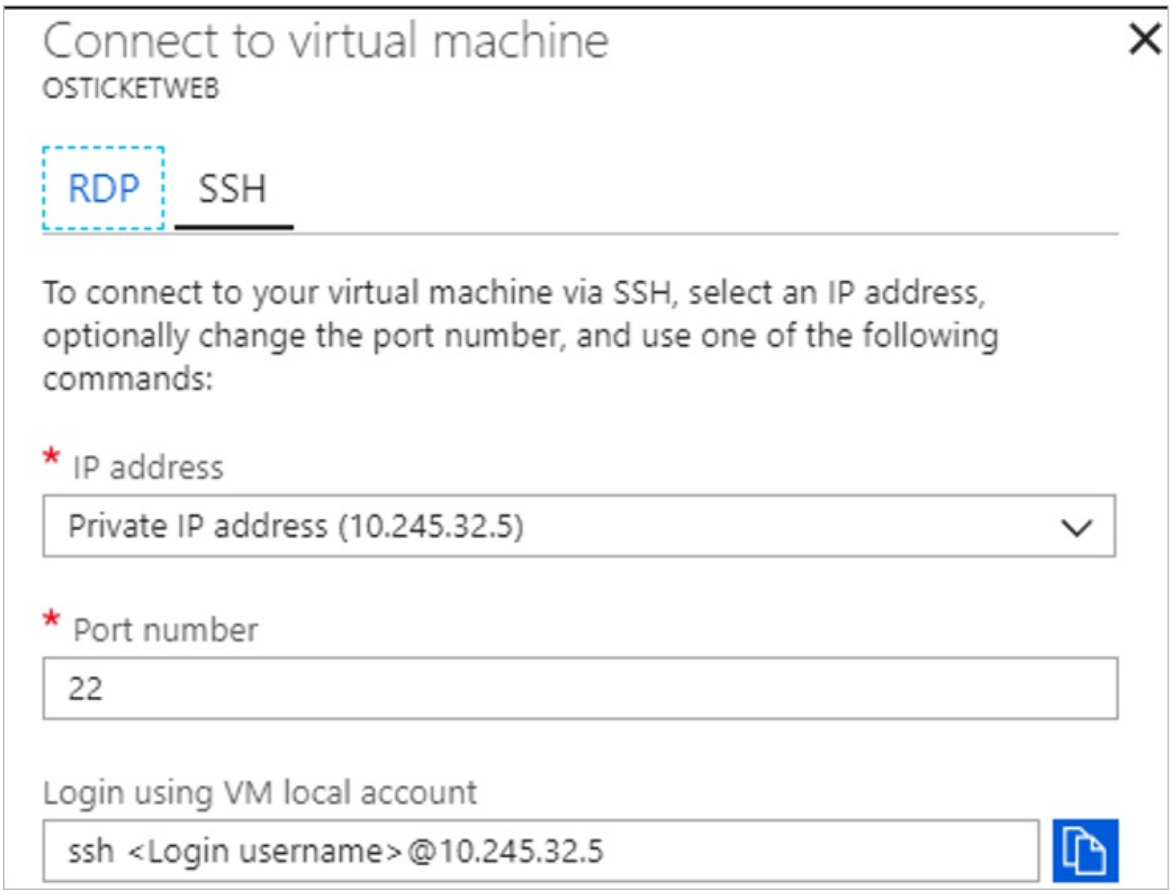
6. Now, import (restore) the database in the Azure Database for MySQL instance from the self-contained file. A new schema (`osticket`) is created for the instance.



Connect the VM to the database

As the final step in the migration process, Contoso admins update the connection string of the application to point to the application database running on the `OSTICKETMYSQL` VM.

1. Make an SSH connection to the `OSTICKETWEB` VM by using PuTTY or another SSH client. The VM is private, so connect by using the private IP address.



```
$ ssh contosoadmin@10.245.32.5
The authenticity of host '10.245.32.5 (10.245.32.5)' can't be established.
ECDSA key fingerprint is SHA256:aJNt+VbYqoZb+mrBlCCj5[redacted].
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '10.245.32.5' (ECDSA) to the list of known hosts.
contosoadmin@10.245.32.5's password:
Welcome to Ubuntu 16.04.3 LTS (GNU/Linux 4.4.0-127-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

136 packages can be updated.
59 updates are security updates.


Last login: [redacted] from 172.16.0.200
contosoadmin@osticketweb:~$
```

2. Make sure that the `OSTICKETWEB` VM can communicate with the `OSTICKETMYSQL` VM. Currently, the configuration is hardcoded with the on-premises IP address `172.16.0.43`.

   Before the update:

```
# Database Options
# ------------------------------
# Mysql Login info
define('DBTYPE','mysql');
define('DBHOST','172.16.0.43');
define('DBNAME','osticket');
define('DBUSER','osticket');
```

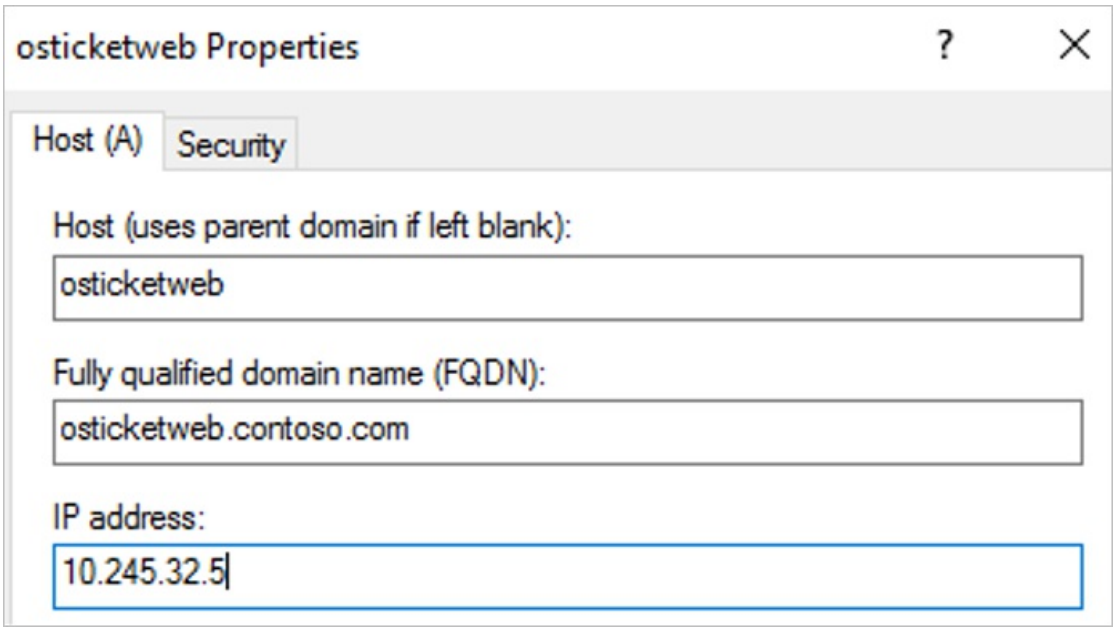
   After the update:

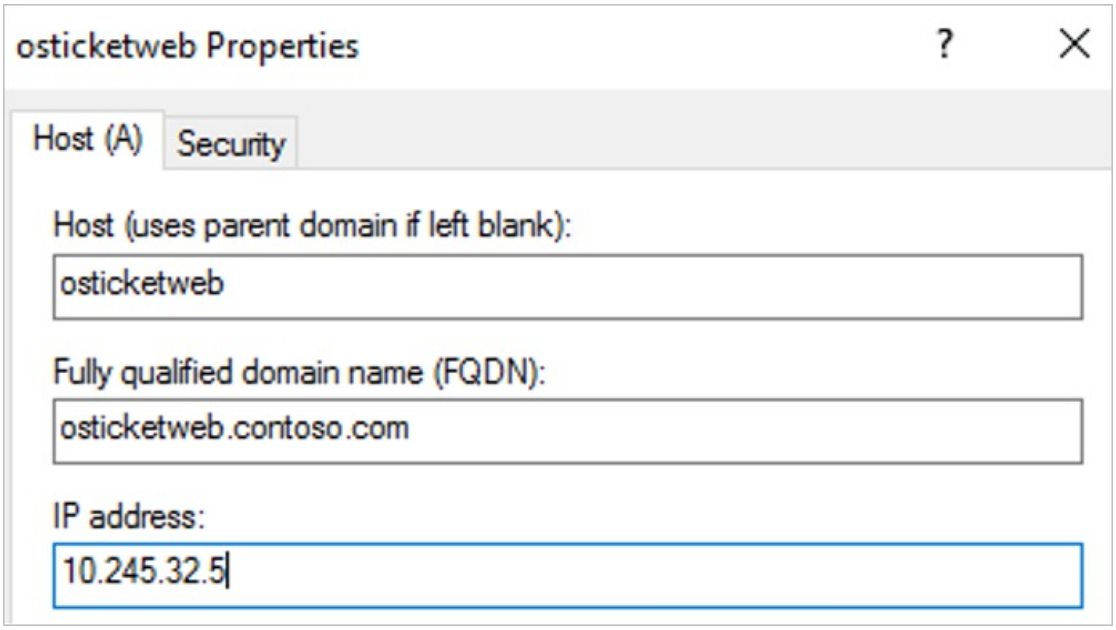
```
# Database Options
# ------------------------------------------
# Mysql Login info
define('DBTYPE','mysql');
define('DBHOST','osticketmysql.contoso.com');
define('DBNAME','osticket');
define('DBUSER','osticket');
-- INSERT --
```

3. Restart the service with `systemctl restart apache2`.

```
contosoadmin@osticketweb:~$ systemctl restart apache2
==== AUTHENTICATING FOR org.freedesktop.systemd1.manage-units
Authentication is required to restart 'apache2.service'.
Authenticating as: contosoadmin,,, (contosoadmin)
Password:
==== AUTHENTICATION COMPLETE ===
contosoadmin@osticketweb:~$
```

4. Finally, update the DNS records for `OSTICKETWEB` and `OSTICKETMYSQL` on one of the Contoso domain controllers.





Need more help?

- Learn about how to run a test migration.
- Learn about how to migrate VMs to Azure.

Review the deployment

With the application now running, Contoso needs to fully operationalize and secure its new infrastructure.

Clean up after migration

With migration complete, the osTicket application tiers are running on Azure VMs.

Now, Contoso needs to do the following tasks:

- Remove the VMware VMs from the vCenter inventory.
- Remove the on-premises VMs from local backup jobs.
- Update internal documentation to show new locations and IP addresses.
- Review any resources that interact with the on-premises VMs. Update any relevant settings or documentation to reflect the new configuration.

- Contoso used Azure Migrate with dependency mapping to assess the `OSTICKETWEB` VM for migration.

Security

The Contoso security team reviews the VM and database to determine any security issues:

- They review the network security groups (NSGs) for the VM to control access. NSGs are used to ensure that only traffic allowed to the application can pass.
- They consider securing the data on the VM disks by using Azure Disk Encryption and Azure Key Vault.
- Communication between the VM and database instance isn't configured for SSL. They'll need to configure SSL to ensure that database traffic can't be hacked.

For more information, see Security best practices for IaaS workloads in Azure.

Business continuity and disaster recovery

For business continuity and disaster recovery, Contoso takes the following actions:

- **Keep data safe.** Contoso backs up the data on the application VM by using Azure VM backup. The company doesn't need to configure backup for the database. Azure Database for MySQL automatically creates and stores server backups. Contoso selected to use geo-redundancy for the database so that it's resilient and production ready.
- **Keep applications up and running.** Contoso replicates the application VMs in Azure to a secondary region by using Site Recovery. For more information, see Quickstart: Set up disaster recovery to a secondary Azure region for an Azure VM.

Licensing and cost optimization

- After deploying resources, Contoso assigns Azure tags as defined during the Azure infrastructure deployment.
- There are no licensing issues for the Contoso Ubuntu servers.
- Contoso will use Azure Cost Management + Billing to ensure the company stays within budgets established by the IT leadership.

Overview of Moodle manual migration

3/6/2021 • 2 minutes to read • Edit Online

Moodle is a free, open-source learning management system written in PHP. This guide explains how to migrate a Moodle deployment from an on-premises environment to Azure. The guide provides steps for two different approaches that use either the Azure portal or the Azure command-line interface (Azure CLI).

Prerequisites

Before starting migration, you need the following prerequisites:

- On-premises software updated and patched to the following versions:
  - Ubuntu 18.04 LTS
  - NGINX 1.14
  - MySQL 5.6, 5.7, or 8.0 database server. This guide uses Azure Database for MySQL.
  - PHP 7.2, 7.3, or 7.4
  - Moodle 3.8 or 3
- Your Moodle website set to **Maintenance mode**.
- Access to the on-premises infrastructure, to back up the Moodle deployment and configurations, including database configurations.
- Azure CLI and AzCopy installed on-premises.
- An Azure subscription and Azure Blob Storage account created.

Moodle migration process

Migrating Moodle with an Azure Resource Manager (ARM) template creates the infrastructure in Azure, then migrates the Moodle software stack and associated dependencies.

The Moodle migration steps to Azure break down into the following three stages:

1. Pre-migration
2. Application migration
3. Post-migration

Next steps

Continue to How to prepare for a Moodle migration.

How to prepare for a Moodle migration

5/18/2021 • 5 minutes to read • Edit Online

Before you migrate a Moodle application from your on-premises environment to Azure, you should export your data. This guide explains the steps of the export process.

Install the Azure CLI

Follow these steps to set up the Azure CLI in your on-premises environment:

1. On a host that you can use for Azure tasks, enter this command to install the Azure CLI:

   ```
   curl -sL https://aka.ms/InstallAzureCLIDeb | sudo bash
   ```

2. In the Azure CLI, enter this command to sign in to your Azure account:

   ```
   az login -u <username> -p <password>
   ```

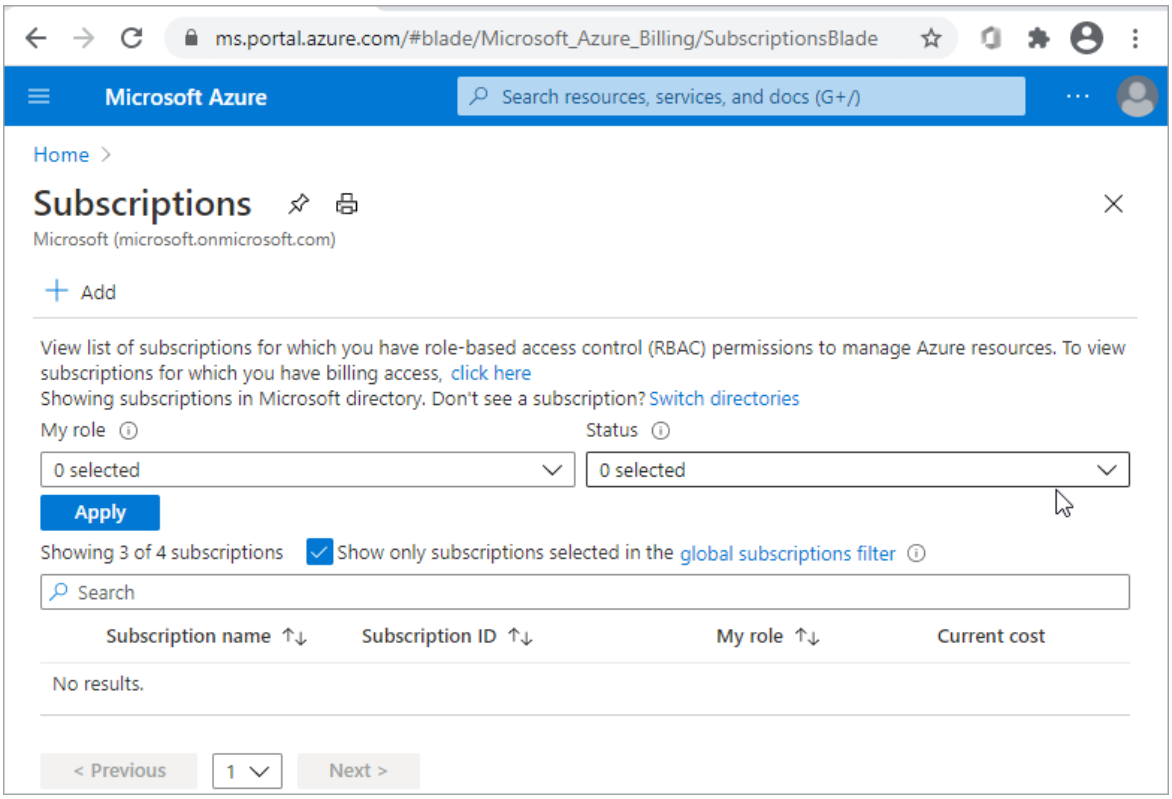
3. If the Azure CLI opens a browser window or tab, sign in to Azure with your Microsoft account. If a browser window doesn't open, go to https://aka.ms/devicelogin, and enter the authorization code displayed in your terminal.

Create a subscription

Skip this step if you already have an Azure subscription.

If you don't have an Azure subscription, you can create one for free. You can also either set up a pay-as-you-go subscription, or you can create a subscription in the Azure portal.

- To use the Azure portal to create a subscription, open **Subscriptions**, select **Add**, and enter the required information.



- To use the Azure CLI to create a subscription, enter this command:

```
az account set --subscription '<subscription name>'
```

An example command is:

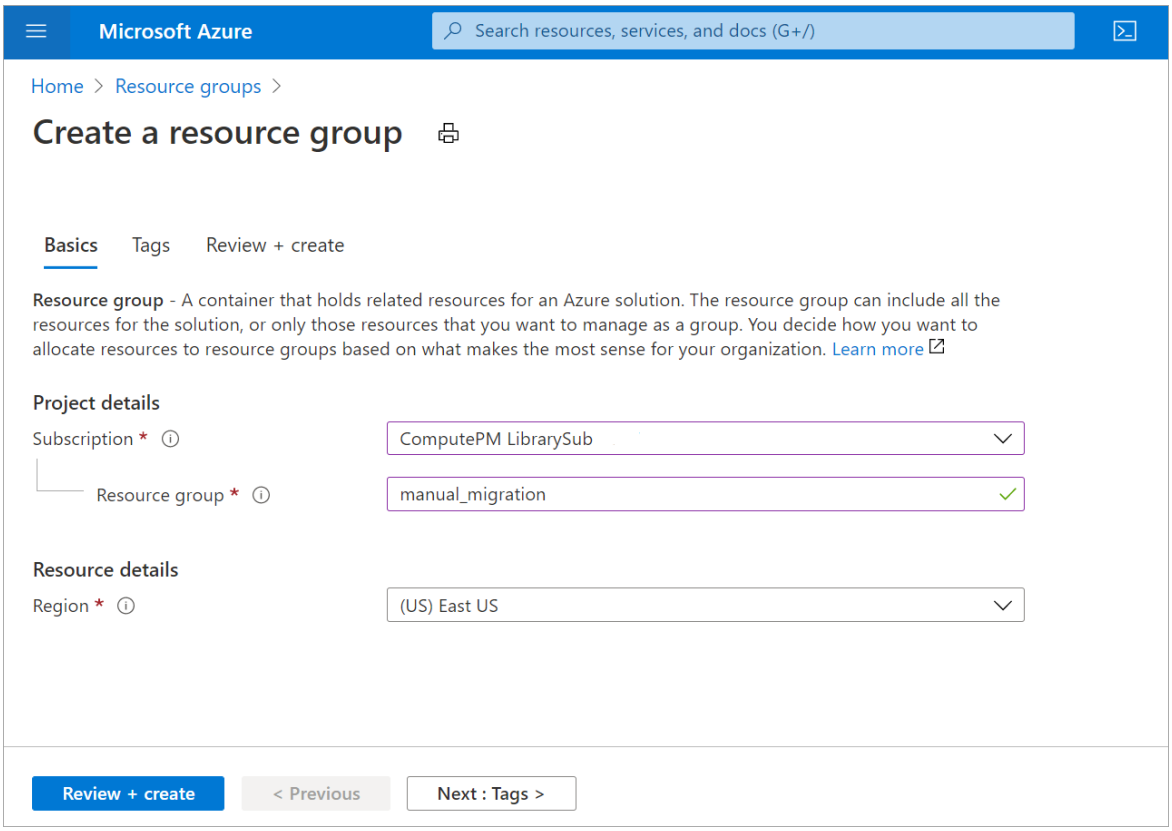
```
az account set --subscription 'ComputePM LibrarySub'
```

Create a resource group

Once you set up your Azure subscription, create a resource group in Azure using either the Azure portal or the Azure CLI.

- To use the Azure portal, follow these steps:

  1. Open **Resource Groups**, and select **Add**.
  2. Enter your subscription name, a resource group name, and a region. See Data residency in Azure for a list of available regions. Make a note of the name of the resource group that you enter so that you can use that name in later steps.
  3. Select **Review + create**.



- To use the Azure CLI to create a resource group, enter this command:

```
az group create -l <region> -n <resource group name> -s '<subscription name>'
```

For example, enter:

```
az group create -l eastus -n manual_migration -s 'ComputePM LibrarySub'
```

The value you provide with the `-l` parameter specifies the default location. Use the same location that you used in previous steps. Make a note of the name of the resource group that you create, and use that name in later steps.

Create a storage account

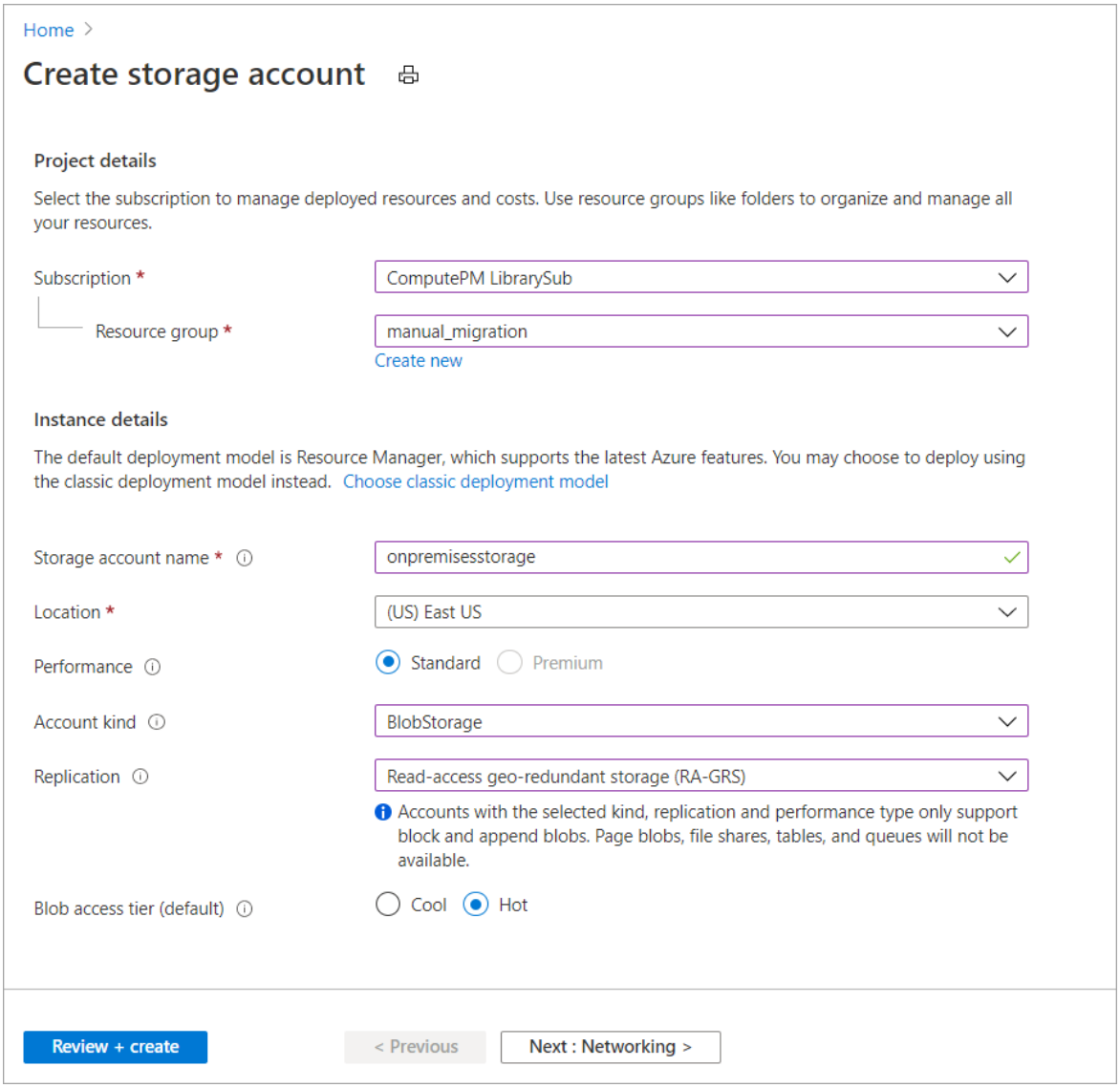
Next, create a storage account within the resource group that you just created. You'll use this storage account to back up your on-premises Moodle data.

You can use either the Azure portal or the Azure CLI to create a storage account.

- To use the Azure portal, follow these steps:

1. Go to **Create storage account**.
2. Enter the following information:
   - Your Azure subscription name
   - The name of the resource group that you just created
   - A storage account name
   - Your region
3. For **Account kind**, select **BlobStorage** from the drop-down list.
4. For **Replication**, select **Read-access geo-redundant storage (RA-GRS)** from the drop-down list.

5. Select **Review + create**.



- To use the Azure CLI to create the storage account, enter this command:

```
az storage account create -n <storage account name> -g <resource group name> --sku <storage account SKU> --kind <storage account type> -l <region>
```

An example command is:

```
az storage account create -n onpremisesstorage -g manual_migration --sku Standard_LRS --kind BlobStorage -l eastus
```

The `--kind` parameter specifies the storage account's type.

Back up on-premises data

Before backing up your on-premises Moodle data, turn on **Maintenance mode** on your Moodle website by following these steps:

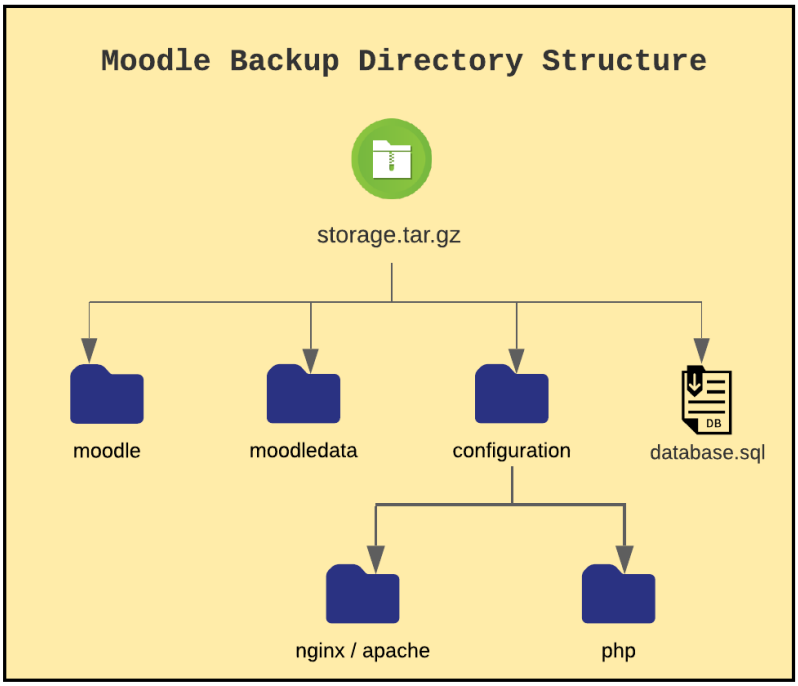
1. From the Moodle instance in your on-premises environment, enter this command:

```
sudo /usr/bin/php admin/cli/maintenance.php --enable
```

2. Enter the following command to check the status of your Moodle website:

```
sudo /usr/bin/php admin/cli/maintenance.php
```

You should back up on-premises Moodle and moodledata files, configurations, and databases to a single directory. The following diagram summarizes this recommendation:



Create a storage directory

Before copying your data, create an empty storage directory in any desired location. For example, if the location is `/home/azureadmin`, enter these commands:

```
sudo -s
cd /home/azureadmin
mkdir storage
```

Back up Moodle directories

In your on-premises environment, the `moodle` directory contains website HTML content. The `moodledata` directory contains Moodle website data.

Enter these commands to copy files from the `moodle` and `moodledata` directories into the storage directory:

```
cp -R /var/www/html/moodle /home/azureadmin/storage/
cp -R /var/moodledata /home/azureadmin/storage/
```

Back up PHP and web server configurations

To back up configuration files, follow these steps:

1. Enter these commands to create a new directory under your storage directory:

   ```
   cd /home/azureadmin/storage
   mkdir configuration
   ```

2. Enter these commands to copy the PHP and NGINX configuration files:

```
cp -R /etc/php /home/azureadmin/storage/configuration/
cp -R /etc/nginx /home/azureadmin/storage/configuration/
```

The `php` directory stores PHP configuration files, such as `php-fpm.conf`, `php.ini`, `pool.d`, and `conf.d`. The `nginx` directory stores ngnix configurations, such as `nginx.conf` and `sites-enabled/dns.conf`.

Back up the database

Follow these steps to back up your database:

1. Enter these commands to check whether mysql-client is installed:

   ```
   sudo -s
   mysql -V
   ```

2. If mysql-client is installed, skip this step. Otherwise, enter this command to install mysql-client:

   ```
   sudo apt-get install mysql-client
   ```

3. Enter this command to back up the database:

   ```
   mysqldump -h <database server name> -u <database user ID> -p<database password> <database name> >
   /home/azureadmin/storage/database.sql
   ```

   For `<database server name>`, `<database user ID>`, `<database password>`, and `<database name>`, use the values that your on-premises database uses.

Create an archive

Enter this command to create an archive file, `storage.tar.gz`, for your backup directory:

```
cd /home/azureadmin/ tar -zcvf storage.tar.gz storage
```

Download and install AzCopy

Enter the following commands to install AzCopy:

```
sudo -s
wget https://aka.ms/downloadazcopy-v10-linux
tar -xvf downloadazcopy-v10-linux
sudo rm /usr/bin/azcopy
sudo cp ./azcopy_linux_amd64_*/azcopy /usr/bin/
```

Copy archived files to Azure Blob Storage

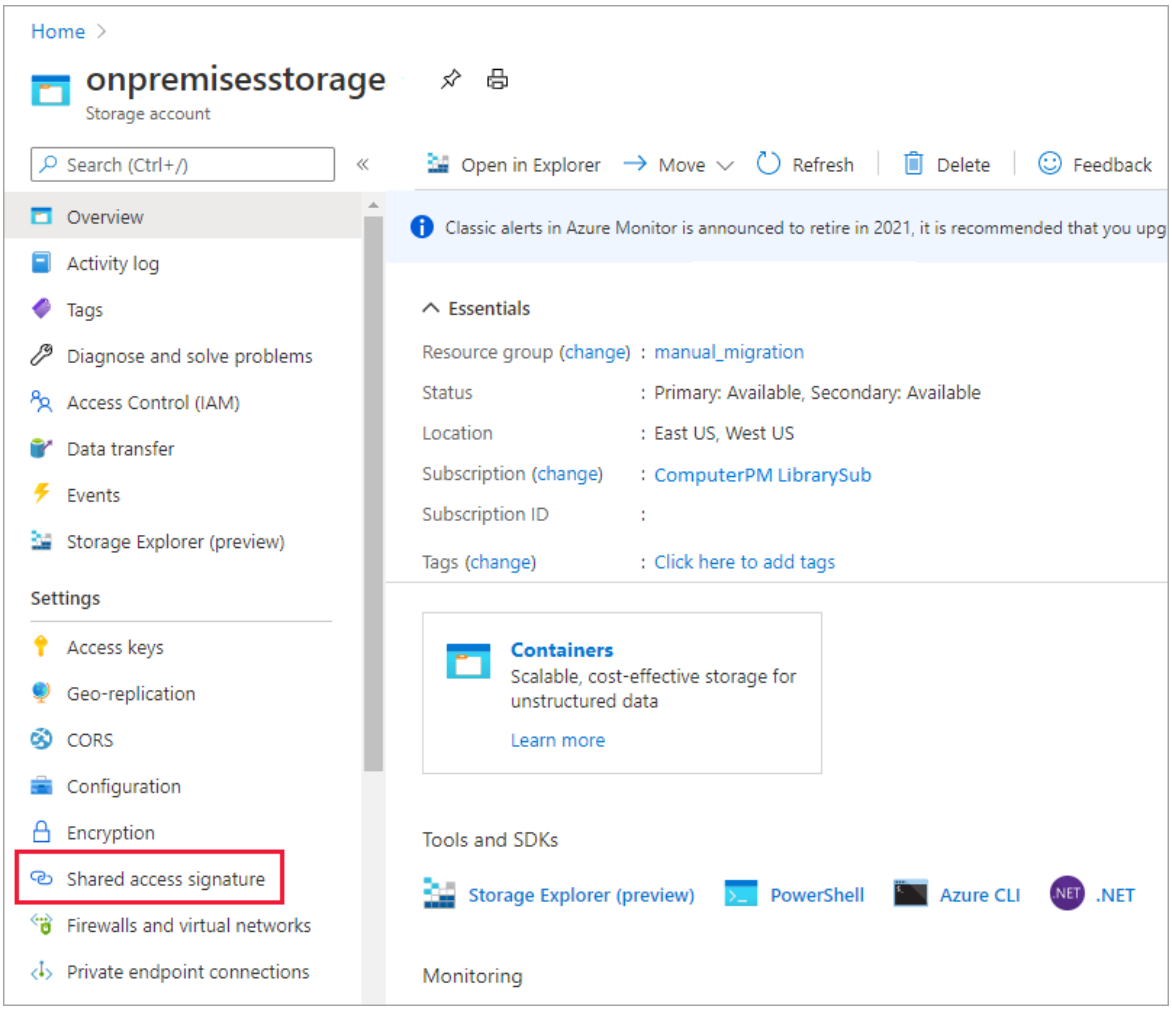
Follow these steps to use AzCopy to copy archived on-premises files to Azure Blob Storage.

Generate a security token

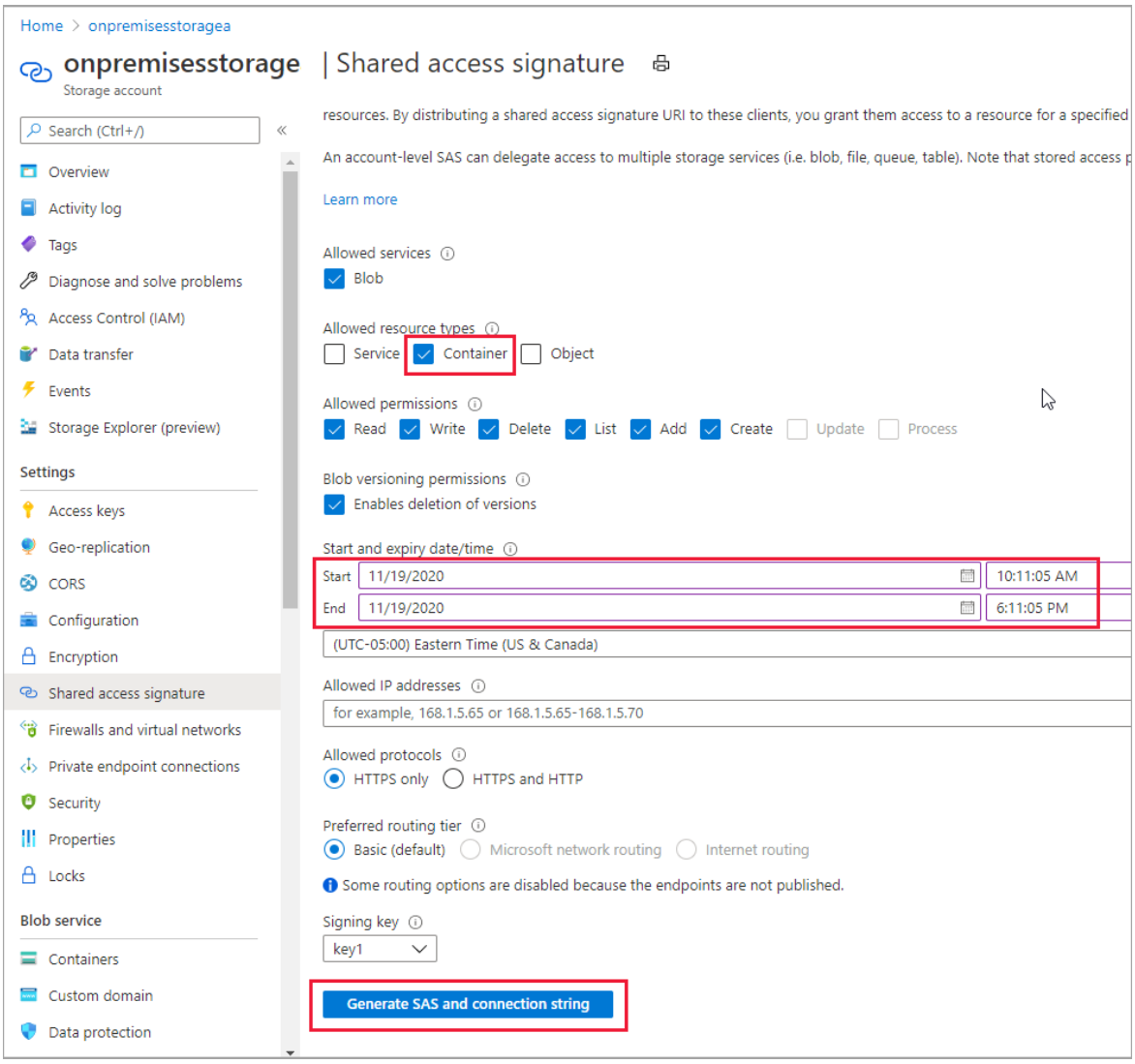
To generate a shared access signature (SAS) token for AzCopy, follow these steps:

1. In the Azure portal, go to the page for the storage account you created earlier.

2. In the left panel, select **Shared access signature**.



3. Under **Allowed resource types**, select **Container**.
4. Under **Start and expiry date/time**, enter a start and end time for the SAS token.
5. Select **Generate SAS and connection string**.



6. Make a copy of the SAS token to use in later steps.

Create a container

Create a container in the storage account. You can either use the Azure CLI or the Azure portal for this step.

- To use the Azure CLI, enter this command:

  ```
  az storage container create --account-name <storage account name> --name <container name> --auth-mode login
  ```

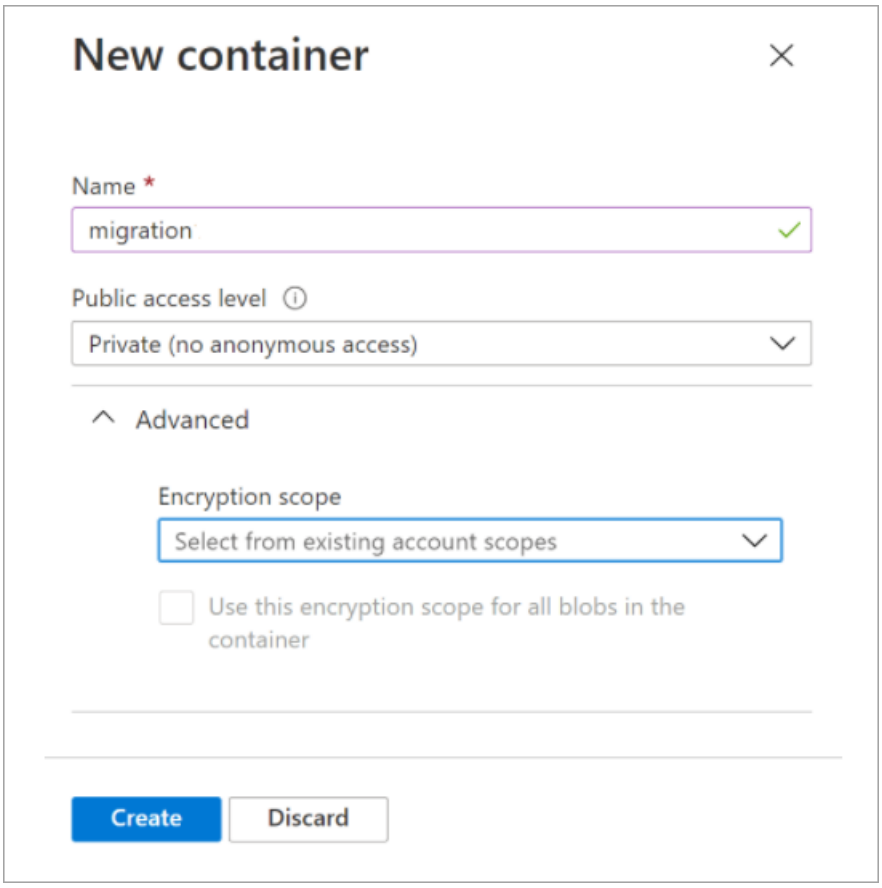
  An example command is:

  ```
  az storage container create --account-name onpremisesstorage --name migration --auth-mode login
  ```

  When you use the `--auth-mode` parameter with a value of `login`, Azure uses your credentials for authentication and then creates the container.

- To use the Azure portal to create the container, follow these steps:

  1. In the portal, go to the page for the storage account you created earlier.
  2. Select **Container**, and then select **Add**.
  3. Enter a name for the container, and then select **Create**.



Copy the archive file to Azure Blob Storage

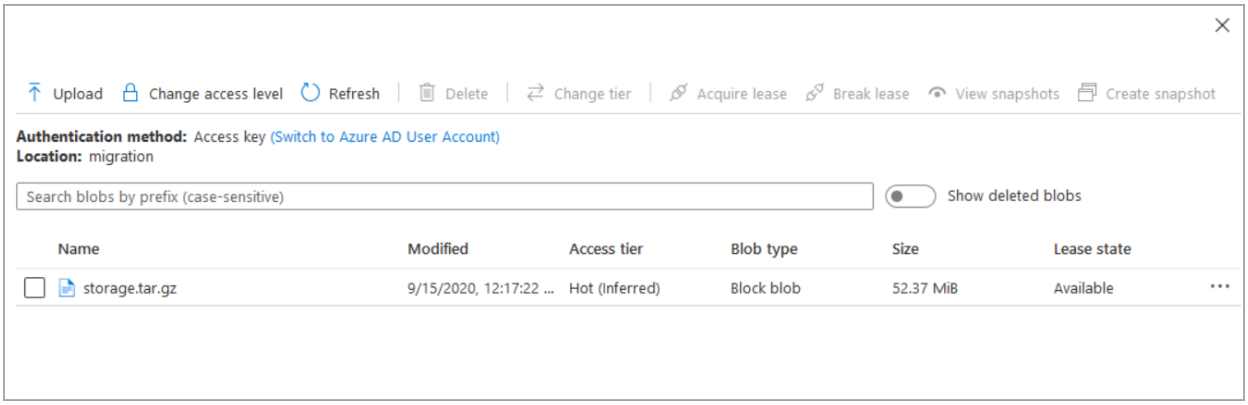
Enter this command to copy your archive file to the container that you created in Blob Storage:

```
sudo azcopy copy /home/azureadmin/storage.tar.gz 'https://<storage account
name>.blob.core.windows.net/<container name>/<SAS token>'
```

An example command is:

```
azcopy copy /home/azureadmin/storage.tar.gz 'https://onpremisesstorage.blob.core.windows.net/migration/?
sv=2019-12-12&ss='
```

Your Blob Storage account should now contain a copy of your archive.



Next steps

Continue to Moodle migration architecture and templates.

Moodle migration architecture and templates

5/18/2021 • 2 minutes to read • Edit Online

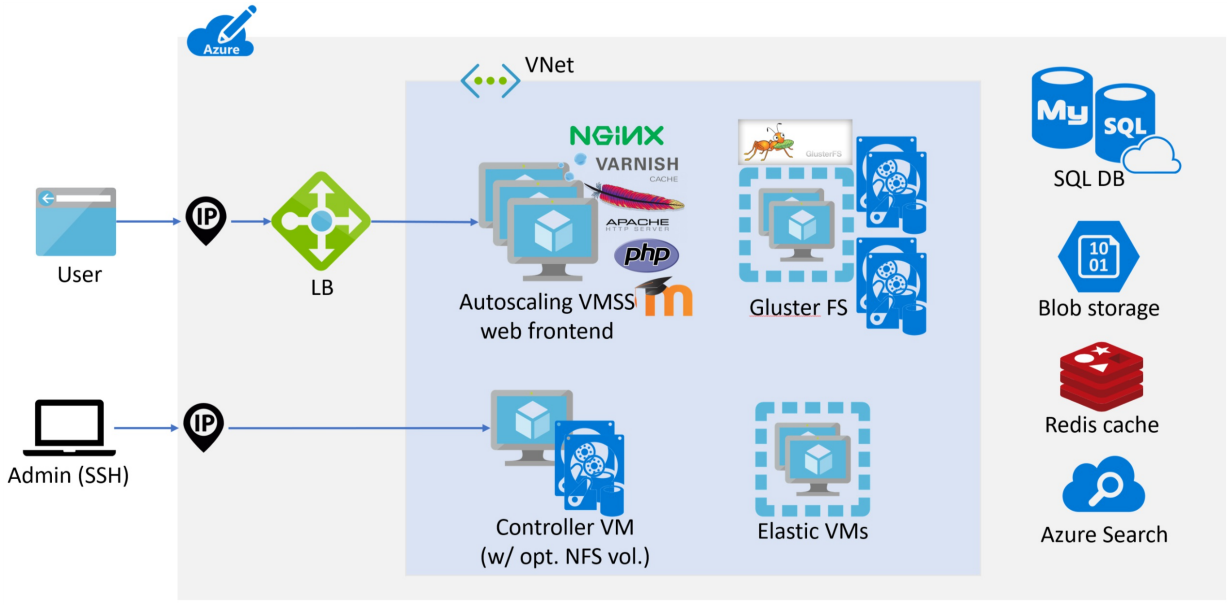
A Moodle migration to Azure includes the following tasks:

1. Deploy Azure infrastructure with Azure Resource Manager (ARM) templates.
2. Download and install AzCopy.
3. Copy the Moodle backup archive to the controller virtual machine instance in the Azure Resource Manager deployment.
4. Migrate the Moodle application and configuration.
5. Set up the Moodle controller instance and worker nodes.
6. Configure PHP and the web server.

This article describes Moodle Azure infrastructure options, and how to deploy Azure resources by using an ARM template that offers your choice of Azure functionality.

Azure infrastructure

The following diagram shows an overview of Azure Moodle infrastructure resources:



ARM template options

To deploy Moodle resources on Azure, you can use a fully configurable ARM template or one of several predefined ARM templates. A fully configurable deployment gives you the most flexibility and deployment choices. You can find the fully configurable template and the predefined templates at the Moodle GitHub repository.

A predefined deployment template uses one of four predefined Moodle sizes: minimal, short-to-mid, large, or maximal.

- *Minimal deployment* requires only two virtual machines (VMs), so it works with an Azure free trial subscription. This deployment uses network file system (NFS), MySQL, and a smaller autoscale web front-end VM SKU with one vCore. This template has a fast deployment time of under 30 minutes.



- *Small-to-mid deployment* supports up to 1,000 concurrent users. This deployment uses NFS, without high availability, and MySQL on eight vCores. This deployment doesn't include options like Elasticsearch or Azure Cache for Redis.

- *Large, high-availability deployment* supports more than 2,000 concurrent users. This deployment uses Azure Files, MySQL with 16 vCores, and Azure Cache for Redis without other options like Elasticsearch.

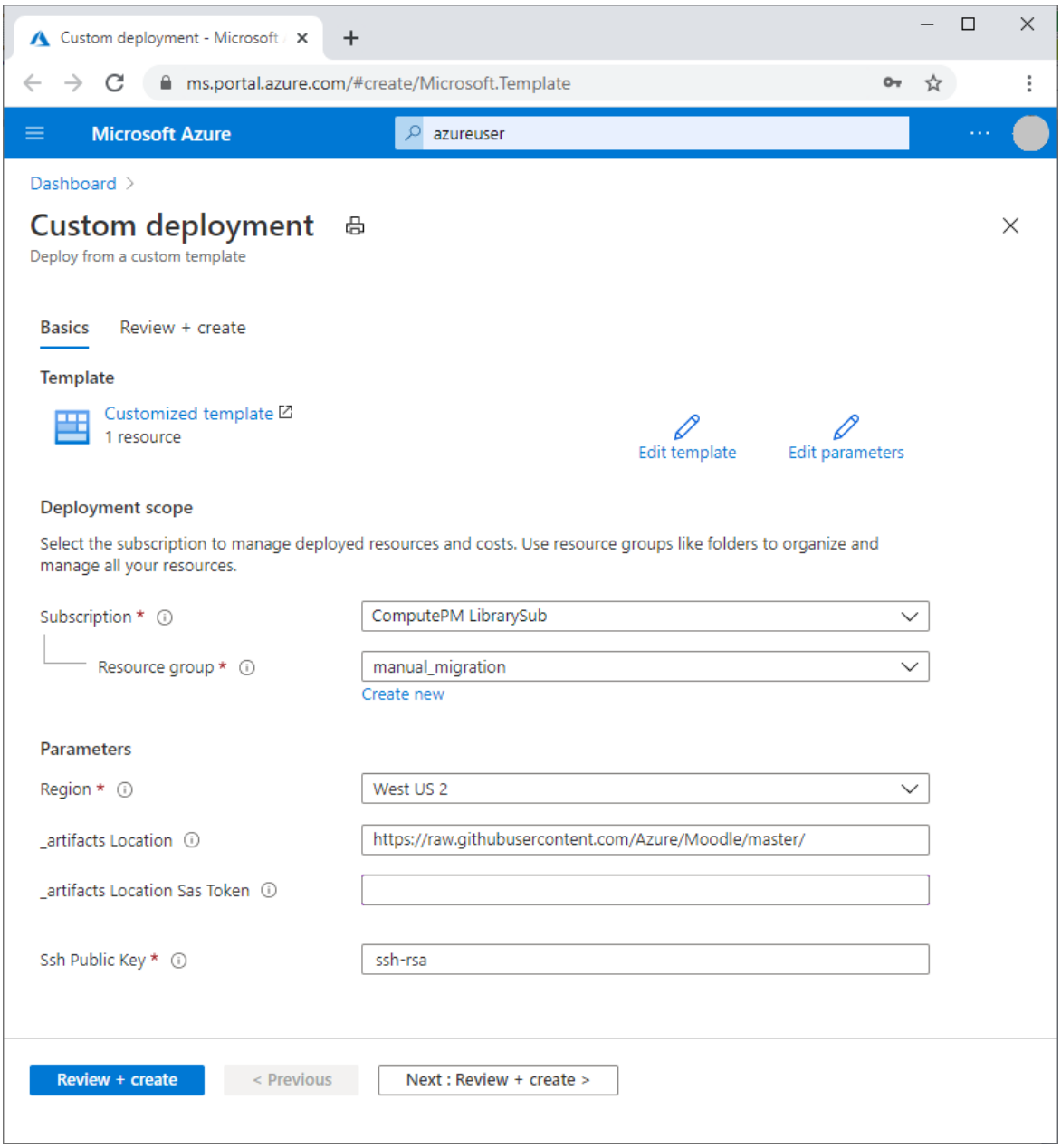
- *Maximal* deployment uses Azure Files, MySQL with the highest SKU, Azure Cache for Redis, Elasticsearch on three VMs, and large storage sizes for both data disks and databases.

Deploy the template

To deploy one of the predefined ARM templates:

1. In the preceding section, select the **Deploy to Azure** button for the deployment you want. This action takes you to the Azure portal.
2. On the **Custom deployment** page in the Azure portal, complete the mandatory **Subscription**, **Resource group**, **Region**, and **SSH Public Key** fields. For information on how to add the SSH key, see Generating a new SSH key and adding it to the ssh-agent.



3. Select **Review + create**.

Edit the template

The predefined ARM templates deploy the following default software versions:

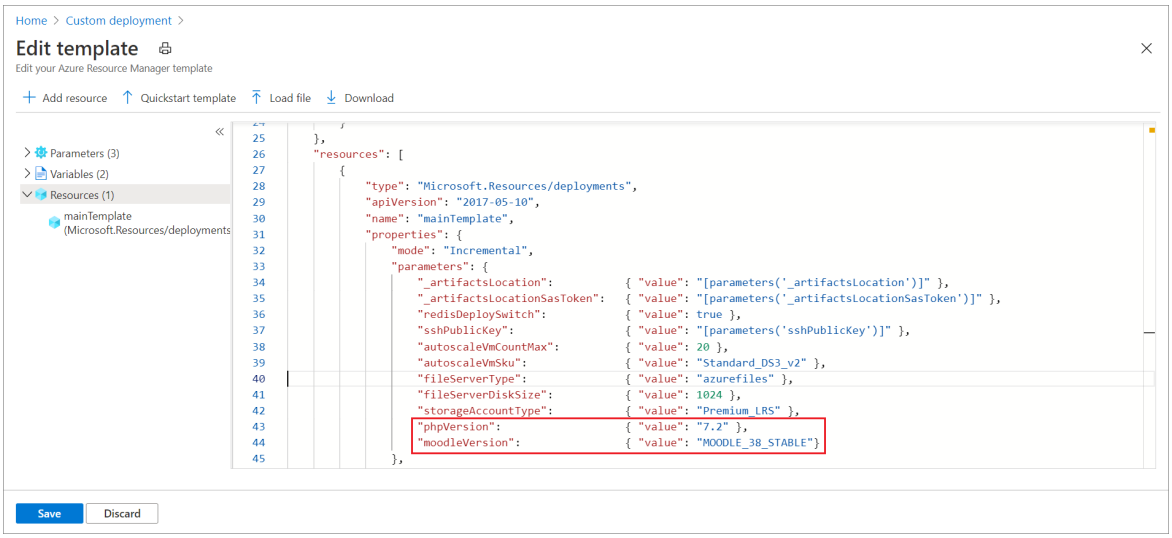
- Ubuntu: 18.04 LTS
- PHP: 7.4
- Moodle: 3.8

If your on-premises PHP and Moodle versions differ from the preceding values, update the template versions to match by following these steps:

1. In the Azure portal, on the ARM template **Custom deployment** page, select **Edit template**.
2. In the **resources** section of the template, under **parameters**, add parameters for your Moodle and PHP versions.

```
"phpVersion":       { "value": "7.2" },
"moodleVersion":    { "value": "MOODLE_38_STABLE"}
```

For example, for Moodle 3.9, the `moodleVersion` value should be `MOODLE_39_STABLE`.



3. Select **Save**.

Next steps

Continue to Moodle migration resources for information about the resources that the ARM template deploys to Azure.

Moodle migration resources

3/6/2021 • 5 minutes to read • Edit Online

When you use an Azure Resource Manager (ARM) template to migrate Moodle, the deployment creates resources within Azure. As part of this deployment process, additional deployments automatically run through child templates. The following sections describe these deployments and the resources they create.

Network template

The network template deployment creates the following resources:

- Azure Virtual Network: A representation of your own network in the cloud. Virtual network is a logical isolation of the Azure cloud that's dedicated to your subscription. When you create a virtual network, your services and virtual machines within it can communicate directly and securely in the cloud. The virtual network that the network template creates includes the virtual network name, API version, location, DNS server name, and address space. The address space contains a range of IP addresses that subnets can use.
- Network security group (NSG): A networking filter, or firewall, that contains a list of security rules. These rules allow or deny network traffic to resources connected to a virtual network.
- Network interface: An interface that an Azure virtual machine can use to communicate with the internet, Azure, and on-premises resources.
- Subnet: A smaller network inside a large network. Subnets are also known as subnetworks. By default, an IP address in a subnet can communicate with any other IP address inside the virtual network.
- Public IP address: An IP address that an Azure resource uses to communicate with the internet. The address is dedicated to the Azure resource.
- Azure Load Balancer: A load balancer that efficiently distributes network or application traffic across multiple servers in a server farm. Load balancer ensures high availability and reliability by only sending requests to servers that are online.
- Azure Application Gateway: An alternative to Load Balancer. All four predefined ARM templates deploy Load Balancer. If you use a fully configurable deployment instead of an ARM template, you can choose Application Gateway instead of Load Balancer. Application Gateway is a web-traffic load balancer that you can use to manage traffic to your web applications. Application Gateway can make routing decisions based on the additional attributes of an HTTP request, such as a URI path or host header.
- Azure Cache for Redis: An in-memory data store based on the open-source software redis. Redis improves the performance and scalability of an application that heavily stores back-end data. It can process large volumes of application requests by keeping frequently accessed data in the server memory. This data can be written to and read from quickly.

Storage template

The storage account template deployment creates an Azure Storage account of type FileStorage. The account has premium performance, locally redundant storage (LRS) replication, and 1 terabyte (TB) of storage. The predefined template is configured so that a storage account with Azure Files creates file shares.

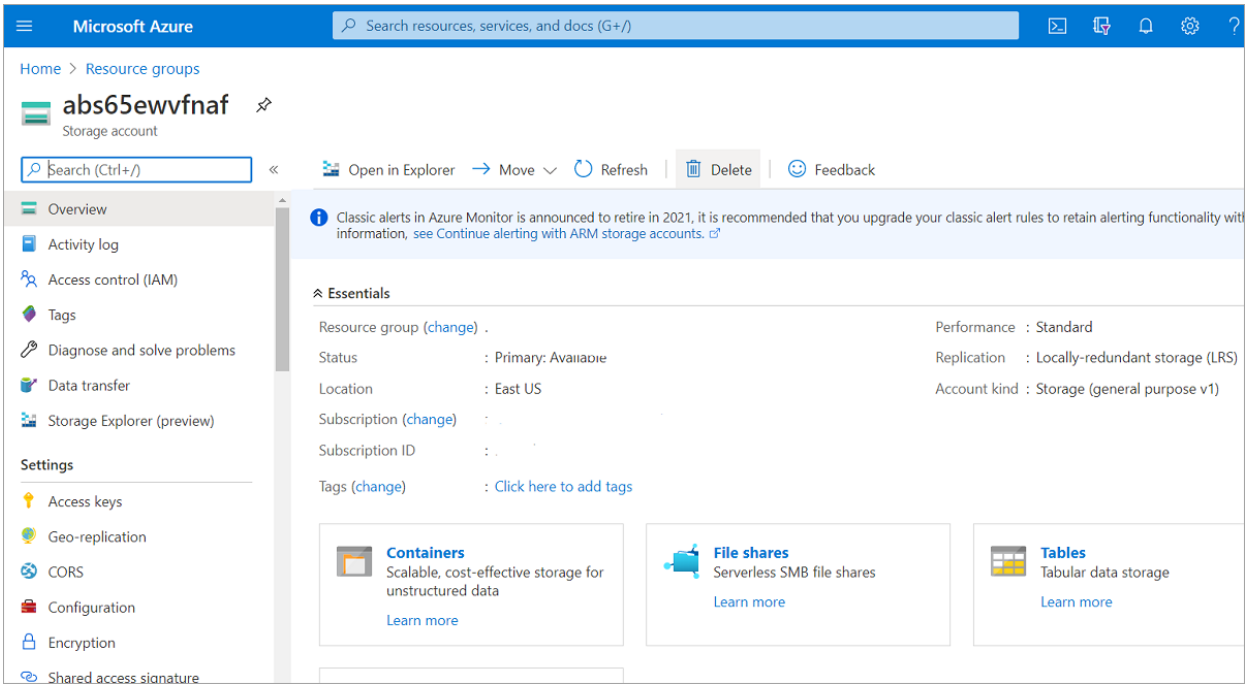
An Azure Storage account contains Azure Storage data objects, such as blobs, files, queues, tables, and disks. The storage account provides a unique namespace for your Azure Storage data that's accessible from anywhere in

the world over HTTP or HTTPS. The following types of Azure storage accounts are available: General-Purpose v1, General-Purpose v2, BlockBlobStorage, FileStorage, and Blob Storage. The replication type can be geo-redundant or LRS and zone-redundant storage. The performance types are standard and premium, and an individual storage account can store up to 500 TB of data, like any other Azure service.

ARM templates support the following storage account types:

- Network File System (NFS): An account type that a remote host can use to mount file systems over a network. The remote host can interact with those file systems as though they're mounted locally. With this design, system administrators can consolidate resources into centralized servers in the network.
- GlusterFS: An open-source distributed file system that can scale out in building-block fashion to store multiple petabytes of data.
- Azure Files: The only public cloud file storage that delivers secure, SMB-based, and fully managed cloud file shares that can also be cached on-premises for performance and compatibility. For NFS and GlusterFS, the replication is standard LRS, and the storage type is general-purpose v1. For Azure Files, the replication is premium LRS, and the type is FileStorage.

These storage mechanisms differ depending on which deployment you choose. NFS and GlusterFS create a container, and Azure Files creates a file share. For minimal and short-to-mid Moodle sizes, the template supports NFS. For large and maximal sizes, the template supports Azure Files. To access the containers and file shares, go to the Azure portal, and select the storage account in the resource group.



Database template

The database template deployment creates an Azure Database for MySQL server. Azure Database for MySQL is easy to set up, manage, and scale. It automates the management and maintenance of your infrastructure and database server, including routine updates, backups, and security. Azure Database for MySQL is built with the latest community edition of MySQL, including versions 5.6, 5.7, and 8.0. To access the database server that the template creates, go to the Azure portal and open the resource group that the deployment process provides. Then go to **Azure Database for MySQL server**. The template gives the database server a server name, a server admin login name, a MySQL version, and a performance configuration.

Virtual machine template

The virtual machine template deployment designates a virtual machine as a controller virtual machine. The

operating system for the controller virtual machine is Ubuntu 18.04.

Virtual machine extensions are small applications that provide post-deployment configuration and automation tasks on Azure Virtual Machines. A virtual machine extension runs a shell script that installs Moodle on the controller virtual machine and captures log files. It creates the `stderr` and `stdout` log files in the `/var/lib/waagent/custom-script/download/0/` folder. You can view these files as a root user.

Scale set template

The scale set template deployment creates a virtual machine scale set. By using a virtual machine scale set, you can deploy and manage a set of autoscaling virtual machines. You can scale the number of virtual machines in the scale set manually or define rules to autoscale based on resource usage like CPU, memory demand, or network traffic. When an instance scales up, it deploys a virtual machine. Then a shell script runs that installs Moodle prerequisites and sets up cron jobs. A virtual machine in a scale set has a private IP address. For more information on viewing the virtual machine instances in a scale set and how to access these instances, see the virtual machine scale set documentation.

Next steps

Continue to Moodle manual migration steps for the next steps in the Moodle migration process.

Moodle manual migration steps

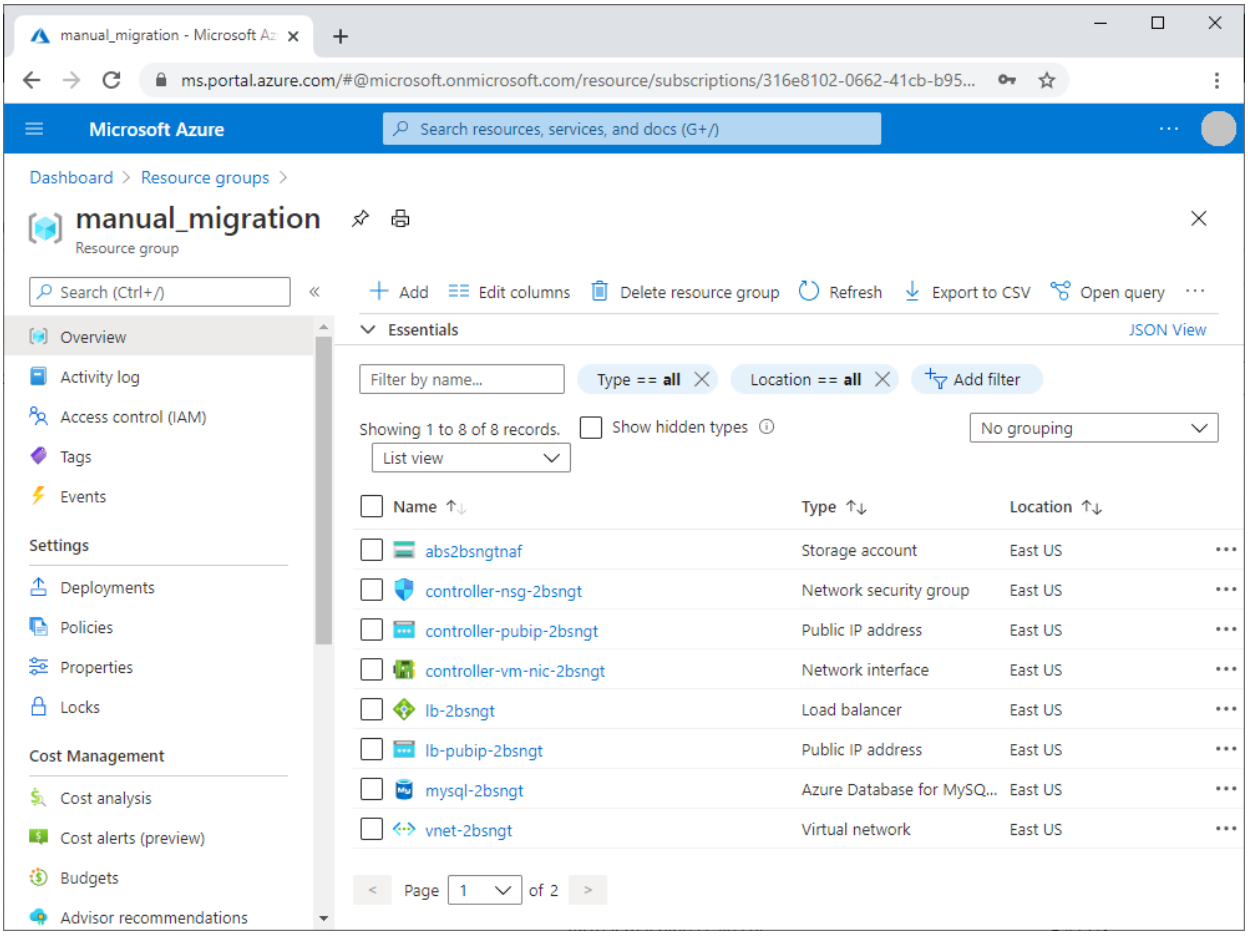
5/18/2021 • 7 minutes to read • Edit Online

This article describes the steps to migrate the on-premises Moodle archive to Azure. The contents of this Moodle archive include the Moodle application, relevant configuration, and a copy of the database from the on-premises Moodle deployment. Once you successfully import the on-premises backup into the Azure infrastructure, you carry out configuration updates for Moodle.

Before you begin this process, make sure to complete all the steps in these articles:

- How to prepare for a Moodle migration
- Moodle migration architecture and templates

After the Azure Resource Manager (ARM) template deployment finishes, sign in to the Azure portal and go to the resource group you created as part of the deployment process. Review the list of newly created infrastructure resources. The created resources look similar to the following image, depending on the ARM template you used for deployment.



Copy the Moodle archive

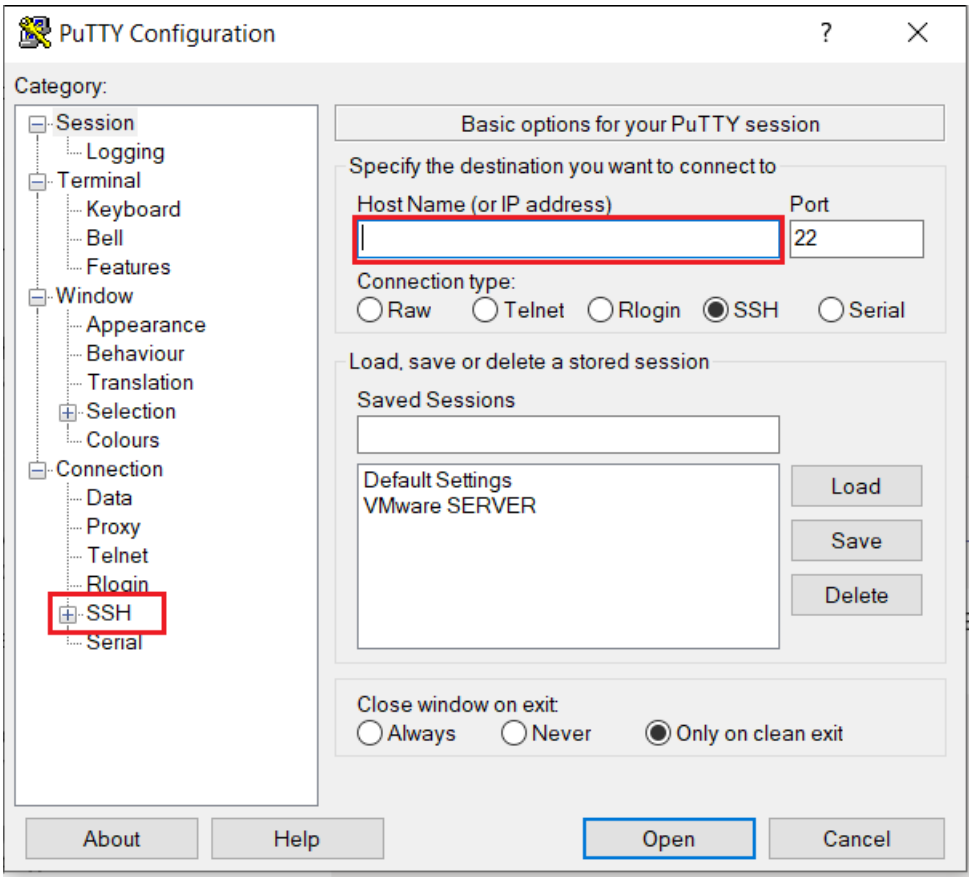
The first step in the migration process is to copy the Moodle backup archive from Azure Blob Storage to the controller virtual machine (VM) for the Moodle deployment. This is the same archive you created in Create an archive.

Sign in to the controller virtual machine

1. Use a free, open-source terminal emulator or serial console tool like PuTTY to sign in to a controller VM.

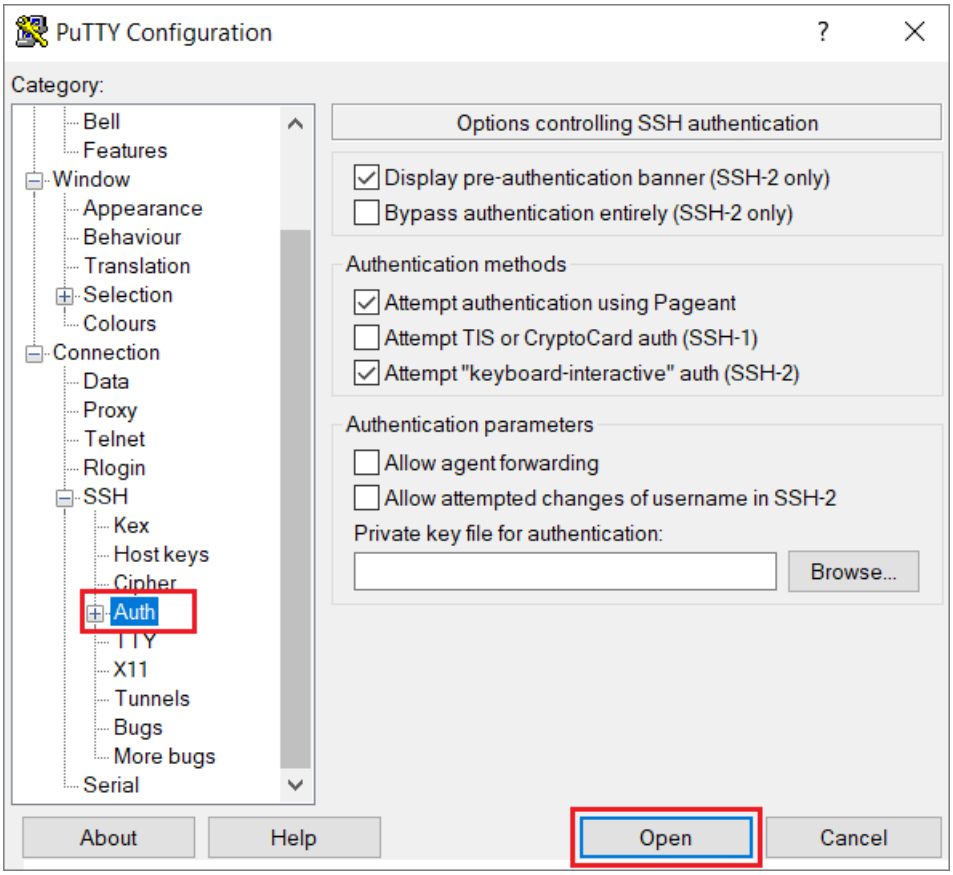
2. In **PuTTY Configuration**, enter the public IP address of the controller VM as the **Host Name**.

3. In the left navigation, expand **SSH**.



4. Select **Auth**, and find the SSH key file you used to deploy the Azure infrastructure with the ARM template.

5. Select **Open**. For user name, enter **azureadmin**, as it's hard-coded in the template.



For more information about PuTTY, see PuTTY general FAQ/troubleshooting questions.

Download and install AzCopy on the controller VM

After you sign in to the controller VM, run the following commands to install AzCopy:

```
sudo -s
wget https://aka.ms/downloadazcopy-v10-linux
tar -xvf downloadazcopy-v10-linux
sudo rm /usr/bin/azcopy
sudo cp ./azcopy_linux_amd64_*/azcopy /usr/bin/
```

Back up the current configuration

Before you start the import process, it's recommended to back up the default or current configuration.

1. Create a backup directory:

   ```
   cd /home/azureadmin/
   mkdir -p backup
   mkdir -p backup/moodle
   mkdir -p backup/moodle/html
   ```

2. Create backups of the `moodle` and `moodledata` directories:

   ```
   mv /moodle/html/moodle /home/azureadmin/backup/moodle/html/moodle
   mv /moodle/moodledata /home/azureadmin/backup/moodle/moodledata
   ```

Copy the Moodle archive to the controller VM

1. Run the following commands to download the compressed `storage.tar.gz` backup file from Azure Blob Storage to the controller VM `/home/azureadmin/` directory:

   ```
   sudo -s
   cd /home/azureadmin/
   azcopy copy "https://<storageaccount>.blob.core.windows.net/<container>/<BlobDirectoryName>
   <SAStoken>" "/home/azureadmin/storage.tar.gz"
   ```

   Substitute your own storage account and SAS token values. For example:

   ```
   azcopy copy "https://onpremisesstorage.blob.core.windows.net/migration/storage.tar.gz?sv=2019-12-
   12&ss=" "/home/azureadmin/storage.tar.gz"
   ```

2. Extract the compressed file to a directory.

   ```
   cd /home/azureadmin
   tar -zxvf storage.tar.gz
   ```

Import Moodle files to Azure

Once extracted, you can find the `storage` directory under `home/azureadmin`. This `storage` directory contains `moodle`, `moodledata`, and `configuration` directories, and a database backup file. You copy each of these files and directories to the target locations in the following steps:

1. Copy the `moodle` and `moodledata` directories to a shared location under `/moodle`.

   ```
   cp -rf /home/azureadmin/storage/moodle /moodle/html/
   cp -rf /home/azureadmin/storage/moodledata /moodle/moodledata
   ```

Import the Moodle database to Azure

Connect to the Azure Database for MySQL server, and import the on-premises Moodle database archive to Azure Database for MySQL.

Connect to the MySQL server

Azure Database for MySQL instances are protected by a firewall. All connections to the server and the databases inside the server are rejected by default. Before connecting to Azure Database for MySQL for the first time, configure the firewall to allow access for the controller VM's public IP address or IP address range.

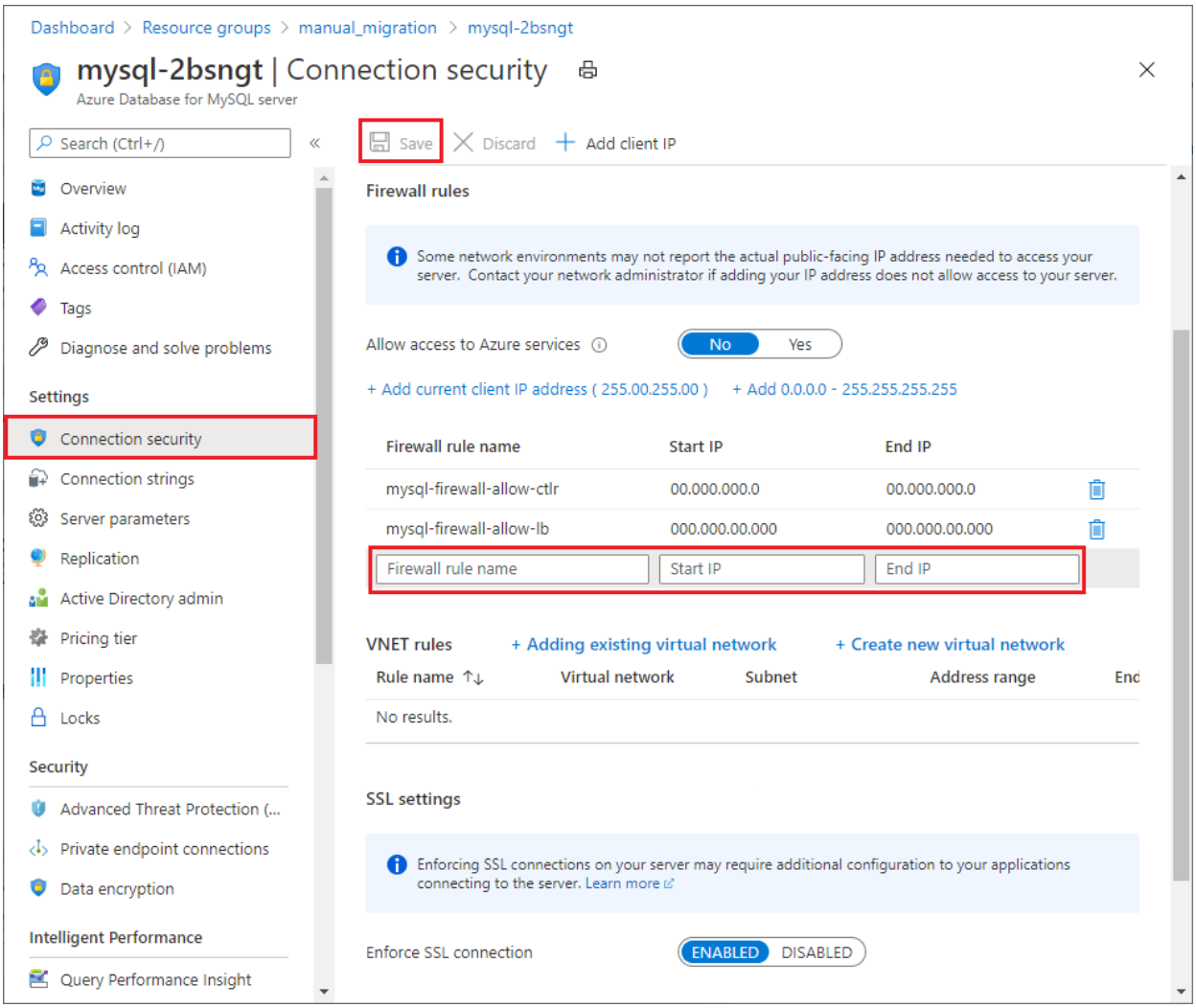
You can configure the firewall by using the Azure command line (Azure CLI) or the Azure portal.

Run the following Azure CLI command, substituting your own values for the placeholders:

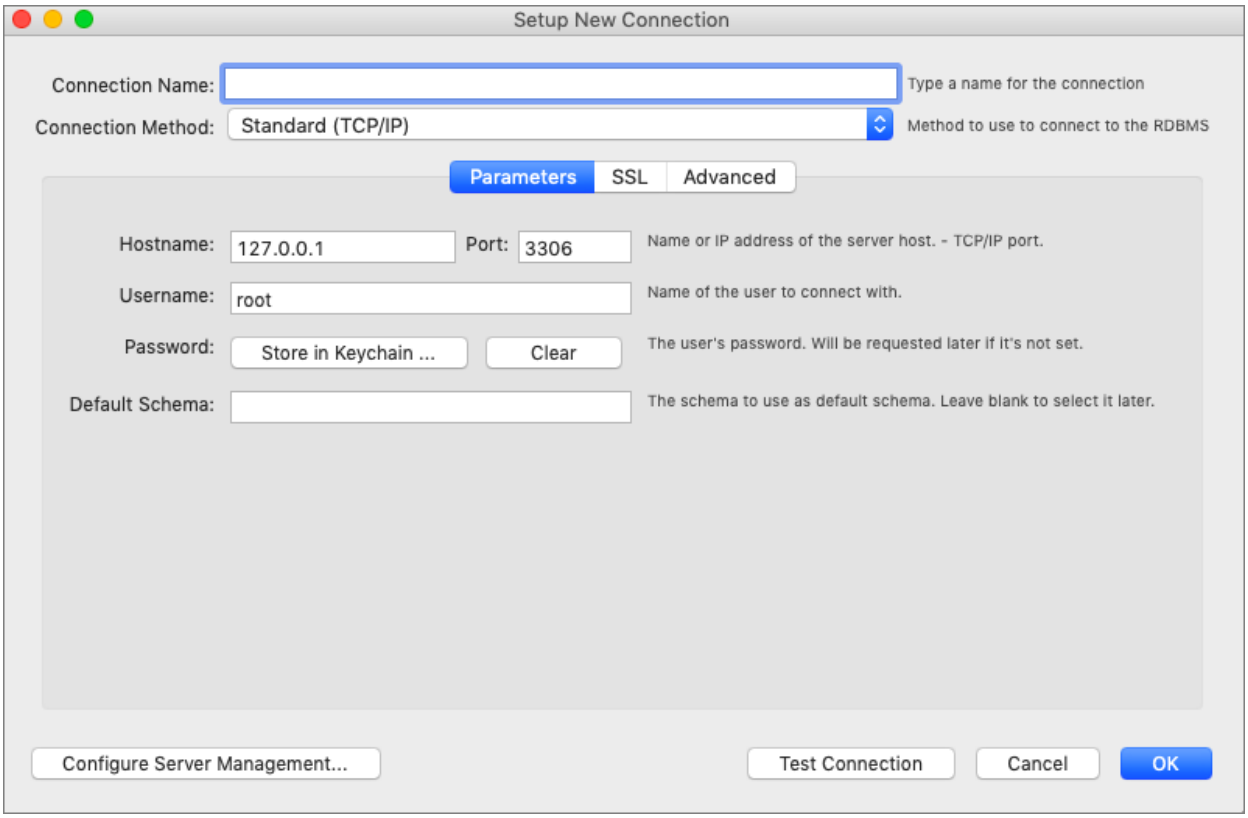
```
az mysql server firewall-rule create --resource-group <myresourcegroup> --server <mydemoserver> --name
<AllowMyIP> --start-ip-address <192.168.0.1> --end-ip-address <192.168.0.1>
```

Or, in the Azure portal, select the Azure Database for MySQL server from your deployed Moodle infrastructure resources. In the server page's left navigation, select **Connection security**.

You can add allowed IP addresses and configure firewall rules here. Select **Save** after you've created the rules.



You can now connect to the MySQL server by using the `mysql` command-line tool or MySQL Workbench.



To get connection information, go to your MySQL server's **Overview** page in the Azure portal. Use the copy icons next to each field to copy the **Server name** and **Server admin login name**.

For example, your server name might be `mydemoserver.mysql.database.azure.com`, and the server admin login name might be `myadmin@mydemoserver`.

You also need the password. If you need to reset the password, select **Reset password** in the menu bar.

Use these database server details in the following sections.

Import the Moodle database to Azure Database for MySQL

1. Create a MySQL database to import the on-premises database into:

```
mysql -h $server_name -u $server_admin_login_name -p$admin_password -e "CREATE DATABASE $moodledbname
CHARACTER SET utf8;"
```

2. Assign the correct permissions to the database:

```
mysql -h $server_name -u $server_admin_login_name -p$admin_password -e "GRANT ALL ON $moodledbname.*
TO '$server_admin_login_name' IDENTIFIED BY '$admin_password';"
```

3. Import the database:

```
mysql -h $server_name -u $server_admin_login_name -p$admin_password $moodledbname <
/home/azureadmin/storage/database.sql
```

Update configurations

After importing the on-premises Moodle database archive to Azure Database for MySQL, update the following configurations on the controller VM as necessary:

- Update the Moodle config file.

- Configure directory permissions.
- Configure PHP and NGINX web servers.
- Update the DNS name and other variables.
- Install any missing PHP extensions.
- Ensure the web server instances on the controller VM are stopped.
- Copy the config files to a shared location for copying to virtual machine scale sets.

Update the Moodle config file

Update the database detail parameters in the Moodle configuration file `/moodle/config.php`.

To get the DNS name for this task:

1. In the Azure portal, select the **Load Balancer public IP address** from your deployed Moodle infrastructure resources.
2. On the **Overview** page, select the copy icon next to the **DNS name**.

To update the `config.php` file:

1. Enter the following commands to edit `config.php` in the `nano` editor:

```
cd /moodle/html/moodle/
nano config.php
```

2. Update the database details in the file, using the values you copied from the Azure portal:

```
$CFG->dbhost    = 'localhost';               // Change 'localhost' to the server name.
$CFG->dbname    = 'moodle';                  // Change 'moodle' to the newly created database name.
$CFG->dbuser    = 'root';                    // Change 'root' to the server admin login name.
$CFG->dbpass    = 'password';                // Change 'password' to the server admin login
password.
$CFG->wwwroot   = 'https://on-premises.com'; // Change 'on-premises' to the DNS name.
$CFG->dataroot  = '/var/moodledata';         // Change the path to '/moodle/moodledata'.
```

3. After making the changes, press Ctrl+O to save the file and Ctrl+X to exit the editor.

You can store the on-premises `dataroot` directory at any location.

Configure directory permissions

- Assign 755 and www-data owner:group permissions to the `moodle` directory.

```
sudo chmod 755 /moodle/html/moodle sudo chown -R www-data:www-data /moodle/html/moodle
```

- Assign 770 and www-data owner:group permissions to the `moodledata` directory.

```
sudo chmod 770 /moodle/moodledata sudo chown -R www-data:www-data /moodle/moodledata
```

Update web config files

Back up and update the NGINX `conf` file:

```
sudo mv /etc/nginx/sites-enabled/*.conf /home/azureadmin/backup/
cd /home/azureadmin/storage/configuration/
sudo cp -rf nginx/sites-enabled/*.conf /etc/nginx/sites-enabled/
```

Back up and update the PHP `www.conf` file:

```
_PHPVER='/usr/bin/php -r "echo PHP_VERSION;" | /usr/bin/cut -c 1,2,3'
echo $_PHPVER
sudo mv /etc/php/$_PHPVER/fpm/pool.d/www.conf /home/azureadmin/backup/www.conf
sudo cp -rf /home/azureadmin/storage/configuration/php/$_PHPVER/fpm/pool.d/www.conf
/etc/php/$_PHPVER/fpm/pool.d/
```

Update the NGINX configuration variables

Update the Azure cloud DNS name to the on-premises Moodle application's DNS name.

1. Open the NGINX configuration file:

   ```
   nano /etc/nginx/sites-enabled/*.conf
   ```

2. The ARM template deployment sets the NGINX server to port 81. Update the `SERVER_PORT` in the file to 81 if it's not 81.

3. Update the `server_name`. For example, for a `server_name on-premises.com`, update `on-premises.com` with the DNS name. In most cases, the DNS name doesn't change in the migration.

4. Update the HTML `root` directory location. For example, update `root /var/www/html/moodle;` to `root /moodle/html/moodle;`.

   The on-premises root directory can be at any location.

5. After making the changes, press Ctrl+O to save the file and Ctrl+X to exit.

Install any missing PHP extensions

The ARM deployment templates install the following PHP extensions:

- `fpm`
- `cli`
- `curl`
- `zip`
- `pear`
- `mbstring`
- `dev`
- `mcrypt`
- `soap`
- `json`
- `redis`
- `bcmath`
- `gd`
- `mysql`
- `xmlrpc`
- `intl`
- `xml`
- `bz2`

If the on-premises Moodle application has any PHP extensions that aren't on the controller VM, you can install them manually.

To get the list of PHP extensions in the on-premises application, run:

```
php -m
```

To install missing extensions, run:

```
sudo apt-get install -y php-<extension>
```

Ensure the web server instances on the controller VM are stopped

1. Restart the web servers.

   ```
   sudo systemctl restart nginx
   sudo systemctl restart php$_PHPVER-fpm
   ```

2. Stop the web servers.

   ```
   sudo systemctl stop nginx
   sudo systemctl stop php$_PHPVER-fpm
   ```

When a request reaches Azure Load Balancer, it now redirects to virtual machine scale set instances and not to the controller VM.

Copy configuration files

Copy PHP and web server configuration files to a shared location for later copying to virtual machine scale set instances.

To create the directory for configuration files in a shared location, run:

```
mkdir -p /moodle/config
mkdir -p /moodle/config/php
mkdir -p /moodle/config/nginx
```

To copy the PHP and web server configuration files to the shared directory, run:

```
cp /etc/nginx/sites-enabled/* /moodle/config/nginx
cp /etc/php/$_PHPVER/fpm/pool.d/www.conf /moodle/config/php
```

Next steps

Continue to Set up the Moodle controller instance and worker nodes for the next steps in the Moodle migration process.

How to set up Moodle worker nodes

5/18/2021 • 2 minutes to read • Edit Online

Follow these steps to configure a virtual machine scale set, or worker nodes, for Moodle.

Virtual machine scale set instances

A virtual machine scale set instance is assigned a private IP address. You can only access this IP address with a controller virtual machine that is in the same virtual network as the IP address. This article describes how to set up that IP address and then configure the Azure virtual machine scale set that your Moodle migration creates.

Access the virtual machine scale set

Follow these steps to access the virtual machine scale set:

1. Determine the private IP address that Azure uses for your virtual machine scale set instance:
   a. Sign in to the Azure portal, and locate the resource group that deployment created.
   b. Open the page for the virtual machine scale set resource.
   c. In the left panel, select **Instances**.
   d. Open the running instance. In the **Overview** section, copy the private IP address that is associated with that instance.
2. Enter these commands to sign in to the virtual machine scale set from the controller virtual machine:

   ```
   sudo -s
   sudo ssh azureadmin@<private IP address>
   ```

   In the command, `<private IP address>` is the private IP address of the virtual machine scale set. For example, enter:

   ```
   sudo -s
   sudo ssh azureadmin@172.31.X.X
   ```

Create a backup directory

An earlier step of the migration process extracted backup files to a directory named `storage` in `/home/azureadmin`. This `storage` directory contains the `moodle` and `moodledata` directories, a configuration directory, and a database backup file. After signing in to your scale set virtual machine instance, enter these commands to create a backup directory for these files:

```
cd /home/azureadmin/
mkdir -p backup
mkdir -p backup/moodle
```

Configure the PHP and web server

To configure the PHP and web server, take these steps:

1. Set the PHP version to a variable:

```
_PHPVER=`/usr/bin/php -r "echo PHP_VERSION;" | /usr/bin/cut -c 1,2,3`
echo $_PHPVER
```

2. Create a backup of the PHP and web server configurations:

```
sudo mv /etc/nginx/sites-enabled/*.conf /home/azureadmin/backup/
sudo mv /etc/php/$_PHPVER/fpm/pool.d/www.conf /home/azureadmin/backup/www.conf
```

3. Copy the PHP and web server configuration files:

```
sudo cp /moodle/config/nginx/*.conf  /etc/nginx/sites-enabled/
sudo cp /moodle/config/php/www.conf /etc/php/$_PHPVER/fpm/pool.d/
```

Install missing extensions

Take these steps to install missing extensions:

1. To obtain a list of PHP extensions that are installed on-premises, enter the following command on an on-premises virtual machine:

```
php -m
```

2. Use an Azure Resource Manager template to install the following PHP extensions:

- `fpm`
- `cli`
- `curl`
- `zip`
- `pear`
- `mbstring`
- `dev`
- `mcrypt`
- `soap`
- `json`
- `redis`
- `bcmath`
- `gd`
- `mysql`
- `xmlrpc`
- `intl`
- `xml`
- `bz2`

3. If the on-premises Moodle application has any additional PHP extensions that aren't in the controller virtual machine, install them manually with this command:

```
sudo apt-get install -y php-<extension name>
```

Next steps

Continue to How to follow up after a Moodle migration.

How to follow up after a Moodle migration

5/18/2021 • 8 minutes to read • Edit Online

Post-migration tasks

After migrating Moodle, you need to take care of some post-migration tasks to complete your configuration. These tasks include:

- Updating log paths in virtual machine scale set instances.
- Restarting servers in virtual machine scale set instances.
- Updating certificates.
- Updating certificate locations.
- Updating the HTML local copy.
- Restarting the PHP and NGINX servers.
- Mapping the DNS name to the Azure Load Balancer IP address.

Controller virtual machine scale set

Take the following steps to finish configuring your virtual machine scale set. You will need to SSH to your virtual machine scale set instance by using the private IP address, as described in Access the virtual machine scale set.

Update log paths

Your on-premises environment and Azure might store log files in different locations. For example, you might need to update these log paths:

- `/var/log/syslogs/moodle/access.log`
- `/var/log/syslogs/moodle/error.log`

Follow these steps to update the log file locations:

1. Enter this command to open the configuration file:

   ```
   nano /etc/nginx/nginx.conf
   ```

2. Find `access_log` and `error_log`, and update the log paths.

3. Press Ctrl+O to save the changes and Ctrl+X to close the file.

Restart servers

Enter these commands to restart the `nginx` and `php-fpm` servers:

```
sudo systemctl restart nginx
sudo systemctl restart php<php version>-fpm
```

Controller virtual machine

Take these steps to complete the controller virtual machine configuration.

Update security certificates

1. Sign in to the controller virtual machine. You can find the certificates for your Moodle application in the

`/moodle/certs` folder.

2. Copy the `.crt` and `.key` files to `/moodle/certs/`. Change the file names to `nginx.crt` and `nginx.key`, respectively, so that the configured NGINX servers recognize them. If your local environment supports the SCP utility or a tool like WinSCP, you can use these tools to copy these files to the controller virtual machine. Otherwise, use these commands:

   ```
   cd /<path to certs location>
   mv /<path to certs location>/*.key /moodle/certs/nginx.key
   mv /<path to certs location>/*.crt /moodle/certs/nginx.crt
   ```

   As an alternative to copying the files, use these commands to generate a self-signed certificate:

   ```
   openssl req -x509 -nodes -days 365 -newkey rsa:2048 \
   -keyout /moodle/certs/nginx.key \
   -out /moodle/certs/nginx.crt \
   -subj "/C=US/ST=WA/L=Redmond/O=IT/CN=mydomain.com"
   ```

   You can only use a self-signed certificate for testing.

3. It's recommended that the certificate files are owned by `www-data:www-data` and are read-only to the owner. Enter these commands to make those changes:

   ```
   chown www-data:www-data /moodle/certs/nginx.*
   chmod 400 /moodle/certs/nginx.*
   ```

4. Update the certificate location by taking these steps:

   a. Enter this command to open the configuration file:

      ```
      nano /etc/nginx/sites-enabled/*.conf
      ```

   b. Find `ssl_certificate` in the file.

   c. Replace the certificate's paths with these values:

      ```
      /moodle/certs/moodle/certs/nginx.crt;
      /moodle/certs/nginx.key;
      ```

   d. Press Ctrl+O to save your changes and Ctrl+X to close the file.

Update local HTML copy

The local copy of the Moodle HTML site content, `/moodle/html/moodle`, is created in the virtual machine scale set in this folder: `/var/www/html/moodle`. The local copy is updated only when the time stamp changes. Enter this command in the controller virtual machine to update the time stamp:

```
sudo -s
/usr/local/bin/update_last_modified_time.moodle_on_azure.sh
```

The last-modified time stamp file, `/moodle/html/moodle/last_modified_time.moodle_on_azure`, contains a script. Every time that the script runs, the `/moodle/html/moodle` directory content is updated in the local copy, `/var/www/html`.

Restart servers

Enter these commands to restart the `nginx` and `php-fpm` servers:

```
sudo systemctl restart nginx
sudo systemctl restart php<php version>-fpm
```

Map the DNS name to the Azure Load Balancer IP address

Follow these steps at the hosting-provider level to map the DNS name to the Azure Load Balancer IP:

1. Enter the following command in the controller virtual machine to turn off maintenance mode on the Moodle website:

   ```
   sudo /usr/bin/php admin/cli/maintenance.php --disable
   ```

2. Enter this command to check the status of the Moodle site:

   ```
   sudo /usr/bin/php admin/cli/maintenance.php
   ```

3. Go to the DNS name to see the migrated Moodle web page.

Frequently asked questions and troubleshooting

Refer to the following information when you have questions about your Moodle migration. These log files can also help you troubleshoot problems:

- Syslog files:
  - Whenever a user goes to your webpage, the system generates either an error log or an access log.
  - You can find them in this folder: `/var/log/nginx/`.
- Cron log file:
  - When a cron job runs, it updates the local copy of the log file.
  - You can find the file in this folder: `/var/log/sitelogs/moodle/cron.log`.

Database connection failure

For errors like *database connection failed* or *could not connect to the database you specified*, here are some potential reasons and solutions:

- Your database server isn't installed or running. To check for this condition in MySQL, enter the following command:

  ```
  $telnet database_host_name 3306
  ```

  If your database is running, you should get a response that includes the version number of the MySQL server.

- The host address is configured incorrectly. If you're running two instances of Moodle on different ports, use the IP address of the host, not `localhost`, in the `$CFG->dbhost` setting. For example, use:

  `$CFG->dbhost = 127.0.0.1:3308`

- You haven't created a Moodle database. Or you haven't assigned adequate permissions for accessing the database. Check the database and the permissions that you granted.

- The Moodle database settings aren't correct. For instance, the database name, username, or password in

your Moodle configuration file, `config.php`, isn't correct. Make sure your MySQL username and password don't contain apostrophes or non-alphanumeric characters.

Internal server error

There are several possible causes for this error: *500: Internal Server Error*. Start by checking your web server error log, which should contain a detailed explanation. Here are some possibilities:

- There's a syntax error in your `.htaccess` or `httpd.conf` file. The correct syntax for directives differs depending on which file you're using. Use the following command to test for configuration errors in your NGINX files:

  ```
  `nginx -t`
  ```

- The web server runs under your own user name, and the access permissions are incorrect. In this case, all files need a maximum permission level of 755. Check that this level is set for your Moodle directory in your control panel. Or use this command to set the level if you have access to the shell:

  ```
  chmod -R 755 moodle
  ```

Memory limit error

When a *403: Forbidden* error occurs, the PHP `memory_limit` value isn't large enough for the PHP script. The `memory_limit` value is the allowed memory size. Increase the PHP `memory_limit` value by small amounts until the message disappears. Use one of these methods:

- For a hosted installation, ask your host's support how to increase the value. Many environments use `.htaccess` files. If your installation does, add the following line to your `.htaccess` file:

  ```
  php_value memory_limit <value>M
  ```

  For example, to increase the value to 40 megabytes, enter:

  ```
  php_value memory_limit 40M
  ```

  If no `.htaccess` file exists, create one in the Moodle directory with that line.

- If you have your own server with shell access, edit your `php.ini` file. Then restart your web server to apply the changes that you made in `php.ini`. To make sure you've updated the value correctly, enter this command:

  ```
  `phpinfo`
  ```

  The output from `phpinfo` should contain a line similar to this one:

  ```
  memory_limit <value>M
  ```

  For example, it might contain this line:

  ```
  memory_limit 40M
  ```

  An alternative to increasing the `memory_limit` value is to turn off the memory limit by entering this command:

```
memory_limit 0
```

Sign-in errors

Sometimes you can't sign in, or you see one of these messages:

- `Your session has timed out. Please log in again.`
- `A server error that affects your login session was detected. Please log in again or restart your browser.`

There might be a problem with your authentication method, especially if you use an external method like LDAP to authenticate users. Try to sign in to another manual account, such as your main admin account. If you can't sign in, check your authentication. If you can sign in to the other account, here are possible reasons and solutions for the Moodle sign-in problem:

- Your hard disk might be full. In this situation, Moodle can't create new sessions, and users can't sign in. Check that your hard disk isn't full, that your server is on shared hosting, and that you haven't reached your disk space quota.
- The web server can't write to the `sessions` subdirectory. Carefully check the permissions in your `moodledata` area.

Fatal errors

The Moodle and moodledata permissions might be incorrect if you see this error:
`fatal error: $cfg->dataroot is not writable. The admin has to fix directory permissions! Exiting.`

Check that these permissions are `www-data:www-data` only. If the permissions are at a different level, use this command to change the group and ownership permissions:

```
sudo chown -R /moodle/moodledata
```

Top-level course errors

If you can't find a top-level course right after you install Moodle, the installation probably didn't finish. Right before the end of a complete installation, Moodle asks you for the administrator profile and prompts you to name the site. If these steps were missing, check your logs for errors. Then restart the database. If you used the web-based installer, install Moodle again by using the command line.

Navigation errors

If you can't navigate freely in Moodle after you sign in, your URL configuration might be incorrect. Make sure that the URL in your `$CFG->wwwroot` setting is the same one that you use to access the site.

File upload errors

If you see a *File not found* error when uploading a file, your web server might not have turned on slash arguments.

- If your web server supports slash arguments, turn them on.
- If your web server doesn't support slash arguments, turn them off in Moodle by clearing the **Use slash arguments** check box in **Administration** > **Site administration** > **Server** > **HTTP**. You may see this message.

> **WARNING**
> Disabling slash arguments will result in SCORM packages not working and slash arguments warnings being displayed!

Maintenance mode errors

When Moodle is in maintenance mode and you try to leave that mode, sometimes you see this message: *This site is undergoing maintenance and is currently unavailable*. This situation comes up when there's a problem with the `maintenance.html` file that Moodle creates in the `moodledata` folder when it goes into maintenance mode. In this case, take these steps:

- Check that the web server user has write permissions in the `moodledata` directory.
- Manually delete the `maintenance.html` file.

Next steps

- Azure Database for MySQL documentation
- What are virtual machine scale sets?
- Storage account overview

Rehost an on-premises dev/test environment on Azure Virtual Machines via Azure Migrate

5/4/2021 • 14 minutes to read • Edit Online

This article demonstrates how the fictional company Contoso rehosts its dev/test environment for two applications running on VMware virtual machines (VMs) by migrating to Azure Virtual Machines.

The SmartHotel360 and osTicket applications used in this example are open source. You can download them for your own testing purposes.

Migration options

Contoso has several options available for moving dev/test environments to Azure:

MIGRATION OPTIONS	OUTCOME
Azure Migrate	Assess and migrate on-premises VMs. Run dev/test servers by using Azure infrastructure as a service (IaaS). Manage VMs with Azure Resource Manager.
Azure DevTest Labs	Quickly provision development and test environments. Minimize waste with quotas and policies. Set automated shutdowns to minimize costs. Build Windows and Linux environments.

NOTE

Read how Contoso moved its dev/test environment to Azure by using DevTest Labs.

Business drivers

The development leadership team has outlined what it wants to achieve with this migration. It aims to quickly move dev/test capabilities out of an on-premises datacenter and no longer purchase hardware to develop software. It also seeks to empower developers to create and run their environments without involvement from IT.

NOTE

Contoso will use the Pay-As-You-Go Dev/Test subscription offer for its environments. Each active Visual Studio subscriber on the team can use the Microsoft software included with the subscription virtual machines for dev/test at no extra charge. Contoso will just pay the Linux rate for VMs that it runs. That includes VMs with SQL Server, SharePoint Server, or other software that's normally billed at a higher rate.

Migration goals

The Contoso development team has pinned down goals for this migration. These goals are used to determine the best migration method:

- Contoso wants to quickly move out of its on-premises dev/test environments.
- After migration, Contoso's dev/test environment in Azure should have enhanced capabilities over the current system in VMware.
- The operations model will move from IT provisioned to DevOps with self-service provisioning.

Solution design

After pinning down goals and requirements, Contoso designs and reviews a deployment solution and identifies the migration process. The process includes the Azure services that Contoso will use for the migration.

Current application

- The dev/test VMs for the two applications are running on VMs (`WEBVMDEV` , `SQLVMDEV` , `OSTICKETWEBDEV` , `OSTICKETMYSQLDEV`). These VMs are used for development before code is promoted to the production VMs.
- The VMs are located on VMware ESXi host `contosohost1.contoso.com` (version 6.5).
- The VMware environment is managed by vCenter Server 6.5 (`vcenter.contoso.com`), running on a VM.
- Contoso has an on-premises datacenter (`contoso-datacenter`) with an on-premises domain controller (`contosodc1`).

Proposed architecture

- Because the VMs are used for dev/test, they'll reside in the `ContosoDevRG` resource group in Azure.
- The VMs will be migrated to the primary Azure region (`East US 2`) and placed in the development virtual network (`VNET-DEV-EUS2`).
- The web front-end VMs will reside in the front-end subnet (`DEV-FE-EUS2`) in the development network.
- The database VM will reside in the database subnet (`DEV-DB-EUS2`) in the development network.
- The on-premises VMs in the Contoso datacenter will be decommissioned after the migration is done.

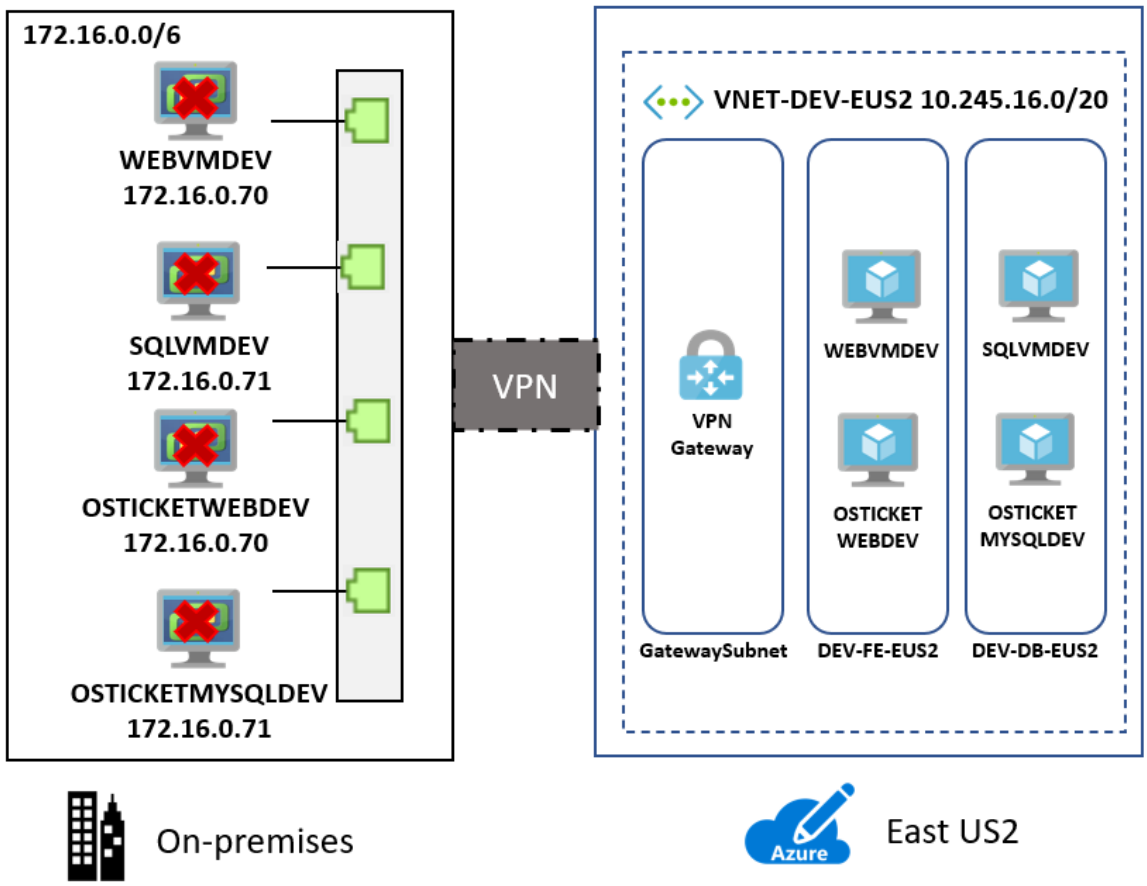


Figure 1: Proposed architecture.

Database considerations

To support ongoing development, Contoso has decided to continue using existing VMs and migrate them to Azure. In the future, Contoso will pursue the use of platform as a service (PaaS) services such as Azure SQL Database and Azure Database for MySQL.

- Database VMs will be migrated as is without changes.
- With the use of the Azure Dev/Test subscription offer, the computers running Windows Server and SQL Server will not incur licensing fees. Avoiding fees will keep the compute costs to a minimum.
- In the future, Contoso will look to integrate its development with PaaS services.

Solution review

Contoso evaluates the proposed design by putting together a list of pros and cons.

CONSIDERATION	DETAILS
Pros	All of the development VMs will be moved to Azure without changes, making the migration simple. Because Contoso is using a lift-and-shift approach for both sets of VMs, special configuration or migration tools aren't needed for the application database. Contoso can take advantage of its investment in the Azure Dev/Test subscription to save on licensing fees. Contoso will retain full control of the application VMs in Azure. Developers will be provided with rights to the subscription, which empowers them to create new resources without waiting for IT to respond to their requests.

CONSIDERATION	DETAILS
Cons	The migration will only move the VMs, not yet moving to PaaS services for their development. This means that Contoso will need have to start supporting the operations of its VMs, including security patches. This was maintained by IT in the past, so Contoso will need to find a solution for this new operational task. The cloud-based solution empowers the developers and doesn't have safeguards for overprovisioning systems. Developers will be able to instantly provision their systems, but they might create resources that cost money and aren't included in the budget.

> **NOTE**
>
> Contoso could address the cons in its list by using DevTest Labs.

Migration process

Contoso will migrate its development front end and database to Azure VMs by using the agentless method in the Azure Migrate: Server Migration tool.

- Contoso prepares and sets up Azure components for Azure Migrate: Server Migration, and prepares the on-premises VMware infrastructure.
- The Azure infrastructure is in place, so Contoso just needs to configure the replication of the VMs through the Azure Migrate: Server Migration tool.
- With everything prepared, Contoso can start replicating the VMs.
- After replication is enabled and working, Contoso migrates the VMs by testing the migration and if successful, failing it over to Azure.
- After the development VMs are up and running in Azure, Contoso will reconfigure its development workstations to point at the VMs now running in Azure.

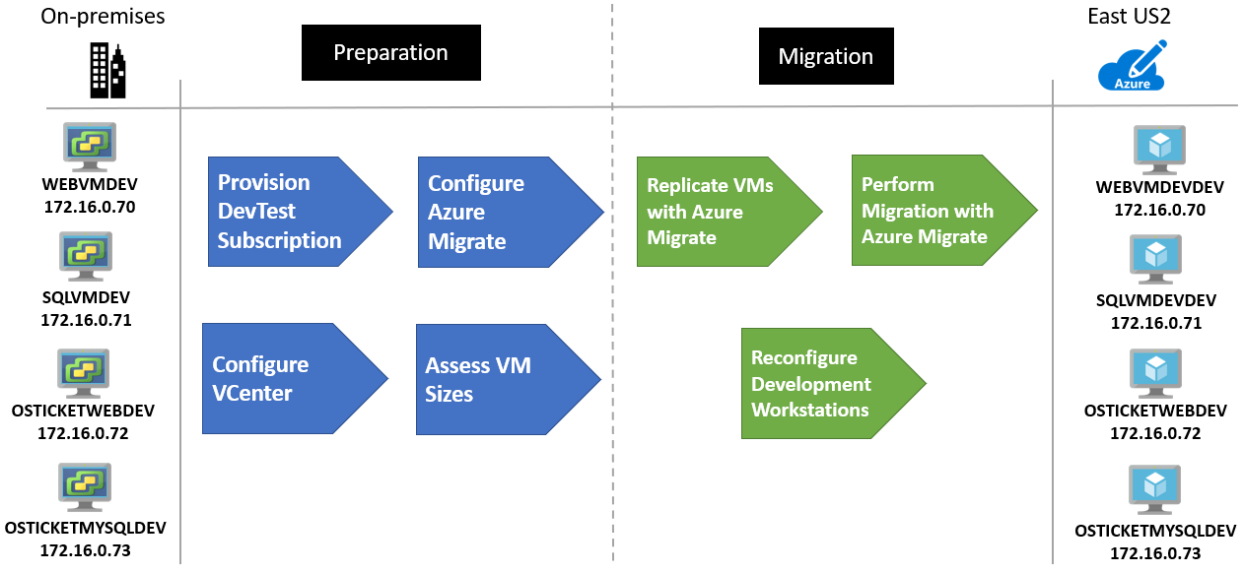


Figure 2: An overview of the migration process.

Azure services

SERVICE	DESCRIPTION	COST
Azure Migrate: Server Migration	The service orchestrates and manages migrating on-premises applications and workloads and AWS or GCP VM instances.	During replication to Azure, Azure Storage charges are incurred. Azure VMs are created and incur charges when the migration occurs and the VMs are running in Azure. Learn more about charges and pricing.

Prerequisites

This is what Contoso needs to run this scenario:

REQUIREMENTS	DETAILS
Azure Dev/Test subscription	Contoso creates an Azure Dev/Test subscription to take advantage of reducing costs up to 80 percent. If you don't have an Azure subscription, create a free account. If you create a free account, you're the admin of your subscription, and you can perform all actions. If you use an existing subscription but you're not the admin, work with the admin to assign you Owner or Contributor permissions. If you need more granular permissions, see Manage Site Recovery access with Azure role-based access control (Azure RBAC).
Azure infrastructure	Learn how Contoso set up an Azure infrastructure. Learn more about specific prerequisites for Azure Migrate: Server Migration.
On-premises servers	On-premises vCenter servers should be running version 5.5, 6.0, 6.5, or 6.7. ESXi hosts should run version 5.5, 6.0, 6.5, or 6.7. One or more VMware VMs should be running on the ESXi host.

Scenario steps

Here's how Contoso admins will run the migration:

- **Step 1: Prepare Azure for Azure Migrate: Server Migration.** They add the server migration tool to their Azure Migrate project.
- **Step 2: Prepare on-premises VMware for Azure Migrate: Server Migration.** They prepare accounts for VM discovery and prepare to connect to Azure VMs after migration.
- **Step 3: Replicate VMs.** They set up replication and start replicating VMs to Azure Storage.
- **Step 4: Migrate the VMs with Azure Migrate: Server Migration.** They run a test migration to make sure everything's working and then run a full migration to move the VMs to Azure.

Step 1: Prepare Azure for the Azure Migrate: Server Migration tool

Contoso needs to migrate the VMs to a virtual network where the Azure VMs will reside when they're created, provisioned, and configured through the Azure Migrate: Server Migration tool.

1. Set up a network: Contoso already set up a network that can be for Azure Migrate: Server Migration when it deployed the Azure infrastructure.
   - The VMs to be migrated are used for development. They will migrate to the Azure development virtual network (`VNET-DEV-EUS2`) in the primary `East US 2` region.
   - Both VMs will be placed in the `ContosoDevRG` resource group, which is used for development resources.
   - The application front-end VMs (`WEBVMDEV` and `OSTICKETWEBDEV`) will migrate to the front-end subnet (`DEV-FE-EUS2`), in the development virtual network.
   - The application database VM (`SQLVMDEV` and `OSTICKETMYSQLDEV`) will migrate to the database subnet (`DEV-DB-EUS2`), in the development virtual network.
2. Provision the Azure Migrate: Server Migration tool.
   a. From Azure Migrate, download the .OVA image and import it into VMware.

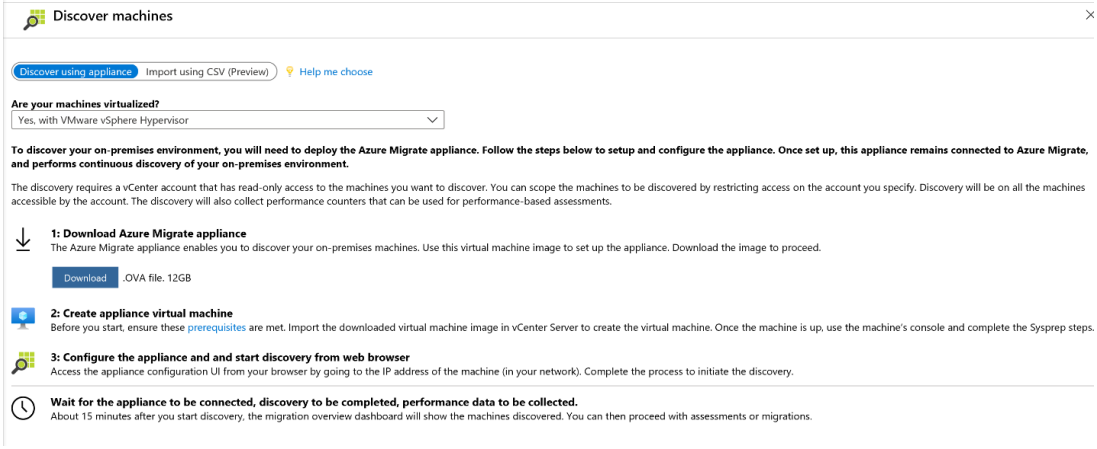


Figure 3: Download the .OVA file.

   b. Start the imported image and configure the tool, including the following steps:
      - Set up the prerequisites.

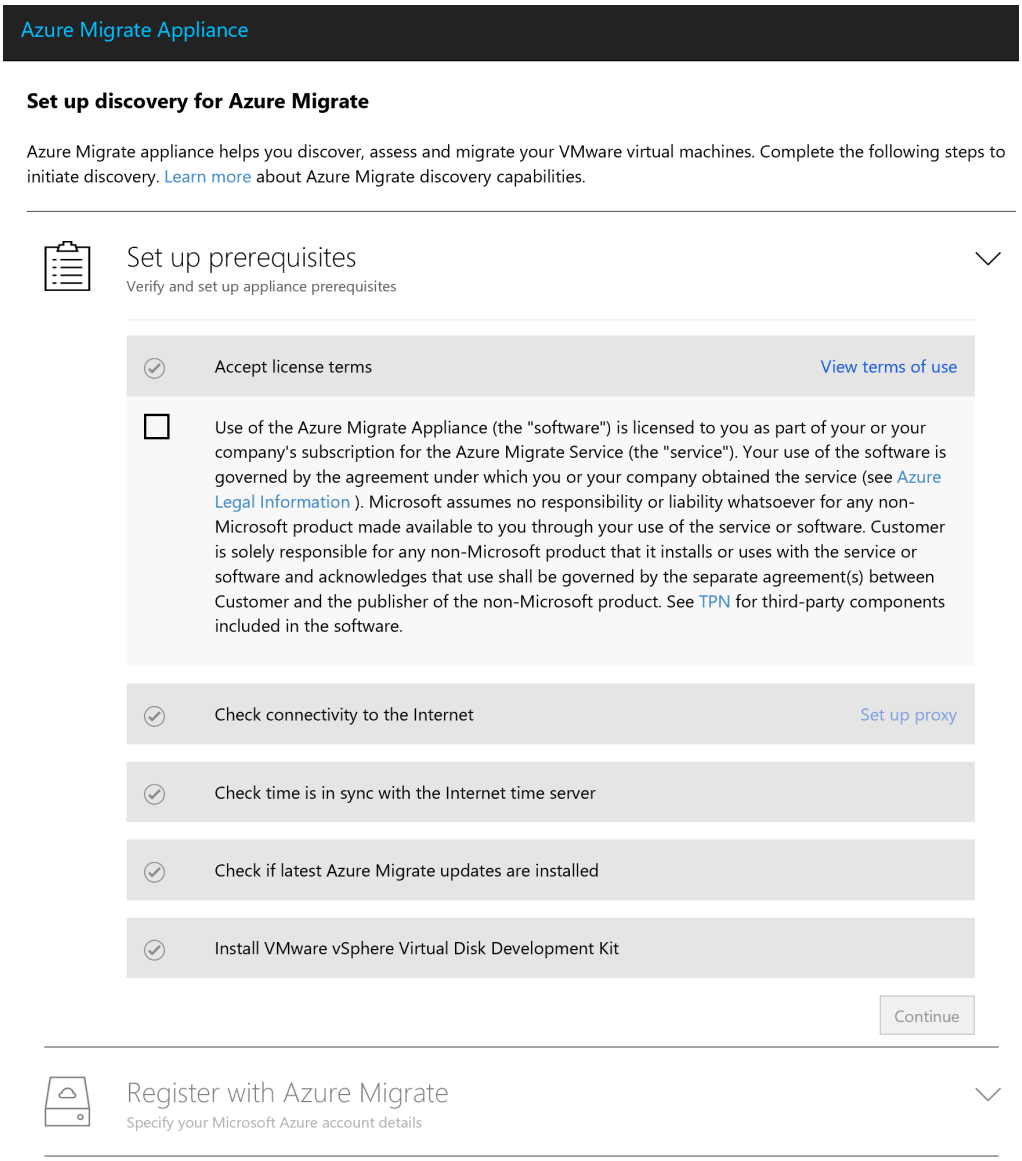


Figure 4: Setting up the prerequisites.

- Point the tool to the Azure subscription.

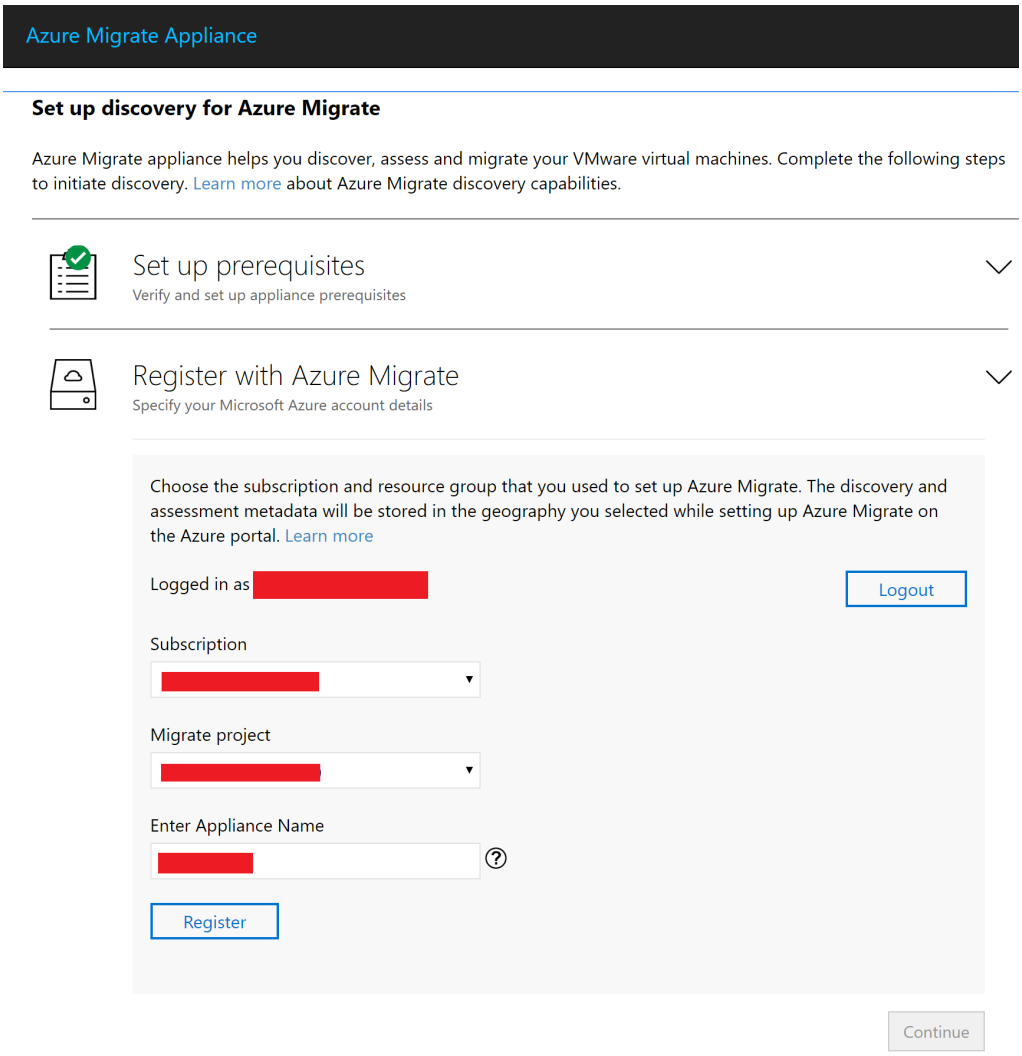


Figure 5: The Azure subscription.

- Set the VMware vCenter credentials.

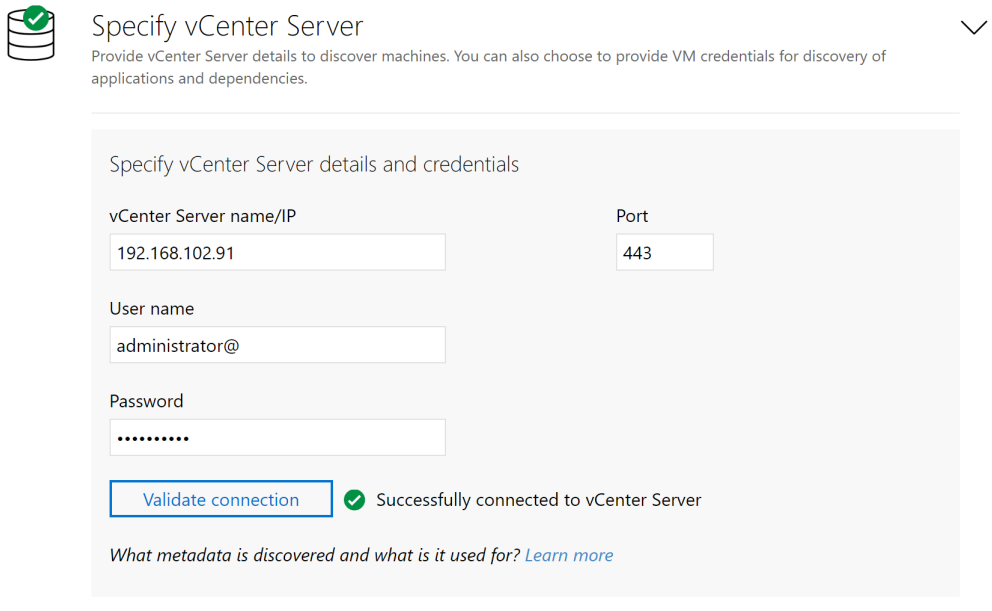


Figure 6: Setting the VMware vCenter credentials.

- Add any Windows-based credentials for discovery.

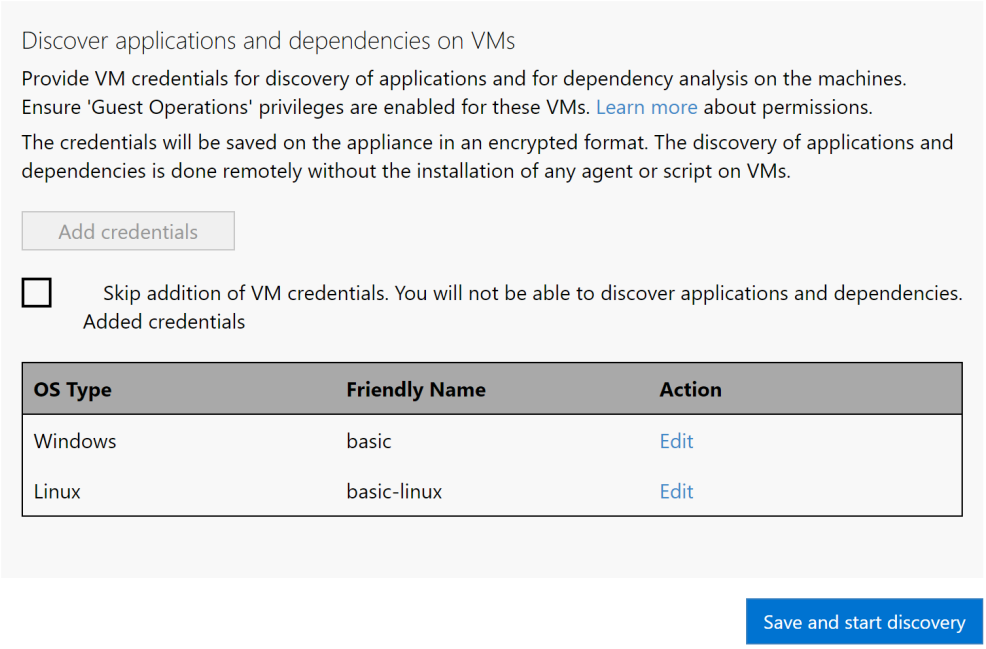


Figure 7: Adding Windows-based credentials for discovery.

3. When you complete the configuration, the tool will take some time to enumerate all the VMs. You'll see them populate the Azure Migrate tool in Azure when this process finishes.

Need more help?

Learn how to set up the Azure Migrate: Server Migration tool.

Prepare on-premises VMs

After migration, Contoso wants to connect to the Azure VMs and allow Azure to manage the VMs. To do this, Contoso admins do the following before migration:

1. For access over the internet, they:
   - Enable RDP or SSH on the on-premises VM before migration.
   - Ensure that TCP and UDP rules are added for the `Public` profile.
   - Check that RDP or SSH is allowed in the operating system firewall.
   - Install SSH via the following command: `sudo apt-get ssh install -y`.
2. For access over Site-to-Site VPN, they:
   - Enable RDP or SSH on the on-premises VM before migration.
   - Check that RDP or SSH is allowed in the operating system firewall.
   - For Windows, set the operating system's SAN policy on the on-premises VM to `OnlineAll`.
3. Install the Azure Windows agent and the Azure Linux agent.

For Windows, there shouldn't be Windows updates pending on the VM when you're triggering a migration. If there are, the admins won't be able to sign in to the VM until the updates finish. After migration, the admins can check **Boot diagnostics** to view a screenshot of the VM. If this doesn't work, they should verify that the VM is running and review troubleshooting tips.

Need more help?

Learn how to prepare VMs for migration.

Step 3: Replicate the on-premises VMs

Before Contoso admins can run a migration to Azure, they need to set up and enable replication. With discovery

completed, they can begin replicating VMware VMs to Azure.

1. In the Azure Migrate project, go to **Servers** > **Azure Migrate: Server Migration**. Then select **Replicate**.

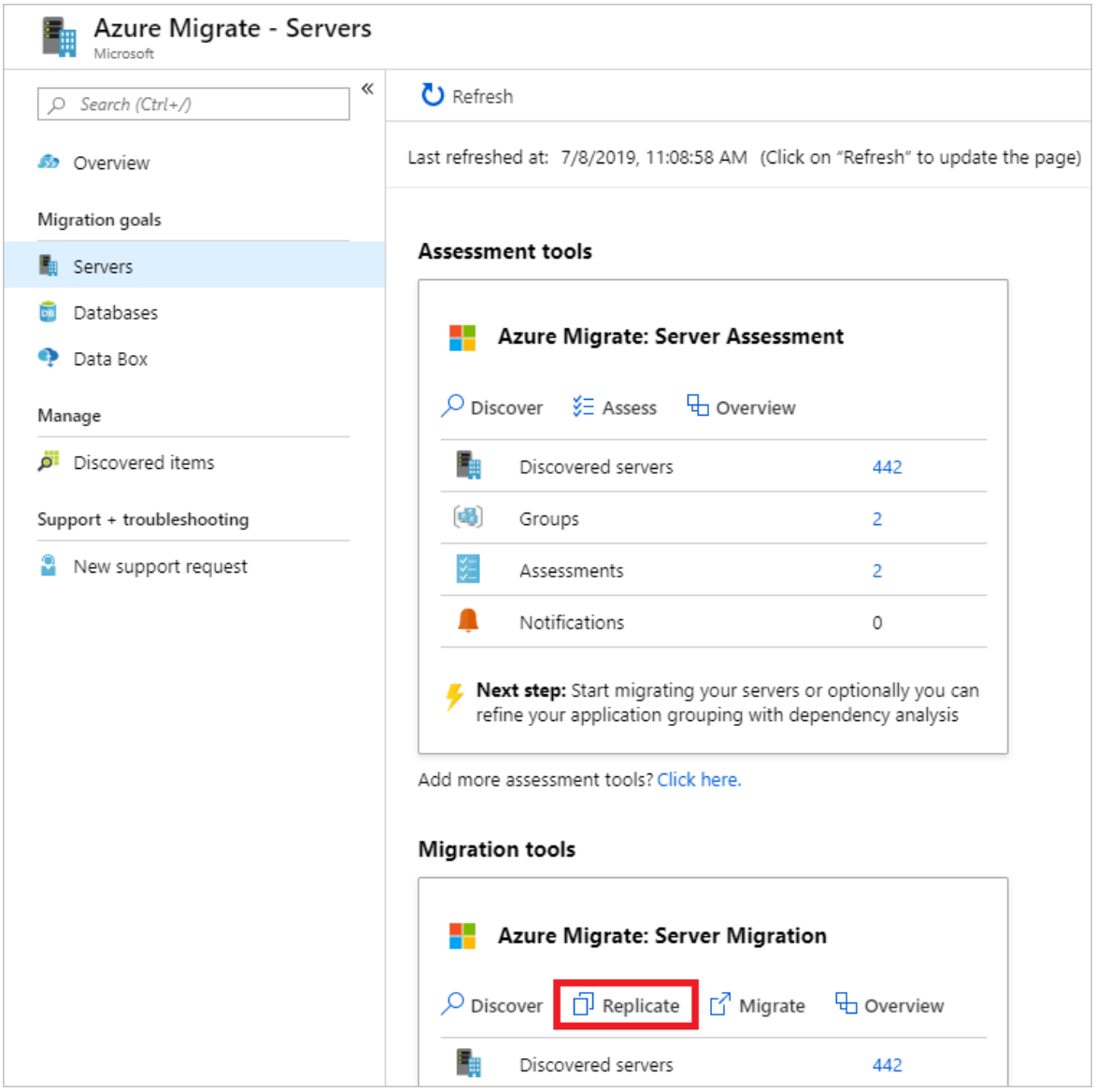


Figure 8: Replicating VMs.

2. In **Replicate** > **Source settings** > **Are your machines virtualized?**, select **Yes, with VMware vSphere**.

3. In **On-premises appliance**, select the name of the Azure Migrate appliance that you set up, and then select **OK**.

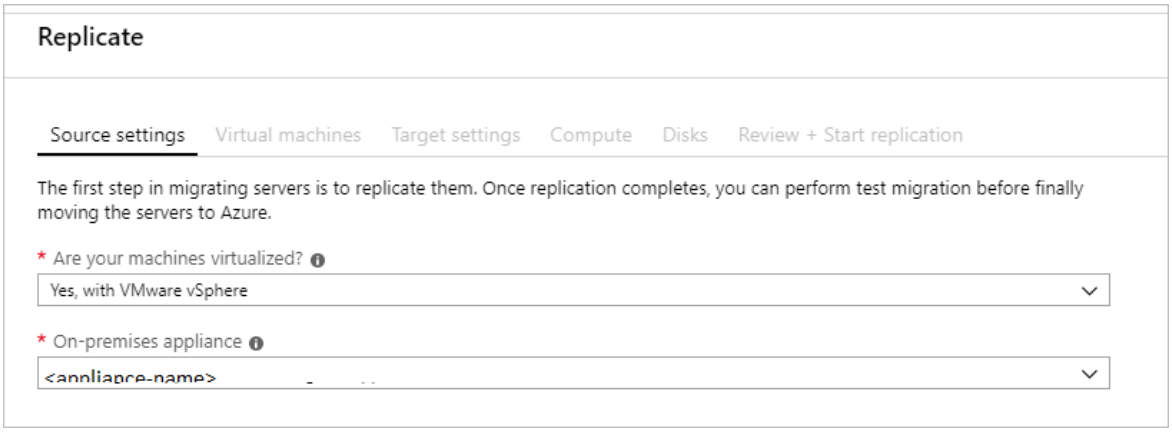


Figure 9: The source settings.

4. In **Virtual machines**, select the machines that you want to replicate.

   - If you've run an assessment for the VMs, you can apply VM sizing and disk type (premium or standard) recommendations from the assessment results. To do this, in **Import migration settings from an Azure Migrate assessment?**, select the **Yes** option.
   - If you didn't run an assessment or you don't want to use the assessment settings, select the **No** option.
   - If you selected to use the assessment, select the VM group and assessment name.

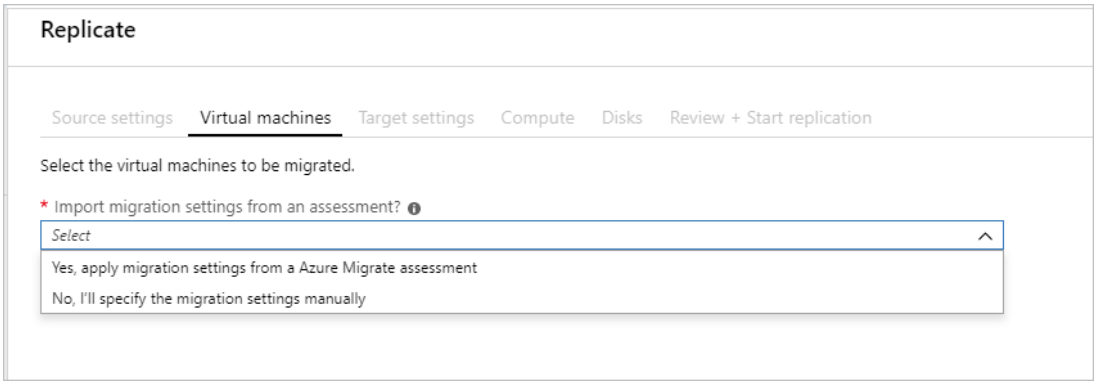


Figure 10: How to set up the prerequisites.

5. In **Virtual machines**, search for VMs as needed and check each VM you want to migrate. Then select **Next: Target settings**.
6. In **Target settings**, select the subscription and target region to which you'll migrate. Then specify the resource group in which the Azure VMs will reside after migration. In **Virtual Network**, select the virtual network or subnet to which the Azure VMs will be joined after migration.
7. In **Azure Hybrid Benefit**, select **No** if you don't want to apply Azure Hybrid Benefit. Then select **Next**. Select **Yes** if you have Windows Server machines that are covered with active Software Assurance or Windows Server subscriptions and you want to apply the benefit to the machines you're migrating. Then select **Next**.

> **NOTE**
> In the case of Contoso, the admins will select **No** to Azure Hybrid Benefit because this is an Azure Dev/Test subscription. This means they'll pay for the compute only. Azure Hybrid Benefit should be used only for production systems that have Software Assurance benefits.

8. In **Compute**, review the VM name, size, OS disk type, and availability set. VMs must conform with Azure requirements.

   - **VM size:** If you're using assessment recommendations, this drop-down list contains the recommended size. Otherwise, Azure Migrate selects a size based on the closest match in the Azure subscription. You can choose a manual size instead in **Azure VM size**.
   - **OS disk:** Specify the OS (boot) disk for the VM. The OS disk has the operating system bootloader and installer.
   - **Availability set:** If the VM should be in an Azure availability set after migration, then specify the set. The set must be in the target resource group that you specify for the migration.

9. In **Disks**, specify whether the VM disks should be replicated to Azure and select the disk type (standard SSD/HDD or premium managed disks) in Azure. Then select **Next**. You can exclude disks from replication. If you do, they won't be present on the Azure VM after migration.
10. In **Review and start replication**, review the settings and select **Replicate** to start the initial replication

for the servers.

> **NOTE**
>
> You can update replication settings at any time before replication starts in **Manage** > **Replicating machines**. Settings can't be changed after replication starts.

Step 4: Migrate the VMs

Contoso admins run a quick test migration and then a full migration to migrate the VMs.

Run a test migration

1. In **Migration goals** > **Servers** > **Azure Migrate: Server Migration**, select **Test migrated servers**.

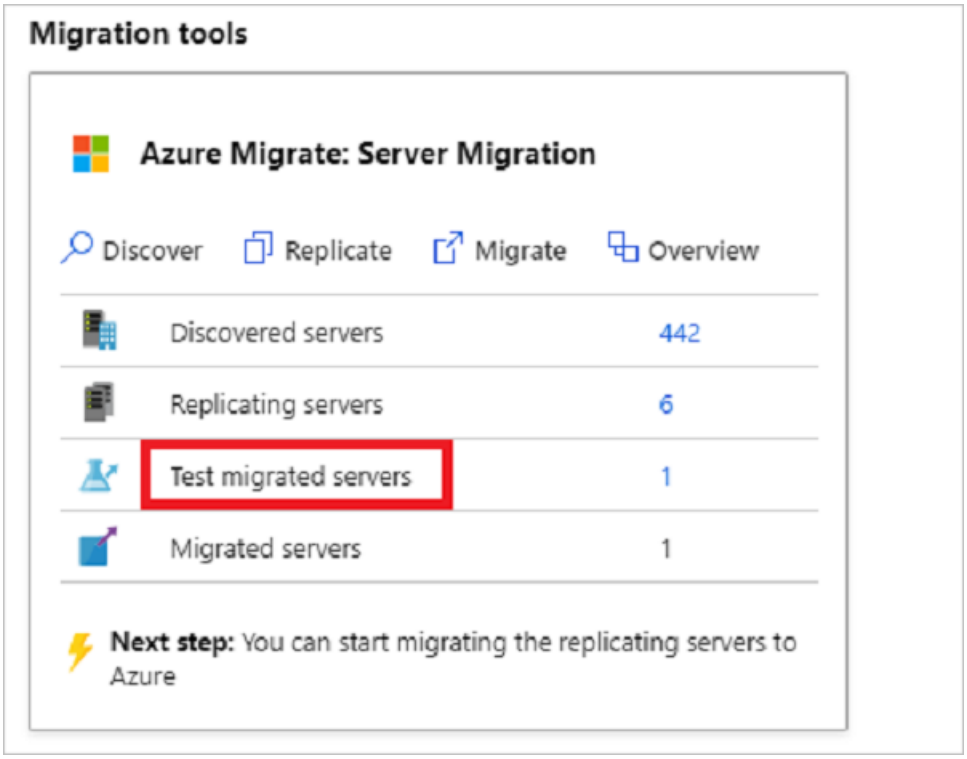


Figure 11: Testing migrated servers.

2. Select and hold (or right-click) the VM to test, and then select **Test migrate**.

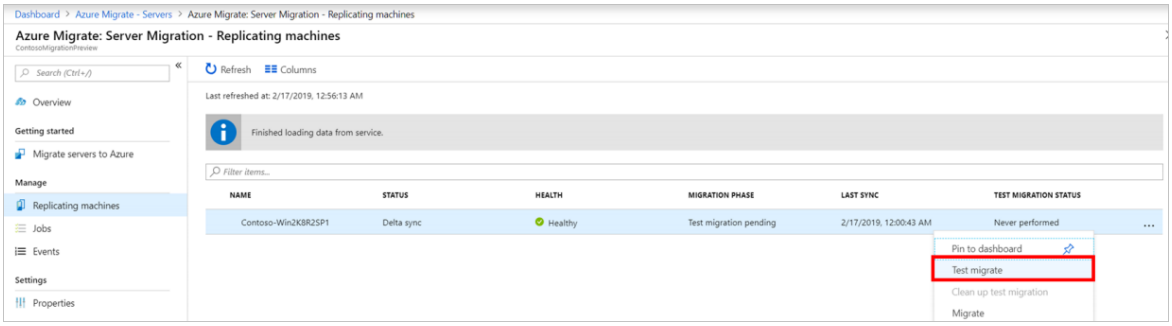


Figure 12: Testing the migration.

3. In **Test Migration**, select the virtual network in which the Azure VM will be located after the migration. We recommend that you use a nonproduction virtual network.
4. The **Test migration** job starts. Monitor the job in the portal notifications.
5. After the migration finishes, view the migrated Azure VM in **Virtual Machines** in the Azure portal. The machine name has a **-Test** suffix.

6. After the test is done, select and hold (or right-click) the Azure VM in **Replicating machines**, and then select **Clean up test migration**.

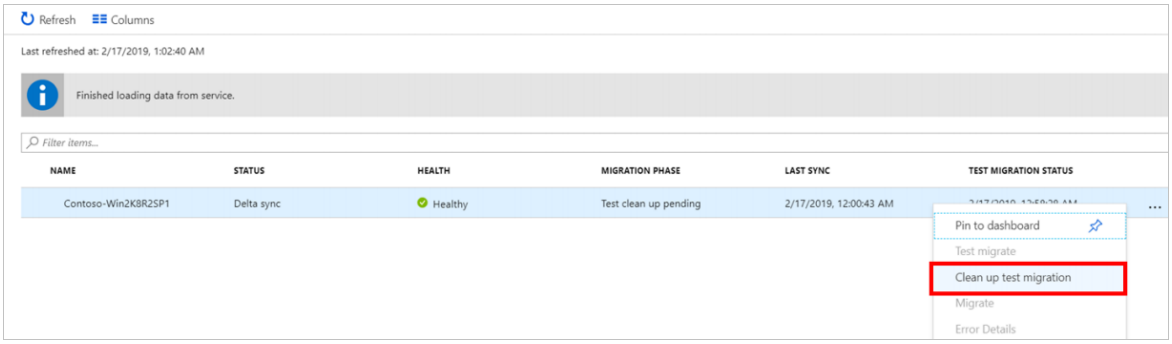


Figure 13: Cleaning up the test migration.

Migrate the VMs

Now Contoso admins run a full migration.

1. In the Azure Migrate project, select **Servers** > **Azure Migrate: Server Migration** > **Replicating servers**.

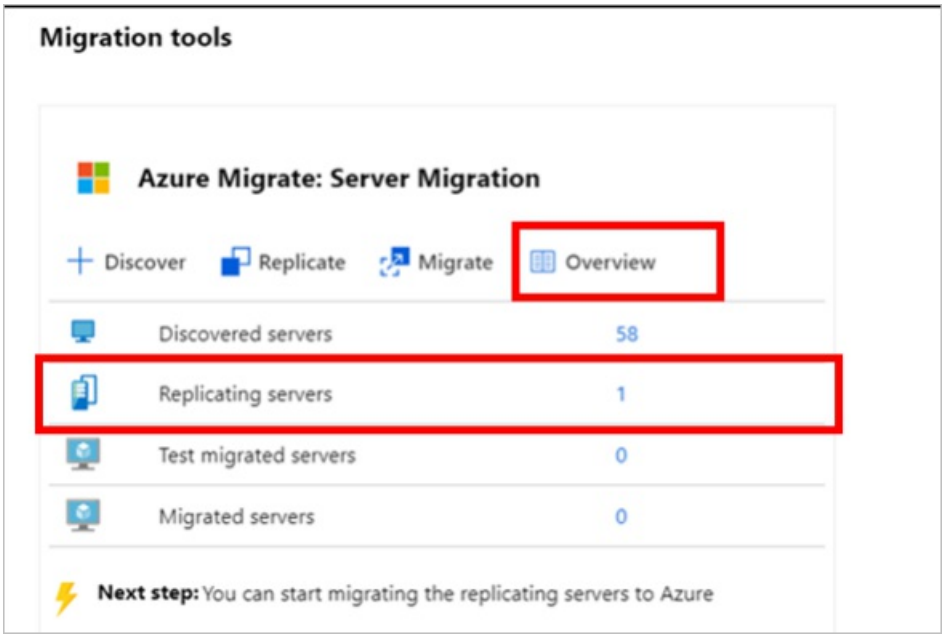


Figure 14: Replicating servers.

2. In **Replicating machines**, select and hold (or right-click) the VM, and then select **Migrate**.
3. In **Migrate** > **Shut down virtual machines and perform a planned migration with no data loss**, select **Yes** > **OK**. By default, Azure Migrate shuts down the on-premises VM and runs an on-demand replication to synchronize any VM changes that occurred since the last replication occurred. This ensures no data loss. If you don't want to shut down the VM, select **No**.
4. A migration job starts for the VM. Track the job in Azure notifications.
5. After the job finishes, you can view and manage the VM from the **Virtual Machines** page.

Need more help?

Learn how to run a test migration and how to migrate VMs to Azure.

Clean up after migration

The development VMs for both the SmartHotel360 and osTicket applications start running on Azure VMs when the migration is complete.

Now, Contoso needs to complete these cleanup steps:

- After the migration is complete, stop replication.
- Remove the `WEBVMDEV`, `SQLVMDEV`, `OSTICKETWEBDEV`, and `OSTICKETMYSQLDEV` VMs from the vCenter inventory.
- Remove all the VMs from local backup jobs.
- Update internal documentation to show the new location and IP addresses for the VMs.
- Review any resources that interact with the VMs, and update any relevant settings or documentation to reflect the new configuration.

Review the deployment

With the application now running, Contoso now needs to fully operationalize and secure it in Azure.

Security

The Contoso security team reviews the Azure VMs to determine any security issues. To control access, the team reviews the network security groups (NSGs) for the VMs. NSGs are used to ensure that only traffic allowed to the application can reach it. The team also considers securing the data on the disk by using Azure Disk Encryption and Azure Key Vault.

For more information, see Security best practices for IaaS workloads in Azure.

Business continuity and disaster recovery

For business continuity and disaster recovery, Contoso takes the following action: keep data safe. Contoso backs up the data on the VMs by using the Azure Backup service. For more information, see An overview of Azure VM backup.

Licensing and cost optimization

Contoso will ensure that all development Azure resources are created through an Azure Dev/Test subscription to save 80 percent. The admins will enable Azure Cost Management + Billing to help monitor and manage the Azure resources.

Conclusion

In this article, Contoso rehosted the development VMs used for its SmartHotel360 and osTicket applications in Azure. The admins migrated the application VMs to Azure VMs by using the Azure Migrate: Server Migration tool.

Migrate a dev/test environment to Azure DevTest Labs

7/15/2021 • 11 minutes to read • Edit Online

This article demonstrates how the fictional company Contoso migrates its dev/test environment to Azure DevTest Labs.

Migration options

Contoso has several options available when moving its dev/test environment to Azure.

MIGRATION OPTIONS	OUTCOME
Azure Migrate	Assess and migrate on-premises VMs. Run dev/test servers by using Azure infrastructure as a service (IaaS). Manage VMs with Azure Resource Manager.
DevTest Labs	Quickly provision development and test environments. Minimize waste with quotas and policies. Set automated shutdowns to minimize costs. Build Windows and Linux environments.

> **NOTE**
> This article focuses on using DevTest Labs to move an on-premises dev/test environment to Azure. Read how Contoso moved dev/test to Azure IaaS via Azure Migrate.

Business drivers

The development leadership team has outlined what it wants to achieve with this migration:

- Empower developers with access to DevOps tools and self-service environments.
- Give access to DevOps tools for continuous integration/continuous delivery (CI/CD) pipelines and cloud-native tools for dev/test, such as AI, machine learning, and serverless.
- Ensure governance and compliance in dev/test environments.
- Save costs by moving all dev/test environments out of the datacenter and no longer purchase hardware to develop software.

> **NOTE**
>
> Contoso will use the Pay-As-You-Go Dev/Test subscription offer for its environments. Each active Visual Studio subscriber on the team can use the Microsoft software included with the subscription on Azure Virtual Machines for dev/test at no extra charge. Contoso will just pay the Linux rate for VMs that it runs. That includes VMs with SQL Server, SharePoint Server, or other software that's normally billed at a higher rate.

> **NOTE**
>
> Azure customers with an Enterprise Agreement can also benefit from the Azure Dev/Test subscription offer. To learn more, review the video for enabling and creating Enterprise Dev/Test subscriptions through the EA portal.

Migration goals

The Contoso development team has pinned down goals for this migration. These goals are used to determine the best migration method:

- Quickly provision development and test environments. It should take minutes, not months, to build the infrastructure that a developer needs to write or test software.
- After migration, Contoso's dev/test environment in Azure should have enhanced capabilities over the current system on-premises.
- The operations model will move from IT-provisioned VMs to DevOps with self-service provisioning.
- Contoso wants to quickly move out of its on-premises dev/test environments.
- All developers will connect to dev/test environments remotely and securely.

Solution design

After pinning down goals and requirements, Contoso designs and reviews a deployment solution. The solution includes the Azure services that it will use for dev/test.

Current architecture

- The dev/test VMs for Contoso's applications are running on VMware in the on-premises datacenter.
- These VMs are used for development and testing before code is promoted to the production VMs.
- Developers maintain their own workstations, but they need new solutions for connecting remotely from home offices.

Proposed architecture

- Contoso will use an Azure Dev/Test subscription to reduce costs for Azure resources. This subscription offers significant savings, including VMs that don't incur licensing fees for Microsoft software.
- Contoso will use DevTest Labs for managing the environments. New VMs will be created in DevTest Labs to support the move to new tools for development and testing in the cloud.
- The on-premises dev/test VMs in the Contoso datacenter will be decommissioned after the migration is done.
- Developers and testers will have access to Azure Virtual Desktop for their workstations.

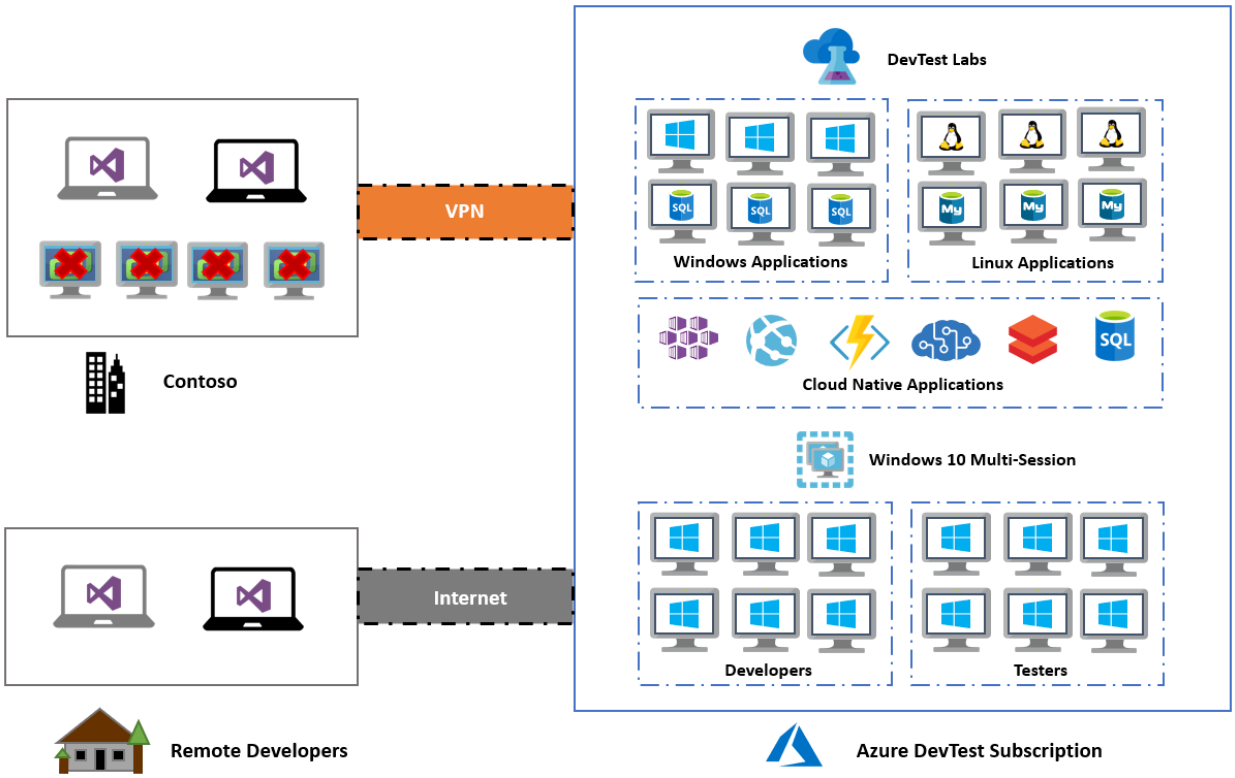


Figure 1: Scenario architecture.

Database considerations

To support ongoing development, Contoso has decided to continue using databases running on VMs. But the current VMs will be replaced with new ones running in DevTest Labs. In the future, Contoso will pursue the use of platform as a service (PaaS) services such as Azure SQL Database and Azure Database for MySQL.

Current VMware database VMs will be decommissioned and replaced with Azure VMs in DevTest Labs. The existing databases will be migrated with simple backups and restores. Using the Azure Dev/Test subscription offer won't incur licensing fees for the Windows Server and SQL Server instances, minimizing compute costs.

Solution review

Contoso evaluates the proposed design by putting together a list of pros and cons.

CONSIDERATION	DETAILS
Pros	All of the current development VMs (application and database) will be replaced by new VMs running in DevTest Labs. This means they can take advantage of the features of a purpose-built cloud development environment. Contoso can take advantage of its investment in the Azure Dev/Test subscription to save on licensing fees. Contoso will retain full control of the application VMs in Azure. Developers will be provided with rights to the subscription, which empowers them to create new resources without waiting for IT to respond to their requests.

CONSIDERATION	DETAILS
Cons	The migration will only move development to the cloud. Developers won't be using PaaS services in their development because they're still using VMs. This means that Contoso will need to start supporting the operations of its VMs, including security patches. IT maintained VMs in the past, and Contoso will need to find a solution for this new operational task. Contoso will have to build new application and database VMs, automating the process. This means it can take advantage of building VMs in the cloud and tools provided by DevTest Labs. So this is a positive outcome even with a con on the list.

Migration process

Contoso will migrate its development application and database VMs to new Azure VMs by using DevTest Labs.

- Contoso already has the Azure infrastructure in place, including the development virtual network.
- With everything prepared, Contoso will provision and configure DevTest Labs.
- Contoso will configure the development virtual network, assign a resource group, and set policies.
- Contoso will create Azure Virtual Desktop instances for developers to use at remote locations.
- Contoso will create VMs within DevTest Labs for development and migrate databases.

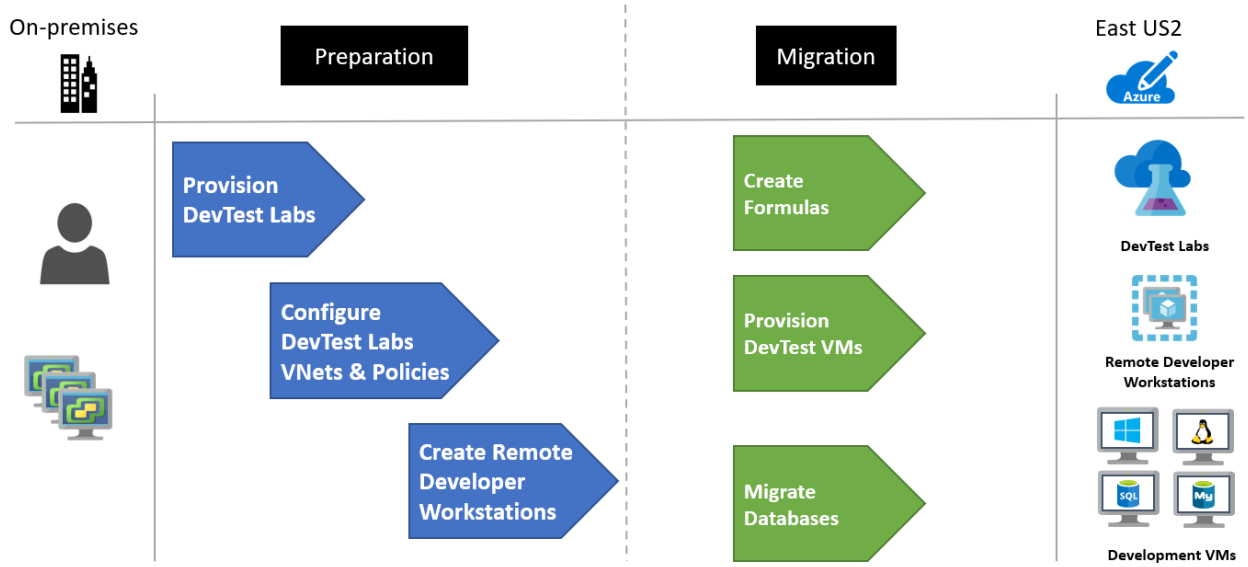


Figure 2: The migration process.

Prerequisites

Here's what Contoso needs to run this scenario.

REQUIREMENTS	DETAILS

REQUIREMENTS	DETAILS
Azure Dev/Test subscription	Contoso creates an Azure Dev/Test subscription to reduce costs up to 80 percent. If you don't have an Azure subscription, create a free account. If you create a free account, you're the admin of your subscription and can perform all actions. If you use an existing subscription and you're not the admin, work with the admin to assign you Owner or Contributor permissions. If you need more granular permissions, review manage Site Recovery access with Azure role-based access control.
Azure infrastructure	Learn how Contoso set up an Azure infrastructure.

Scenario steps

Here's how Contoso admins will run the migration:

- Step 1: Provision a new Azure Dev/Test subscription and create a DevTest Labs instance.
- Step 2: Configure the development virtual network, assign a resource group, and set policies.
- Step 3: Create Windows 10 Enterprise multi-session virtual desktops for developers to use from remote locations.
- Step 4: Create formulas and VMs within DevTest Labs for development and migrate databases.

Step 1: Provision a new Azure Dev/Test subscription and create a DevTest Labs instance

Contoso admins first need to provision a new subscription by using the Azure Dev/Test offer, and then create a DevTest Labs instance.

They set these up as follows:

The admins follow the link to the Azure Dev/Test subscription offer and provision a new subscription, which saves them up to 80 percent on their systems. This offer allows them to run Windows 10 images on Azure for dev/test. They will gain access to Azure Virtual Desktop to simplify the management experience of the remote developers.



Figure 3: An Azure Dev/Test subscription offer.

With their new subscription provisioned, Contoso admins use the Azure portal to create a new DevTest Labs instance. The new lab is created in the `ContosoDevRG` resource group.

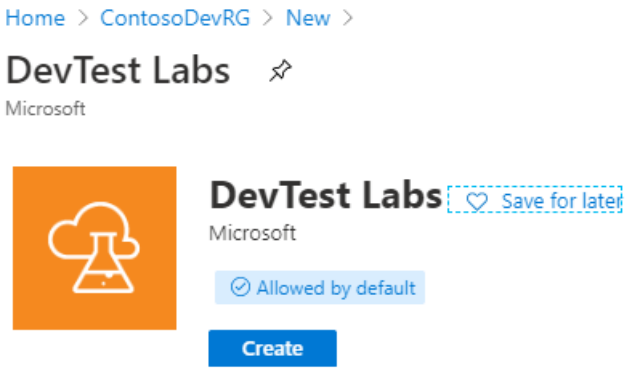


Figure 4: Creating a new DevTest Labs instance.

Step 2: Configure the development virtual network, assign a resource group, and set policies

With the DevTest Labs instance created, Contoso performs the following configurations:

1. Configure the virtual network:

   a. In the portal, Contoso opens the DevTest Labs instance and selects **Configuration and policies**.

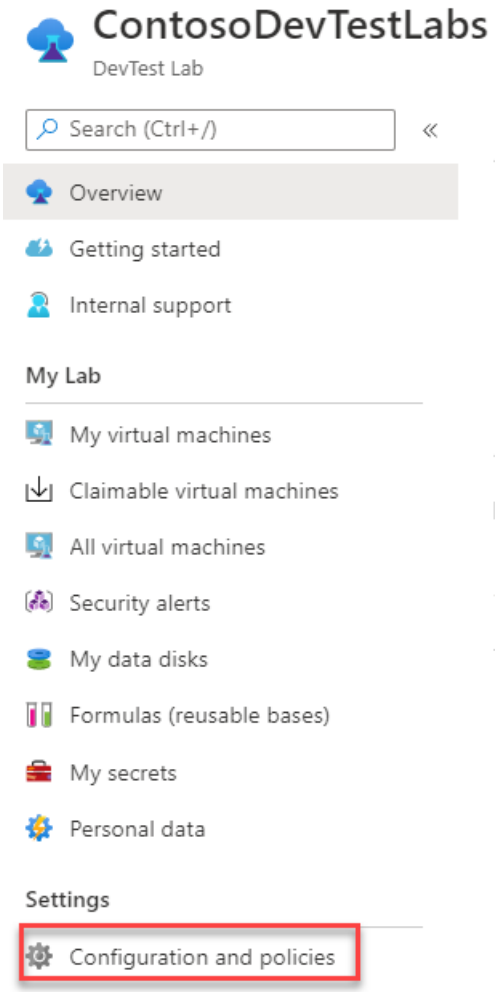


Figure 5: DevTest Labs instance: configuration and policies.

   b. Contoso selects **Virtual Networks** > **+ Add**, chooses `vnet-dev-eus2`, and then selects **Save**. This allows the development virtual network to be used for VM deployments. A virtual network was also created during the deployment of the DevTest Labs instance.

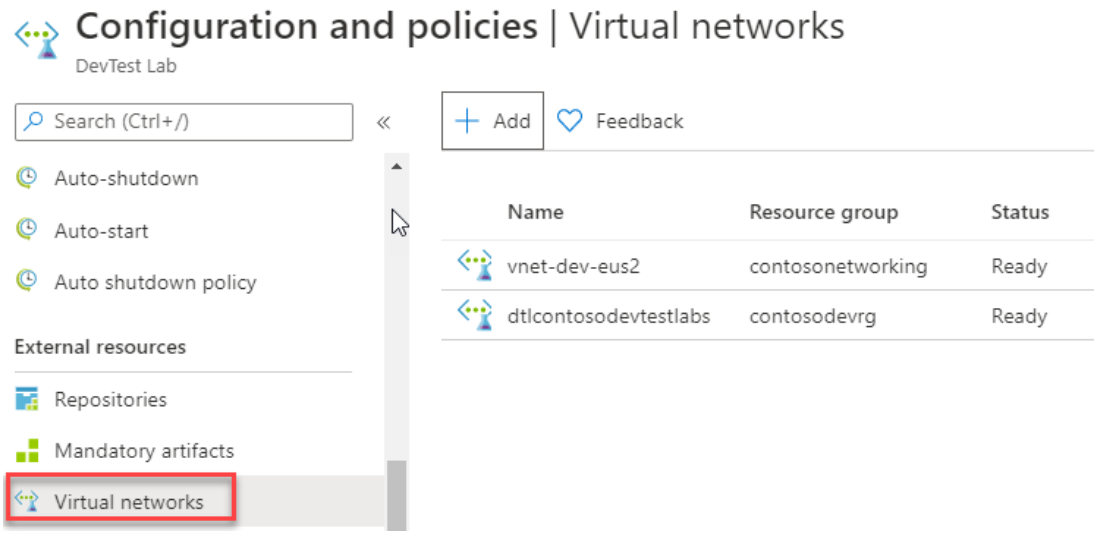


Figure 6: Virtual networks.

2. Assign a resource group:

   - To ensure that resources are deployed to the `ContosoDevRG` resource group, Contoso configures this in the lab settings. It also assigns its developers the **Contributor** role.

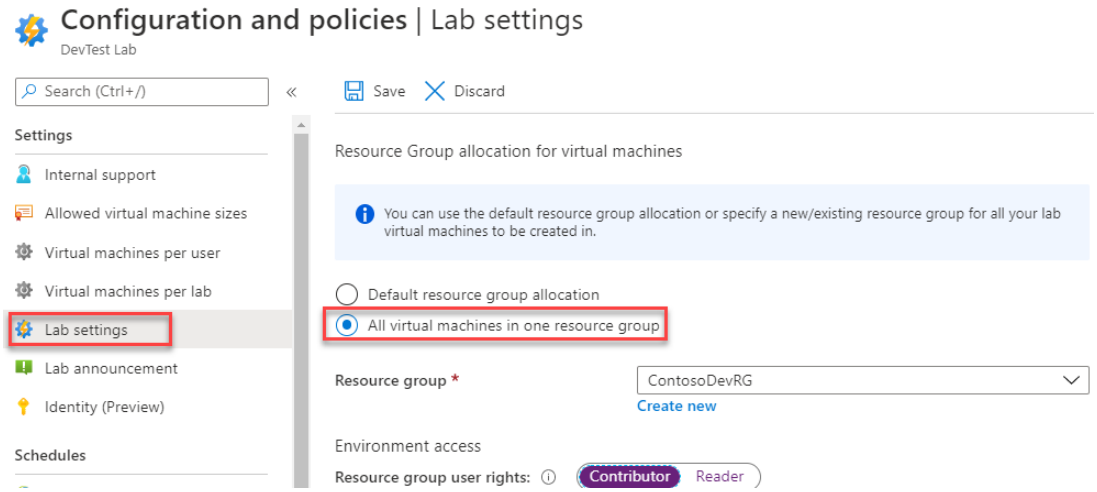


Figure 7: Assigning a resource group.

> **NOTE**
>
> The Contributor role is an administrator-level role with all rights except the ability to provide access to other users. Read more about Azure role-based access control.

3. Set lab policies:

   a. Contoso needs to ensure that its developers are using DevTest Labs within team policies. Contoso configures DevTest Labs with these policies.

   b. Contoso enables auto-shutdown with a local time of 7:00:00 PM and the correct time zone.

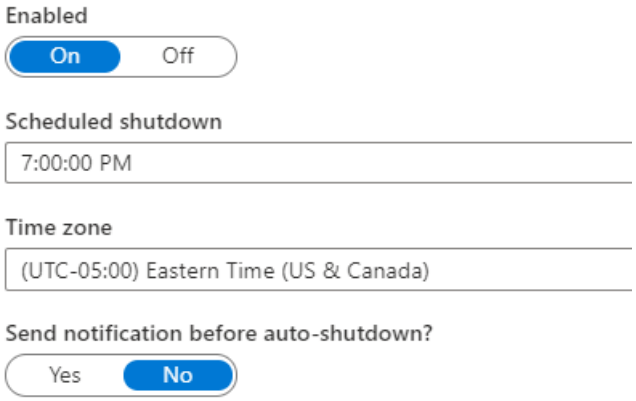


Figure 8: Auto-shutdown.

c. Contoso enables auto-start to have the VMs running when the developers come online to work. They're configured to the local time zone and for the days of the week when the developers work.

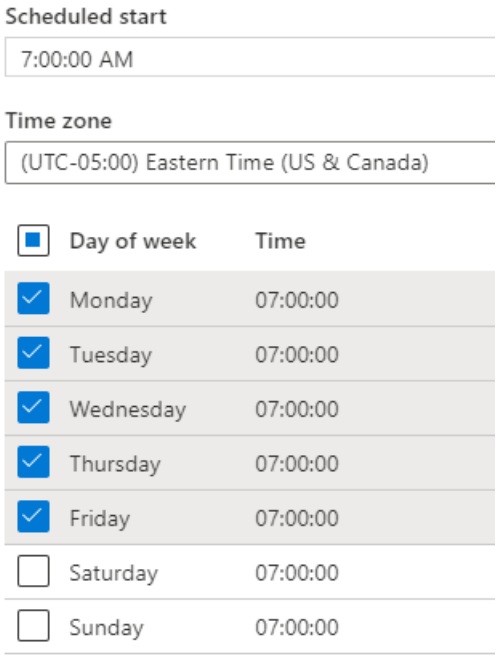


Figure 9: Auto-start.

d. Contoso configures the allowed VM sizes, ensuring that large and expensive VMs can't be started.

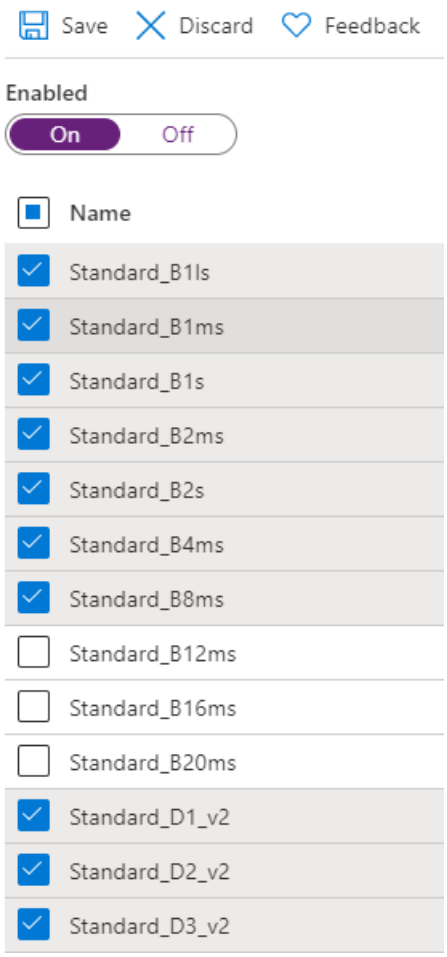


Figure 10: Allowed VM sizes.

e. Contoso configures the support message.

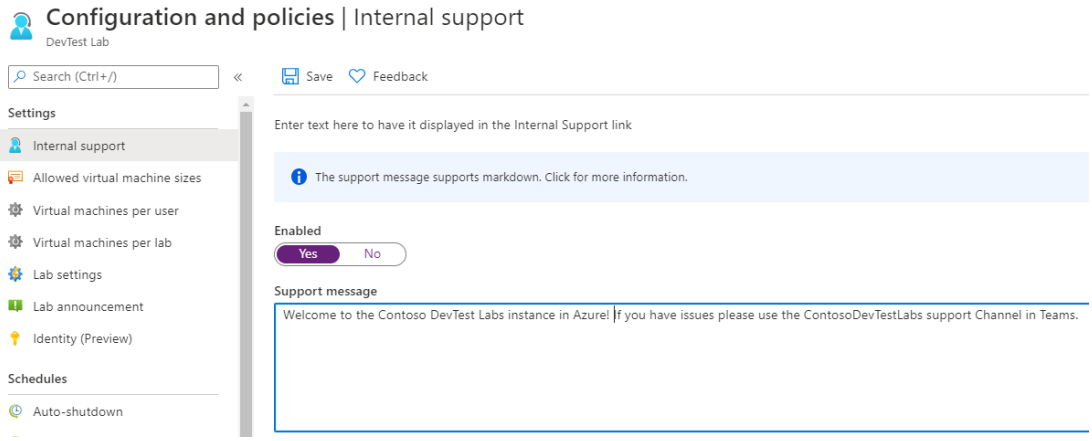


Figure 11: A support message.

Step 3: Create Windows 10 Enterprise multi-session virtual desktops for developers to use from remote locations

Contoso needs to create an Azure Virtual Desktop base for remote developers.

1. Contoso selects **All virtual machines** > **+ Add** and chooses a Windows 10 Enterprise multi-session base for a VM.

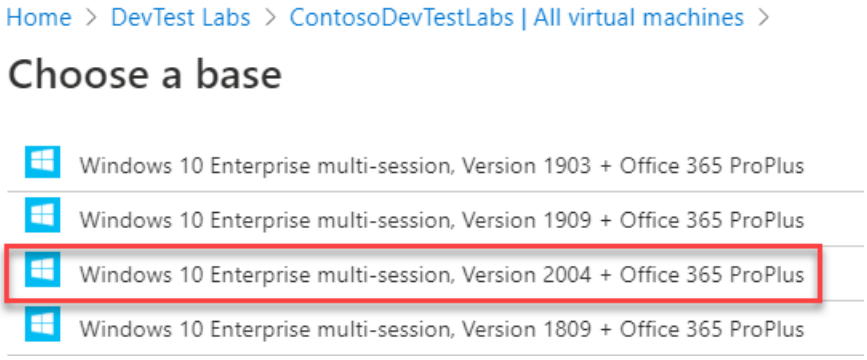


Figure 12: A Windows 10 Enterprise multi-session base.

2. Contoso configures the size of the VM along with the artifacts to be installed. In this case, the developers have access to common development tools such as Visual Studio Code, Git, and Chocolatey.

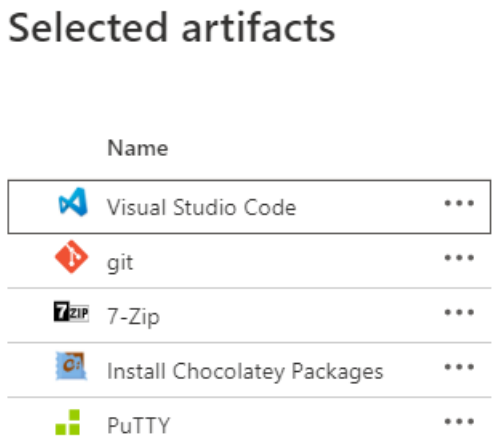


Figure 13: Artifacts.

3. Contoso reviews the VM configuration for accuracy.

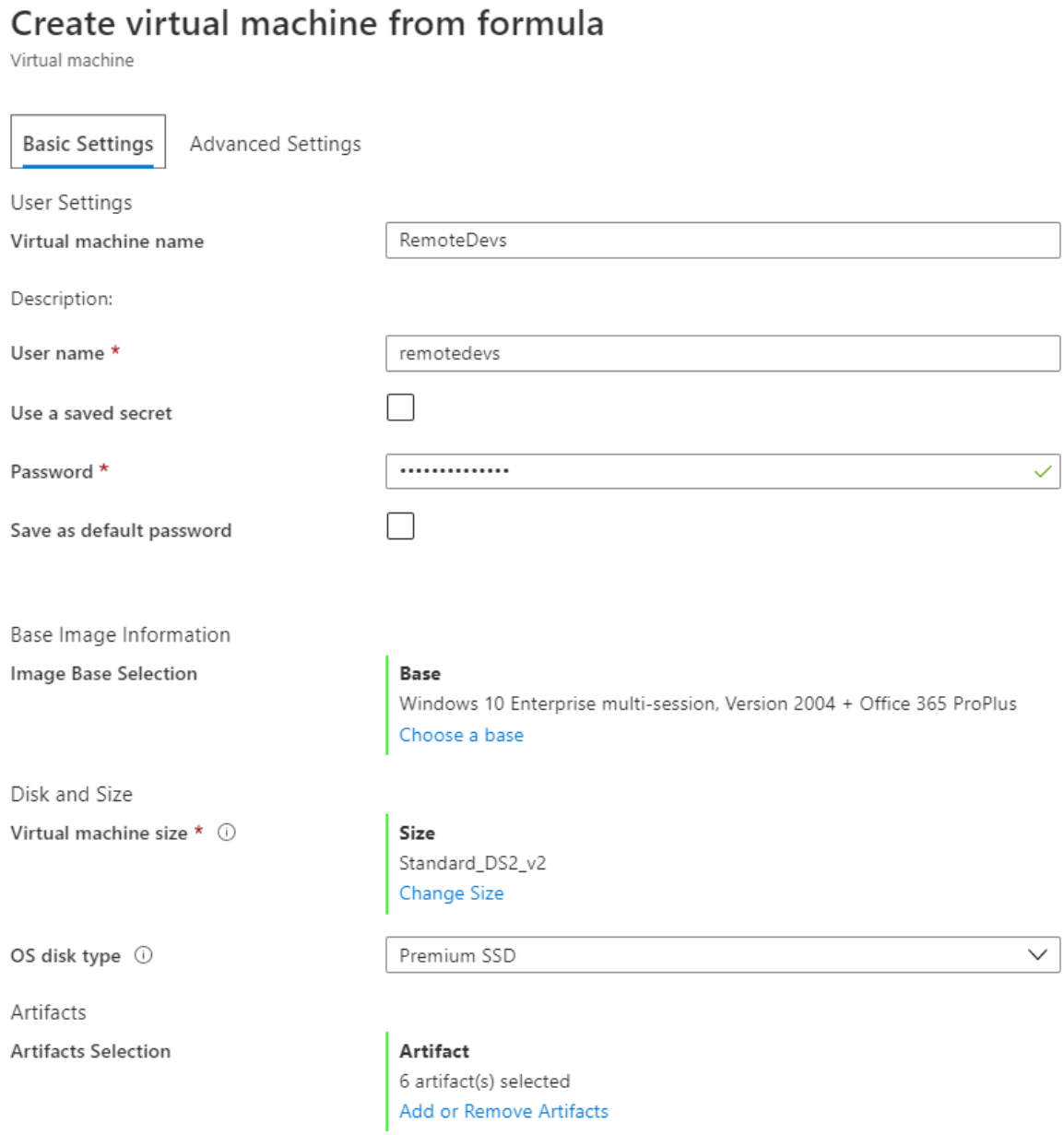


Figure 14: Create a virtual machine from a base.

4. After the VM is created, Contoso's remote developers can connect to and use this development workstation for their work. The selected artifacts are installed, saving developers time in configuring their workstation.

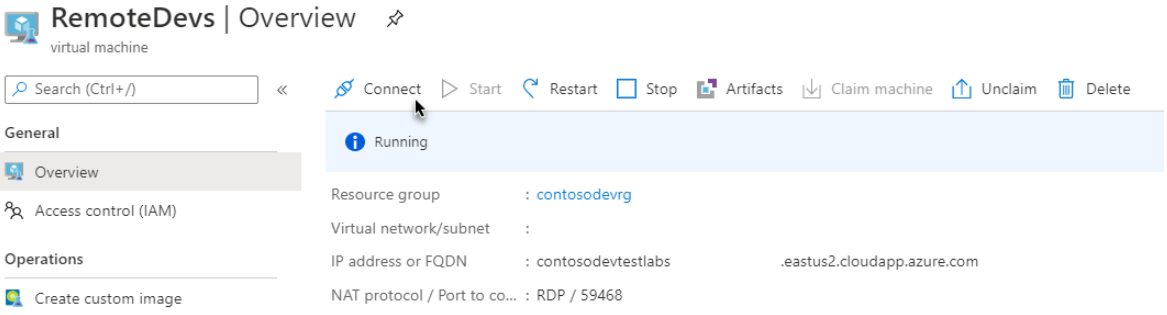


Figure 15: A remote developer VM.

Step 4: Create formulas and VMs within DevTest Labs for development and migrate databases

With DevTest Labs configured and the remote developers' workstation up and running, Contoso focuses on building its VMs for development. To get started, Contoso completes the following steps:

1. Contoso creates formulas (reusable bases) for application and database VMs, and it provisions application and database VMs by using the formulas.

   Contoso selects **Formulas** > **+ Add**, and then a **Windows Server 2012 R2 Datacenter** base.

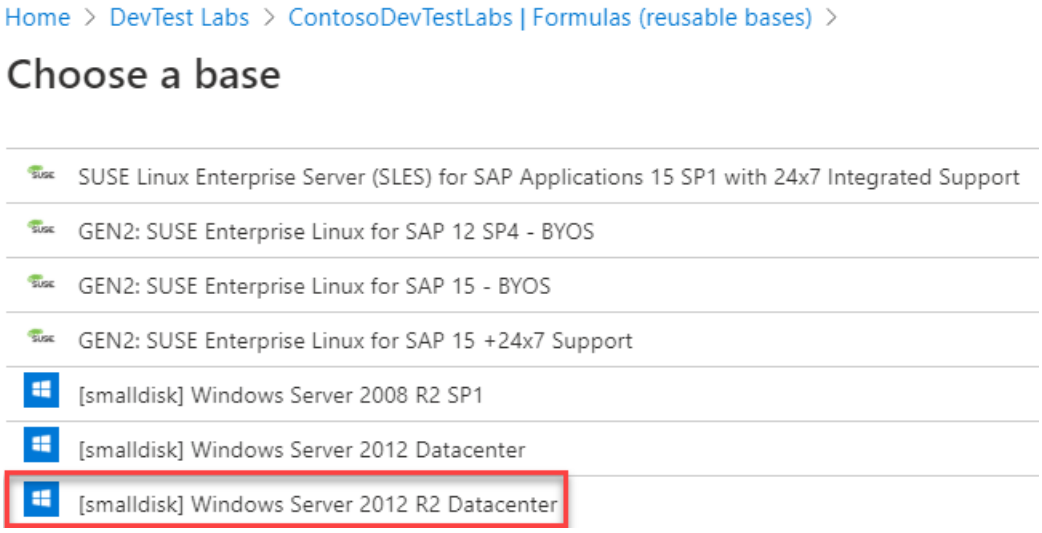


   Figure 16: A Windows 2012 R2 base.

2. Contoso configures the size of the VM along with the artifacts to be installed. In this case, the developers have access to common development tools such as Visual Studio Code, Git, and Chocolatey.

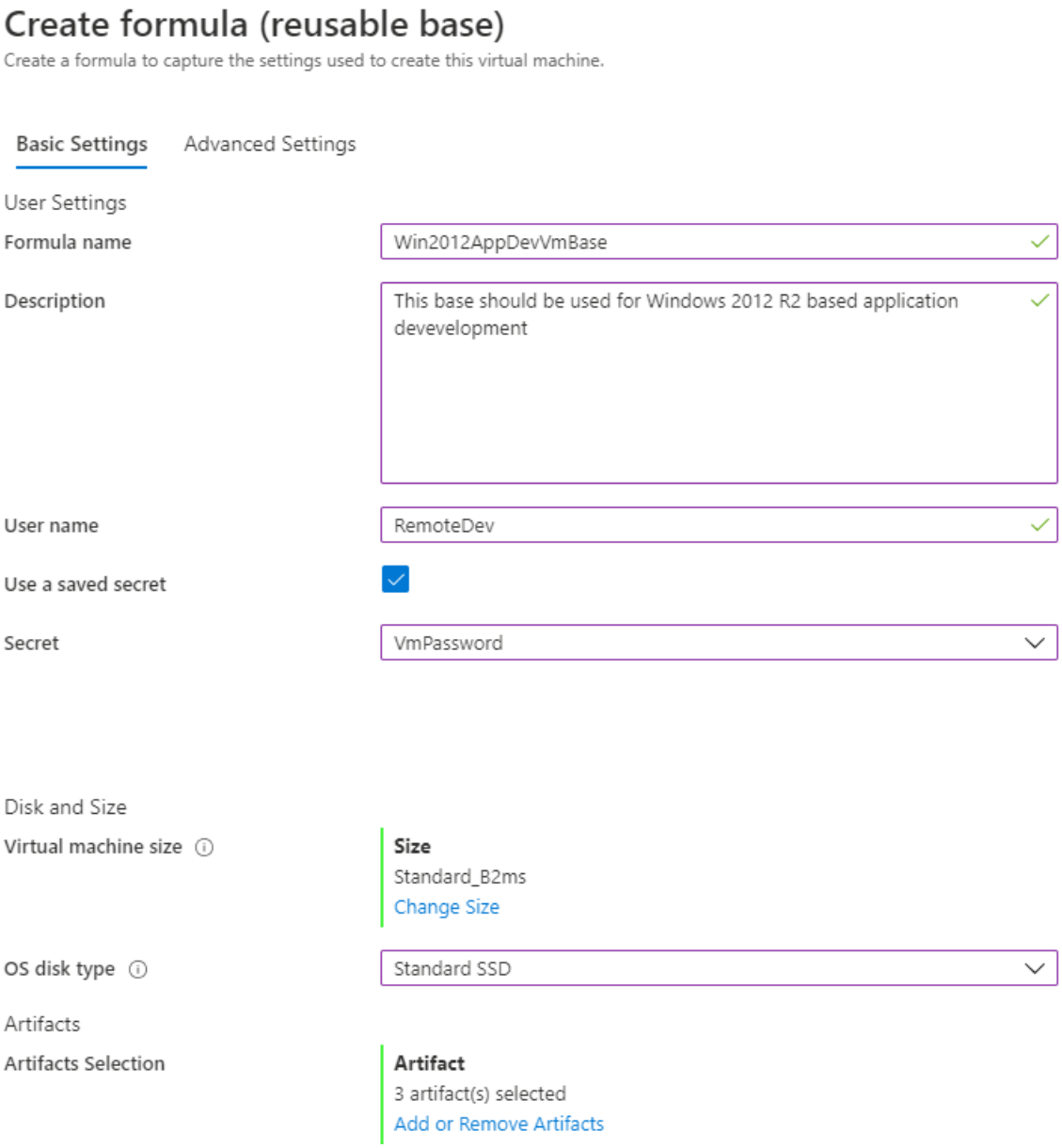


Figure 17: A Windows 2012 R2 base configuration.

3. To create the database VM formula, Contoso follows the same basic steps. This time, it selects a SQL Server 2012 image for the base.

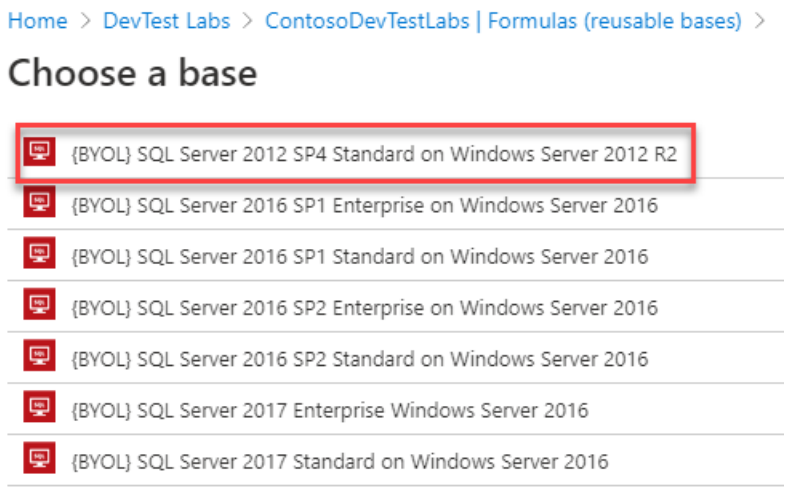


Figure 18: A SQL Server 2012 image.

4. Contoso configures the formula with the size and artifacts. The artifacts include SQL Server Management Studio, which is required for this database development VM formula.

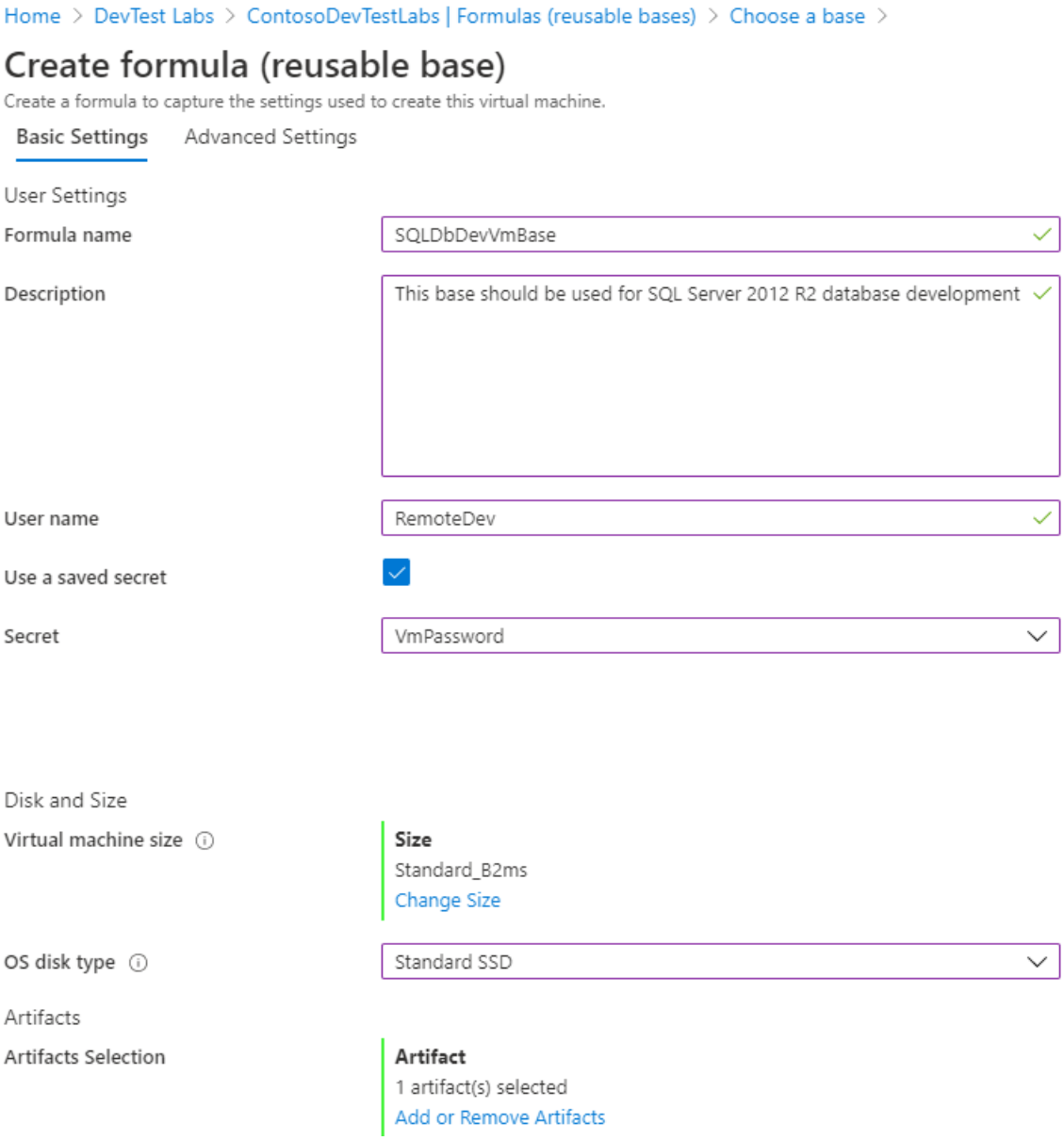


Figure 19: An SQL 2020 R2 base configuration.

Learn more about using formulas with Azure DevTest Labs.

5. Contoso has now created the Windows base formulas for its developers to use for applications and databases.

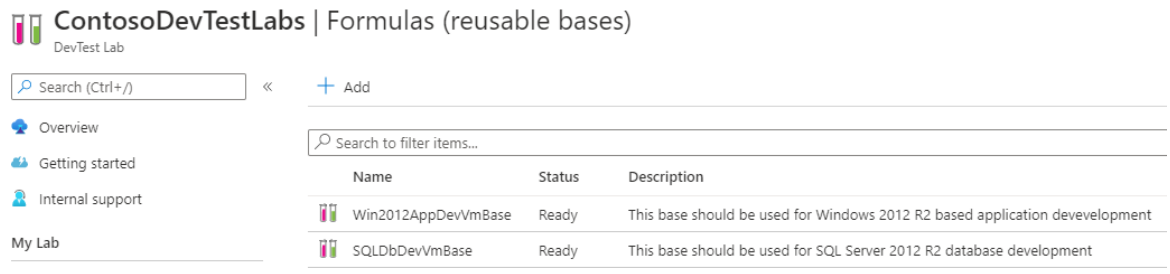


Figure 20: Windows base formulas.

The next steps provision application and database VMs through the formulas:

1. With the formulas created, Contoso next selects **All virtual machines** and then the `Windows2012AppDevVmBase` formula to match the configuration of its current application development VMs.

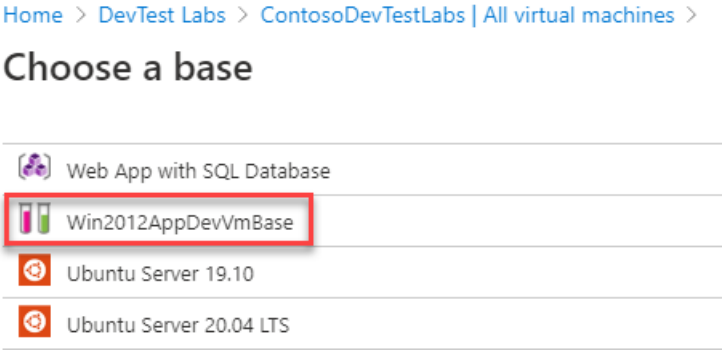


Figure 21: An application development VM.

2. Contoso configures the VM with the size and artifacts that are required for this application VM.

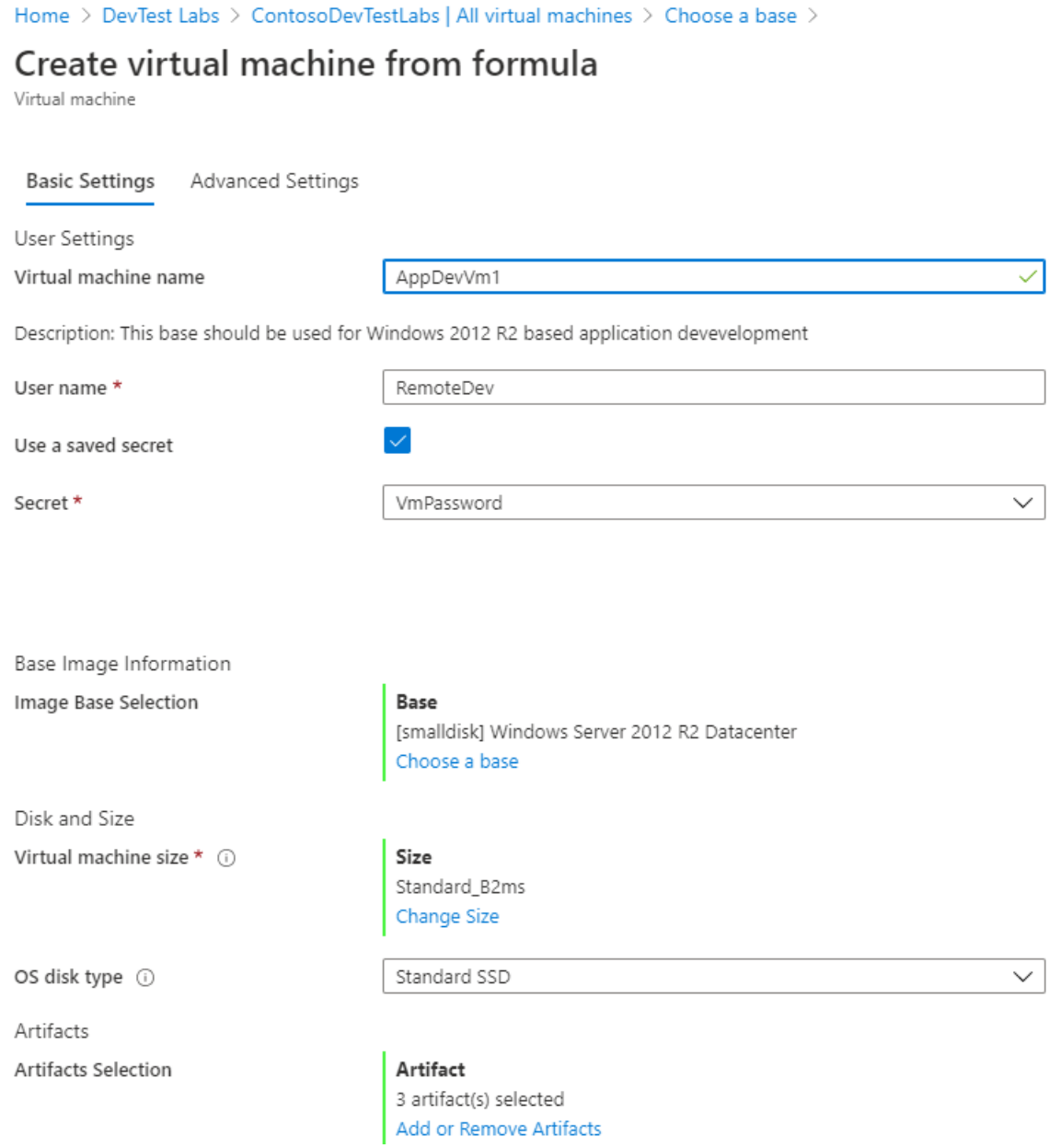


Figure 22: Size and artifact configurations for a VM.

3. Contoso provisions the database VM by using the `SQLDbDevVmBase` formula to match the configuration of its current database development VMs.

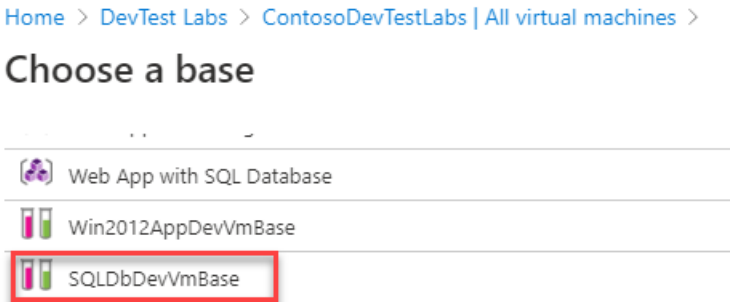


Figure 23: A database VM.

4. Contoso configures the VM with the size and artifacts that are required.

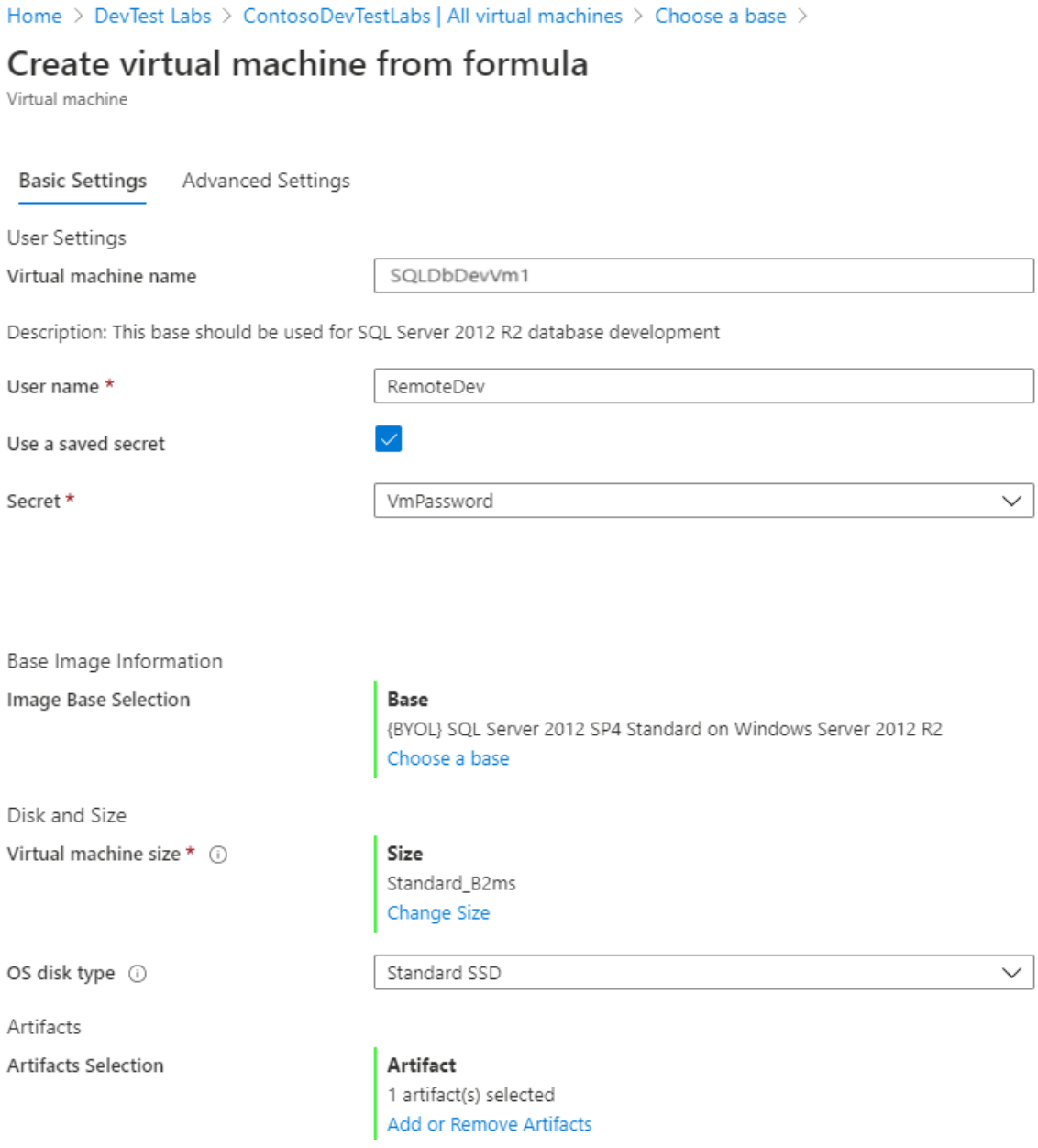


Figure 24: Database configurations for a VM.

5. With the first VMs created along with the remote developers' workstation, Contoso's developers are ready to start writing code in Azure.

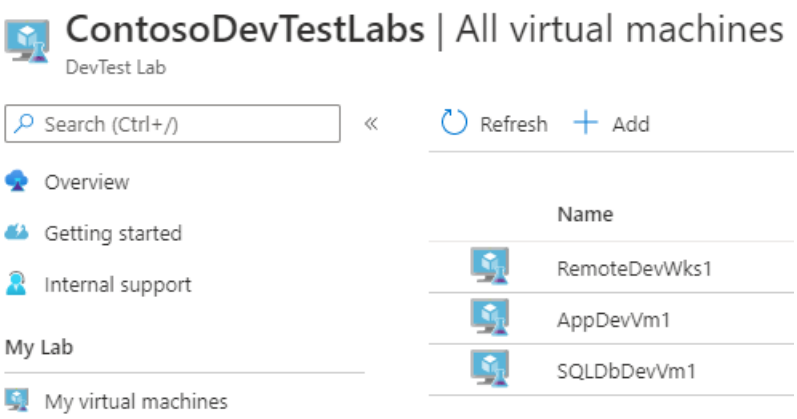


Figure 25: Contoso VMs.

6. Contoso can now restore its development databases either from backups or by using some type of code generation process to build the schema on the VMs. With SQL Server Management Studio already installed through the artifacts, these are simple tasks that don't require installing any tools.

Clean up after migration

Contoso will continue using these steps to migrate its VMs to Azure by using DevTest Labs. With each migration complete, all development VMs are now running in DevTest Labs.

Now, Contoso needs to complete these cleanup steps:

- Remove the VMs from the vCenter inventory.
- Remove all the VMs from local backup jobs.
- Update internal documentation to show the new location and IP addresses for the VMs.
- Review any resources that interact with the VMs, and update any relevant settings or documentation to reflect the new configuration.

Security

The Contoso security team reviews the Azure VMs to determine any security issues. To control access, the team reviews the network security groups (NSGs) for the VMs. NSGs are used to ensure that only traffic allowed to the application can reach it. The team also considers securing the data on the disk by using Azure Disk Encryption and Azure Key Vault. For more information, see Security best practices for IaaS workloads in Azure.

Licensing and cost optimization

- Contoso will ensure that all development Azure resources are created through an Azure Dev/Test subscription to take advantage of the 80 percent savings.
- Budgets will be reviewed for all DevTest Labs instances and policies for the VMs to ensure that costs are contained and overprovisioning doesn't happen mistakenly.
- Contoso will enable Azure Cost Management + Billing to help monitor and manage the Azure resources.

Conclusion

In this article, Contoso moved its development environments to DevTest Labs. It also implemented Azure Virtual Desktop as a platform for remote and contract developers.

Need more help?

Create a DevTest Labs instance in your subscription now, and learn how to use DevTest Labs for developers.

Migrate an application to Azure App Service and SQL Database

5/18/2021 • 15 minutes to read • Edit Online

This article demonstrates how the fictional company Contoso refactors a two-tier Windows .NET application that's running on VMware VMs as part of a migration to Azure. The Contoso team migrates the application front-end virtual machine (VM) to an Azure App Service web app and the application database to Azure SQL Database.

The SmartHotel360 application that we use in this example is provided as open-source software. If you want to use it for your own testing purposes, you can download it from GitHub.

Business drivers

The Contoso IT leadership team has worked closely with business partners to understand what they want to achieve with this migration:

- **Address business growth.** Contoso is growing, and there is pressure on their on-premises systems and infrastructure.
- **Increase efficiency.** Contoso needs to remove unnecessary procedures and streamline processes for developers and users. The business needs IT to be fast and not waste time or money, thus delivering faster on customer requirements.
- **Increase agility.** To enable their success in a global economy, Contoso IT needs to be more responsive to the needs of the business. It must be able to react more quickly to changes in the marketplace. IT must not get in the way or become a business blocker.
- **Scale.** As the business grows successfully, Contoso IT must provide systems that are able to grow at the same pace.
- **Reduce costs.** Contoso wants to minimize licensing costs.

Migration goals

To help determine the best migration method, the Contoso cloud team pinned down the following goals:

REQUIREMENTS	DETAILS

REQUIREMENTS	DETAILS
Application	The application in Azure will remain as critical as it is today on-premises. It should have the same performance capabilities as it currently does in VMware. The team doesn't want to invest in the application. For now, admins will simply move the application safely to the cloud. The team wants to stop supporting Windows Server 2008 R2, which the application currently runs on. The team also wants to move away from SQL Server 2008 R2 to a modern platform as a service (PaaS) database, which will minimize the need for management. Contoso wants to take advantage of its investment in SQL Server licensing and Software Assurance where possible. In addition, Contoso wants to mitigate the single point of failure on the web tier.
Limitations	The application consists of an ASP.NET application and a Windows Communication Foundation (WCF) service running on the same VM. They want to spread these components across two web apps using the Azure App Service.
Azure	Contoso wants to move the application to Azure, but they don't want to run it on VMs. Contoso wants to use Azure PaaS services for both the web and data tiers.
DevOps	Contoso wants to move to a DevOps model that uses Azure DevOps for their builds and release pipelines.

Solution design

After pinning down their goals and requirements, Contoso designs and reviews a deployment solution. They also identify the migration process, including the Azure services that they'll use for the migration.

Current application

- The SmartHotel360 on-premises application is tiered across two VMs, `WEBVM` and `SQLVM`.
- The VMs are located on VMware ESXi host contosohost1.contoso.com version 6.5.
- The VMware environment is managed by vCenter Server 6.5 (`vcenter.contoso.com`), which runs on a VM.
- Contoso has an on-premises datacenter (contoso-datacenter), with an on-premises domain controller (contosodc1).
- The on-premises VMs in the Contoso datacenter will be decommissioned after the migration is done.

Proposed solution

- For the database tier of the application, Contoso compared Azure SQL Database to SQL Server by referring to the features comparison: Azure SQL Database and Azure SQL Managed Instance. Contoso decided to use Azure SQL Database for a few reasons:
  - Azure SQL Database is a managed relational database service. It delivers predictable performance at multiple service levels, with near-zero administration. Advantages include dynamic scalability with no downtime, built-in intelligent optimization, and global scalability and availability.

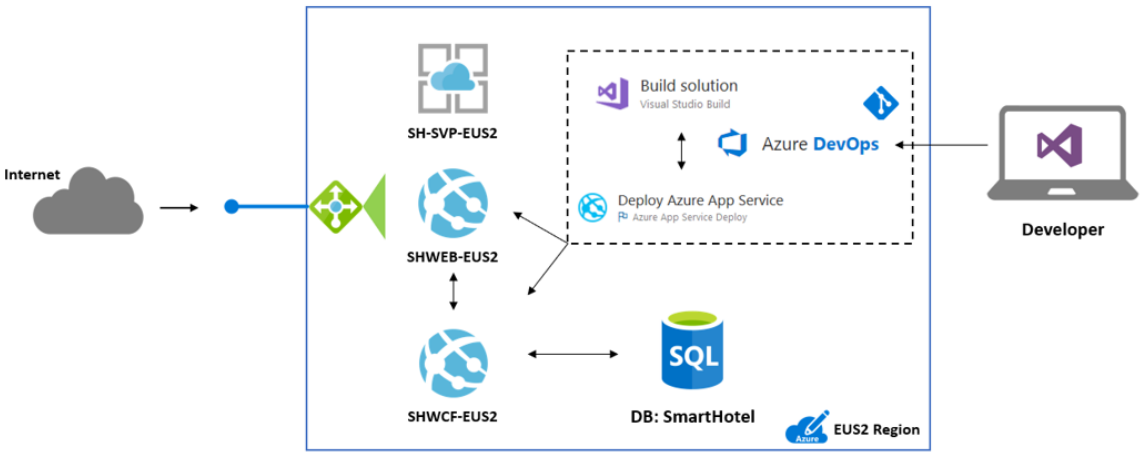
    - Contoso can use the lightweight Data Migration Assistant to assess the on-premises database migration to Azure SQL Database.
    - Contoso can use Azure Database Migration Service to migrate the on-premises database to Azure SQL Database.
    - With Software Assurance, Contoso can exchange existing licenses for discounted rates on a database in SQL Database by using the Azure Hybrid Benefit for SQL Server. This approach could provide a cost saving of up to 30 percent.
    - SQL Database provides security features such as Always Encrypted, dynamic data masking, row-level security, and SQL threat detection.
- For the application web tier, Contoso has decided to use Azure App Service. This PaaS service enables them to deploy the application with just a few configuration changes. Contoso will use Visual Studio to make the change, and they'll deploy two web apps, one for the website and one for the WCF service.
- To meet requirements for a DevOps pipeline, Contoso will use Azure DevOps for source code management with Git repos. They'll use automated builds and release to build the code and deploy it to the Azure App Service.

Solution review

Contoso evaluates their proposed design by putting together a pros and cons list, as shown in the following table:

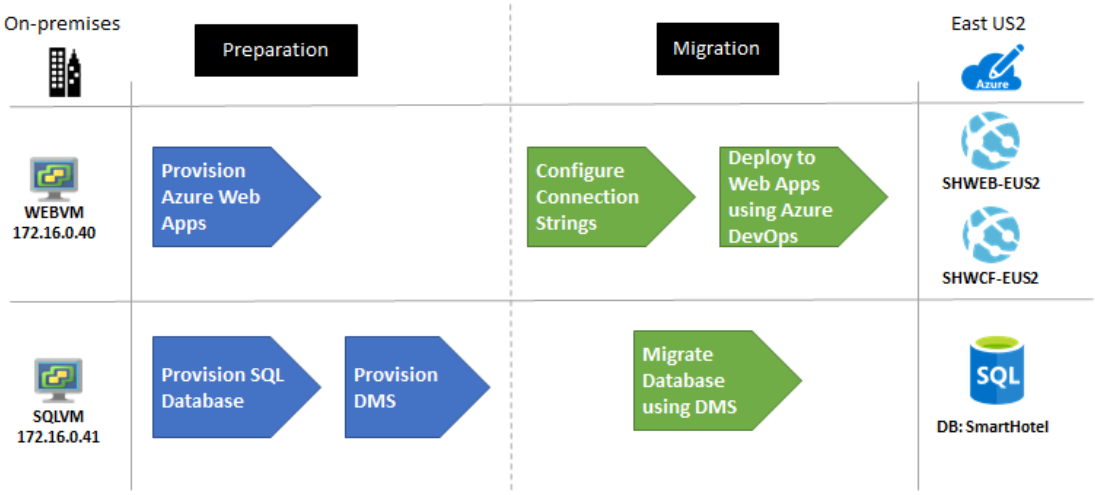
CONSIDERATION	DETAILS
Pros	The SmartHotel360 application code doesn't require changes for migration to Azure. Contoso can take advantage of their investment in Software Assurance by using the Azure Hybrid Benefit for both SQL Server and Windows Server. After the migration, Windows Server 2008 R2 won't need to be supported. For more information, see the Microsoft Lifecycle Policy. Contoso can configure the web tier of the application with multiple instances, so that the web tier is no longer a single point of failure. The database will no longer depend on the aging SQL Server 2008 R2. SQL Database supports the technical requirements. Contoso assessed the on-premises database by using Data Migration Assistant and found that it's compatible. Azure SQL Database has built-in fault tolerance that Contoso doesn't have to set up. This ensures that the data tier is no longer a single point of failover. If Contoso uses Azure Database Migration Service to migrate their database, it will have the infrastructure ready for migrating databases at scale.
Cons	Azure App Service supports only one application deployment for each web app. This means that two web apps must be provisioned, one for the website and one for the WCF service.

Proposed architecture



Migration process

1. Contoso provisions an Azure SQL managed instance and then migrates the SmartHotel360 database to it by using Azure Database Migration Service.
2. Contoso provisions and configures web apps, and deploys the SmartHotel360 application to them.



Azure services

SERVICE	DESCRIPTION	COST
Azure App Service Migration Assistant	A free and simple path to seamlessly migrate .NET web applications from on-premises to the cloud with minimal to no code changes.	It's a downloadable tool, free of charge.

SERVICE	DESCRIPTION	COST
Data Migration Assistant	Contoso will use Data Migration Assistant to assess and detect compatibility issues that might affect database functionality in Azure. Data Migration Assistant assesses feature parity between SQL sources and targets, and it recommends performance and reliability improvements.	It's a downloadable tool, free of charge.
Azure Database Migration Service	Azure Database Migration Service enables seamless migration from multiple database sources to Azure data platforms with minimal downtime.	Learn about supported regions and Database Migration Service pricing.
Azure SQL Database	An intelligent, fully managed relational cloud database service.	Cost is based on features, throughput, and size. Learn more.
Azure App Service	Helps create powerful cloud applications that use a fully managed platform.	Pricing is based on size, location, and usage duration. Learn more.
Azure DevOps	Provides a continuous integration and continuous deployment (CI/CD) pipeline for application development. The pipeline starts with a Git repository for managing application code, a build system for producing packages and other build artifacts, and a release management system to deploy changes in dev, test, and production environments.	

Prerequisites

To run this scenario, Contoso must meet the following prerequisites:

REQUIREMENTS	DETAILS
Azure subscription	Contoso created subscriptions earlier in this article series. If you don't have an Azure subscription, create a free account. If you create a free account, you're the administrator of your subscription and can perform all actions. If you use an existing subscription and you're not the administrator, you need to work with the admin to assign you Owner or Contributor permissions.
Azure infrastructure	Contoso set up their Azure infrastructure as described in Azure infrastructure for migration.

Scenario steps

Here's how Contoso will run the migration:

- **Step 1: Assess and migrate the web apps.**. Contoso uses the Azure App Service Migration Assistant tool to run pre-migration compatibility checks and migrate their web apps to Azure App Service.
- **Step 2: Provision a database in Azure SQL Database**. Contoso provisions an Azure SQL Database instance. After the application website is migrated to Azure, the WCF service web app will point to this instance.
- **Step 3: Assess the database**. Contoso assesses the database for migration by using Data Migration Assistant and then migrates it via Azure Database Migration Service.
- **Step 4: Set up Azure DevOps**. Contoso creates a new Azure DevOps project, and imports the Git repo.
- **Step 5: Configure connection strings**. Contoso configures connection strings so that the web tier web app, the WCF service web app, and the SQL instance can communicate.
- **Step 6: Set up build and release pipelines in Azure DevOps**. As a final step, Contoso sets up build and release pipelines in Azure DevOps to create the application, and then deploys them to two separate web apps.

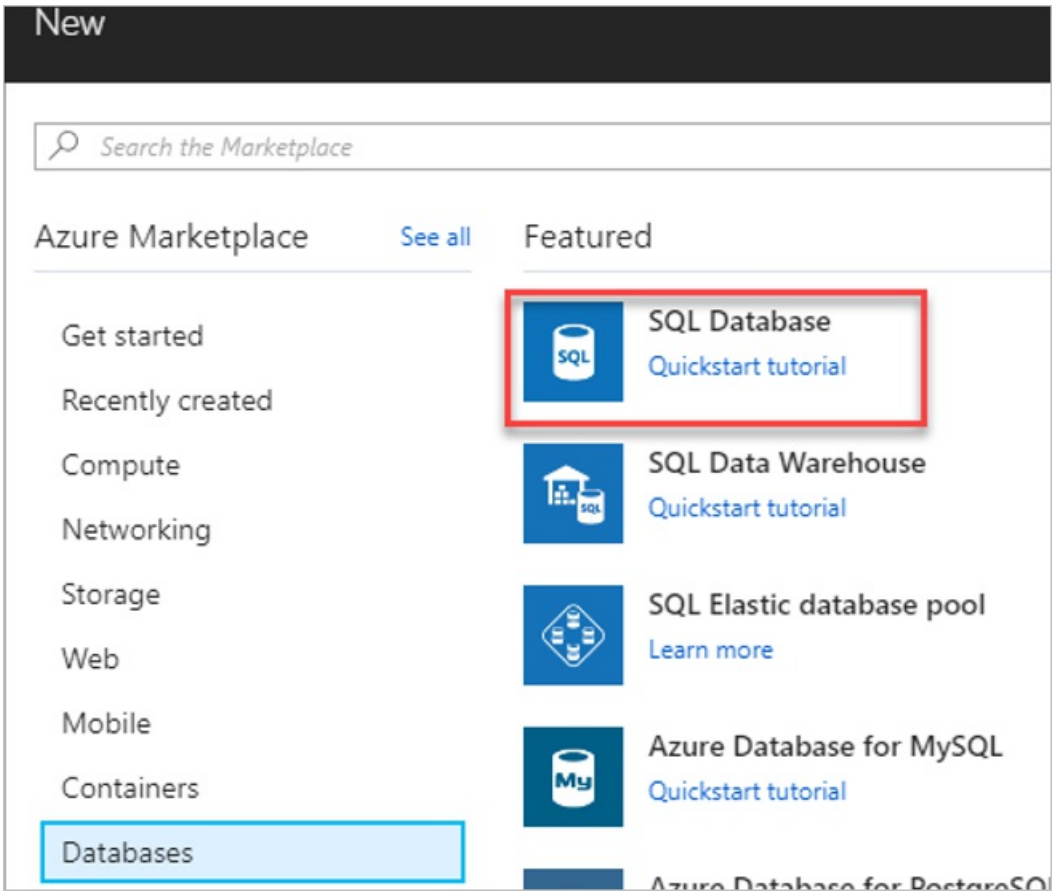
Step 1: Assess and migrate the web apps

Contoso admins assess and migrate their web app using the Azure App Service Migration Assistant tool. They use the Migrate ASP.NET Apps to Azure learning path as a guide during the process. The admins perform these actions:

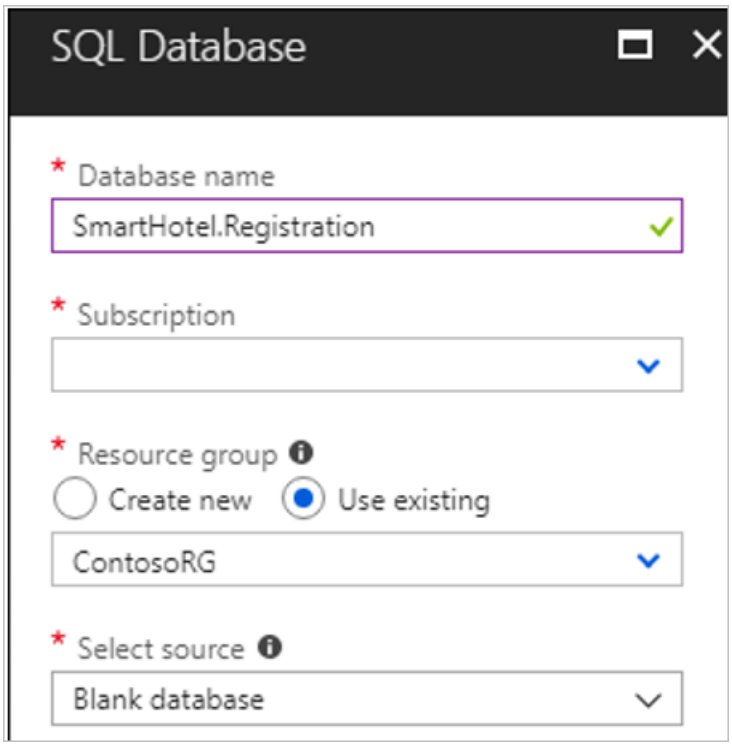
- They use the Azure App Service migration assessment tool to evaluate any dependencies between their web apps and to determine whether there are any incompatibilities between their on-premises web apps and what's supported on Azure App Service.
- They download the Azure App Service Migration Assistant and sign in to their Azure account.
- They choose a subscription, a resource group, and the website's domain name.

Step 2: Provision a database in Azure SQL Database

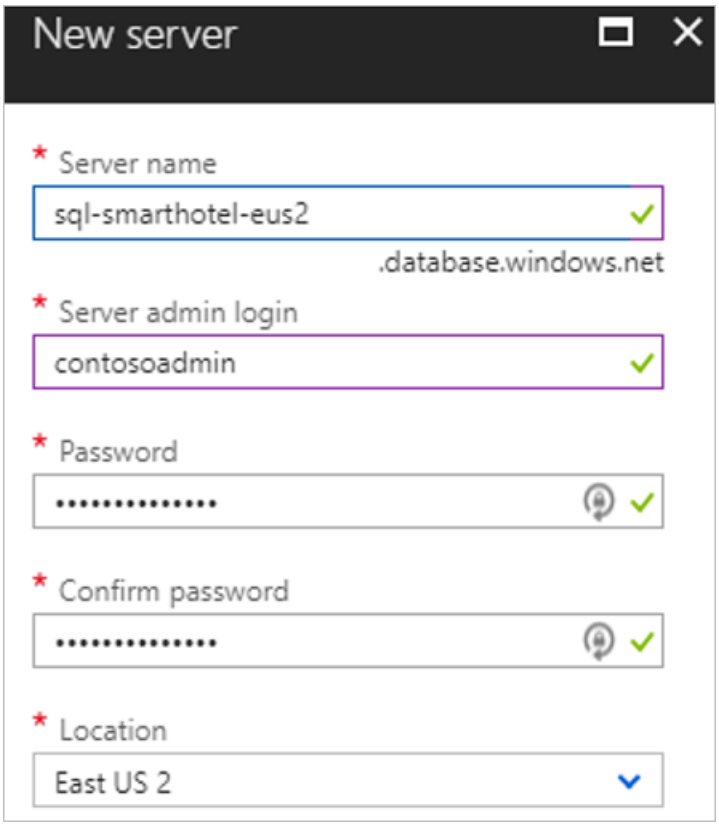
1. Contoso admins decide to create an Azure SQL Database instance.



2. They specify a database name to match the database, `SmartHotel.Registration`, that's running on the on-premises VM. They place the database in the `ContosoRG` resource group. This is the resource group they use for production resources in Azure.

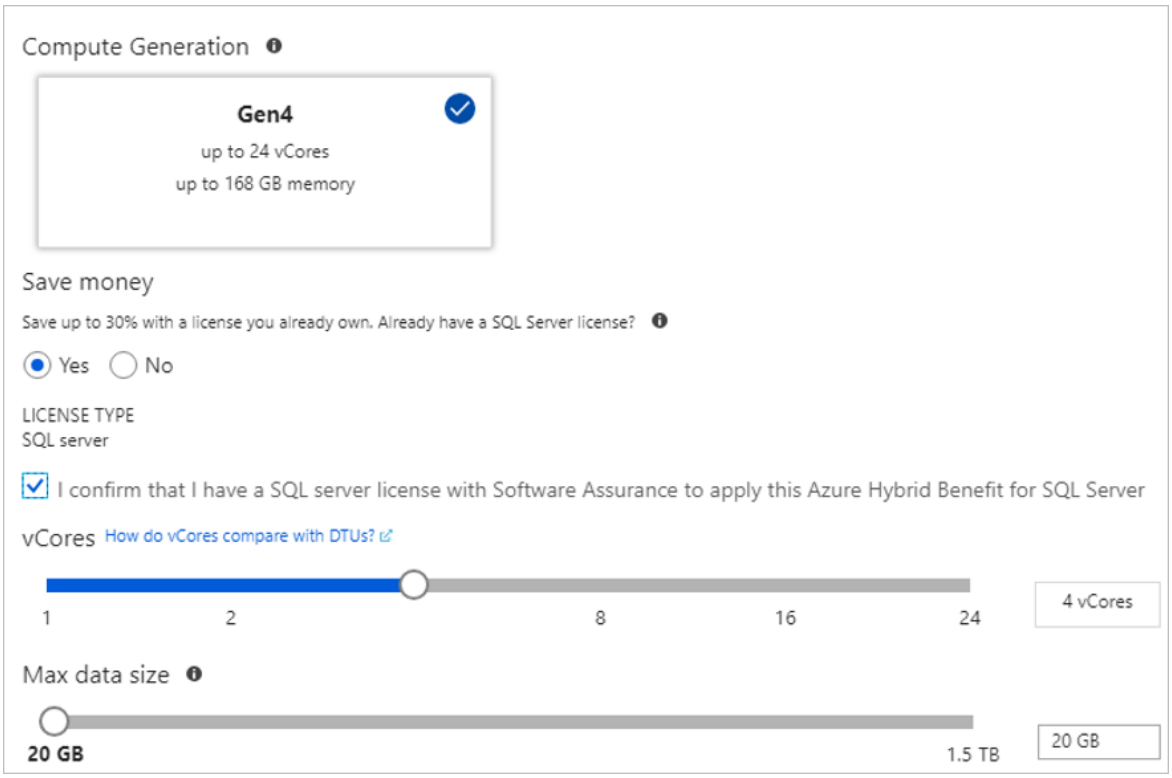


3. They set up a new SQL Server instance, `sql-smarthotel-eus2`, in the primary region.

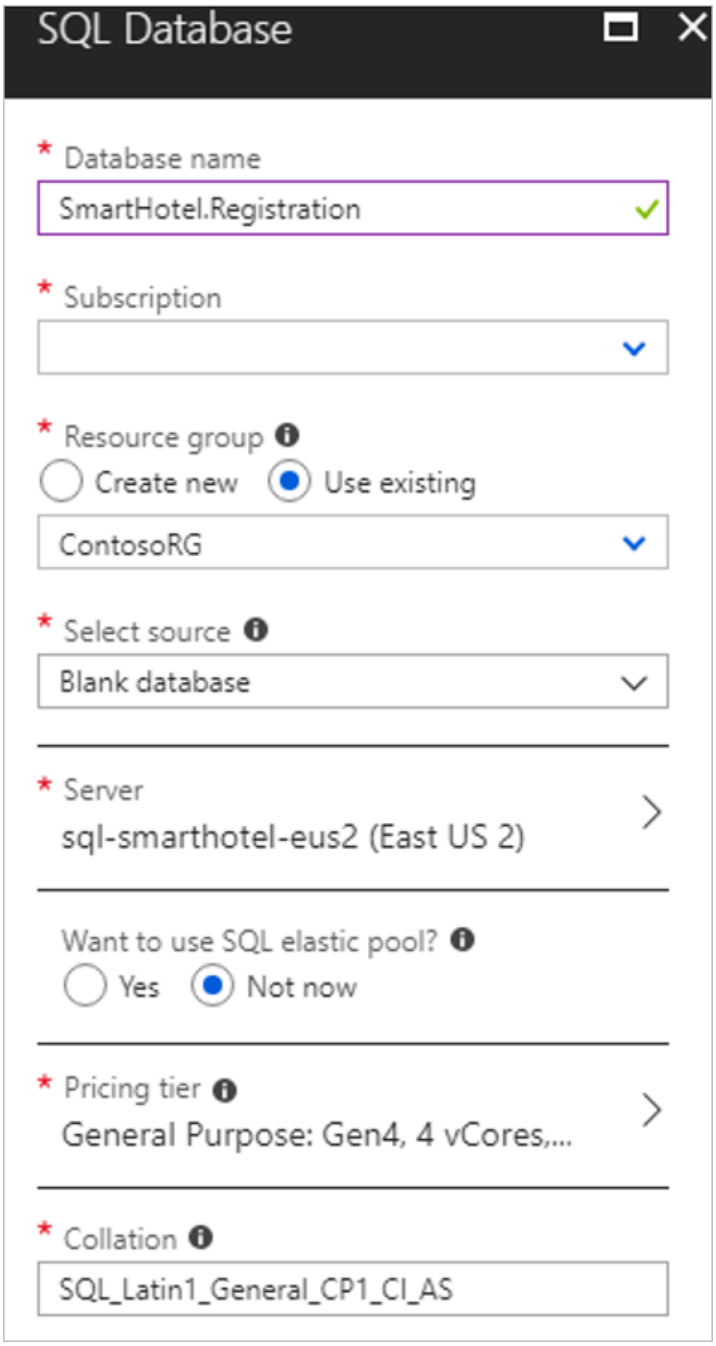


4. They set the pricing tier to match their server and database needs. And they select to save money with Azure Hybrid Benefit because they already have a SQL Server license.

5. For sizing, they use vCore-based purchasing and set the limits for their expected requirements.



6. They create the database instance.



7. They open the database and note the details they'll need when they use Data Migration Assistant for migration.



Need more help?

- Get help provisioning a SQL database.
- Learn about vCore resource limits.

Step 3: Assess the database

Contoso admins assess the database by using Data Migration Assistant and then migrate it by using Azure

Database Migration Service by referring to the step-by-step migration tutorial. They can perform online, offline, and hybrid (preview) migrations.

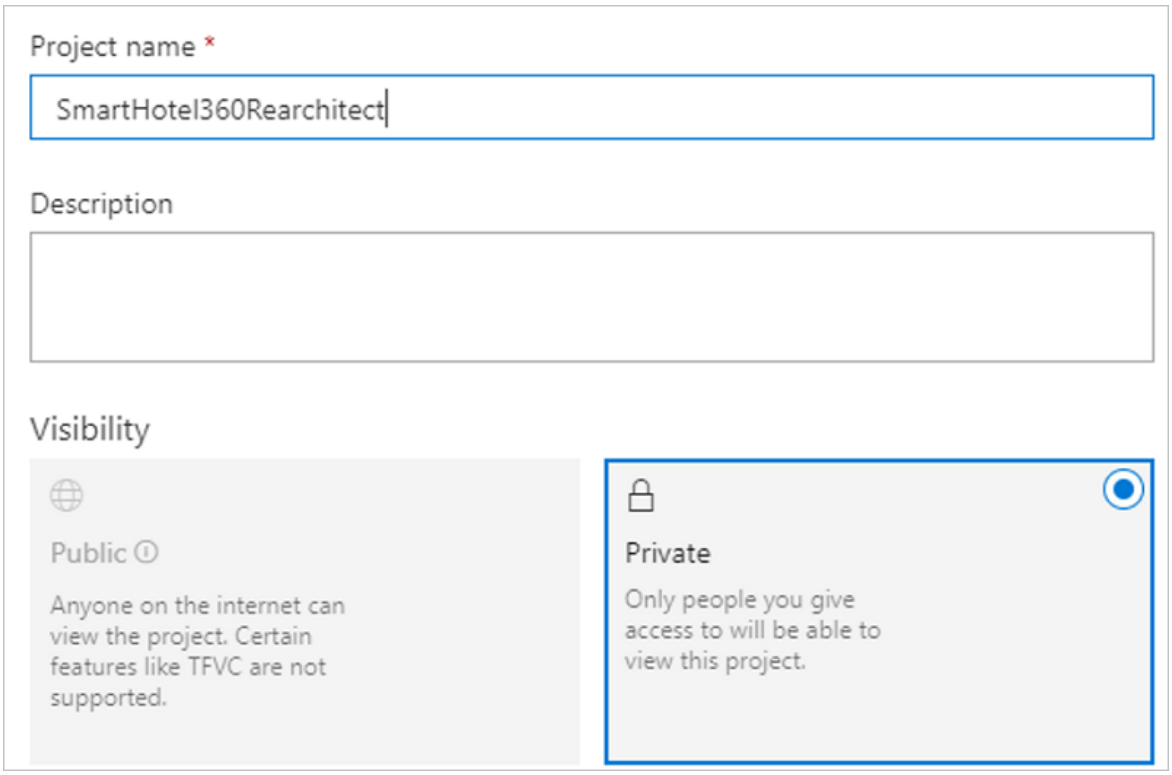
In brief, the admins do the following:

- They use Data Migration Assistant to discover and resolve any database migration issues.
- They create an Azure Database Migration Service instance with a Premium SKU that's connected to the virtual network.
- They ensure that the instance can access the remote SQL Server via the virtual network. This entails ensuring that all incoming ports are allowed from Azure to SQL Server at the virtual network level, the network VPN, and the machine that hosts SQL Server.
- They configure the instance:
  - Create a migration project.
  - Add a source (on-premises database).
  - Select a target.
  - Select the databases to migrate.
  - Configure advanced settings.
  - Start the replication.
  - Resolve any errors.
  - Perform the final cutover.

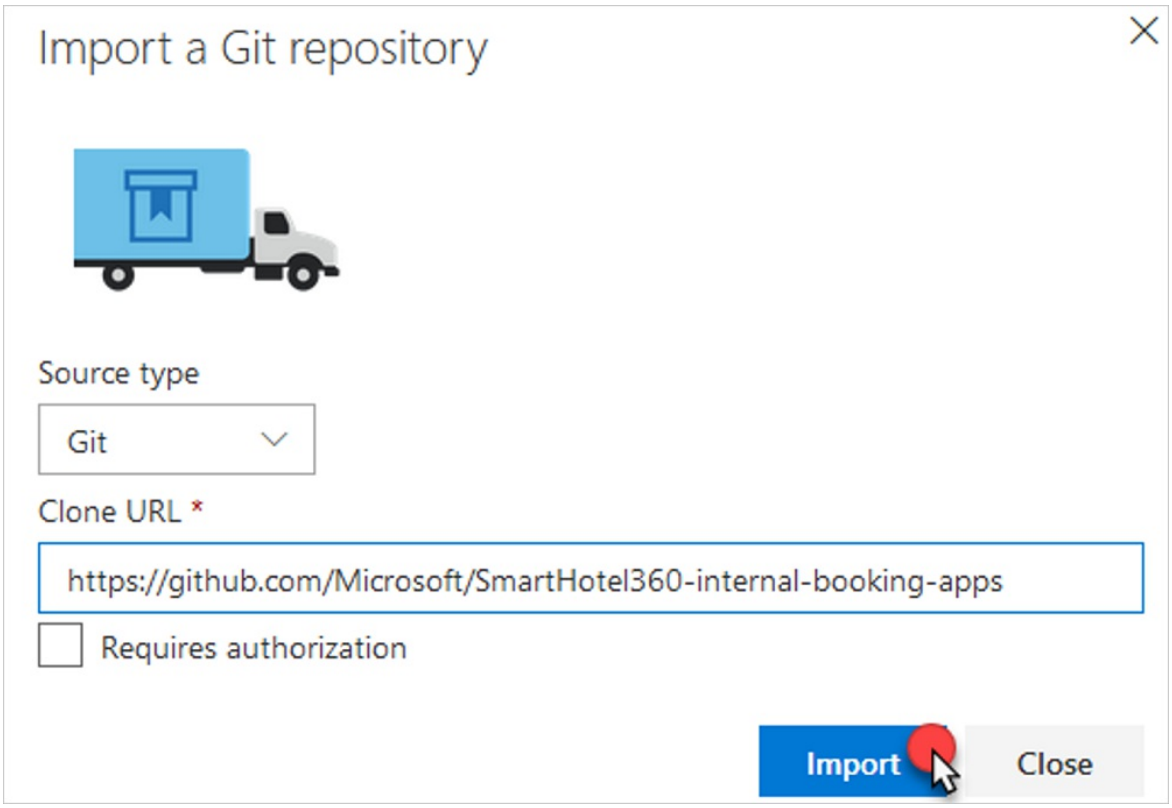
Step 4: Set up Azure DevOps

Contoso needs to build the DevOps infrastructure and pipelines for the application. To do this, Contoso admins create a new DevOps project, import the code, and then set up build and release pipelines.

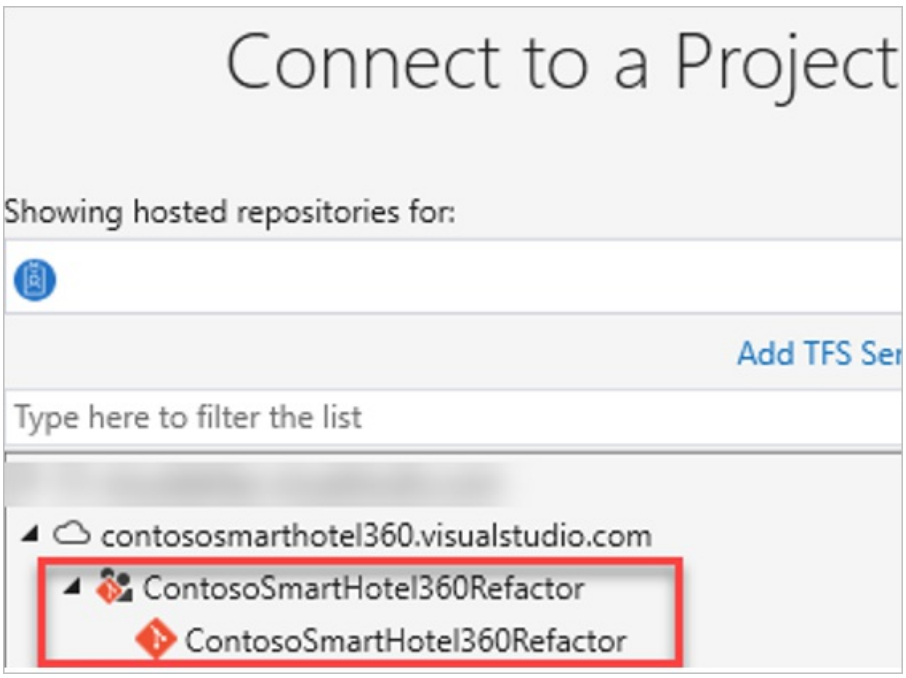
1. In the Contoso Azure DevOps account, they create a new project, `ContosoSmartHotelRefactor`, and then select **Git** for version control.



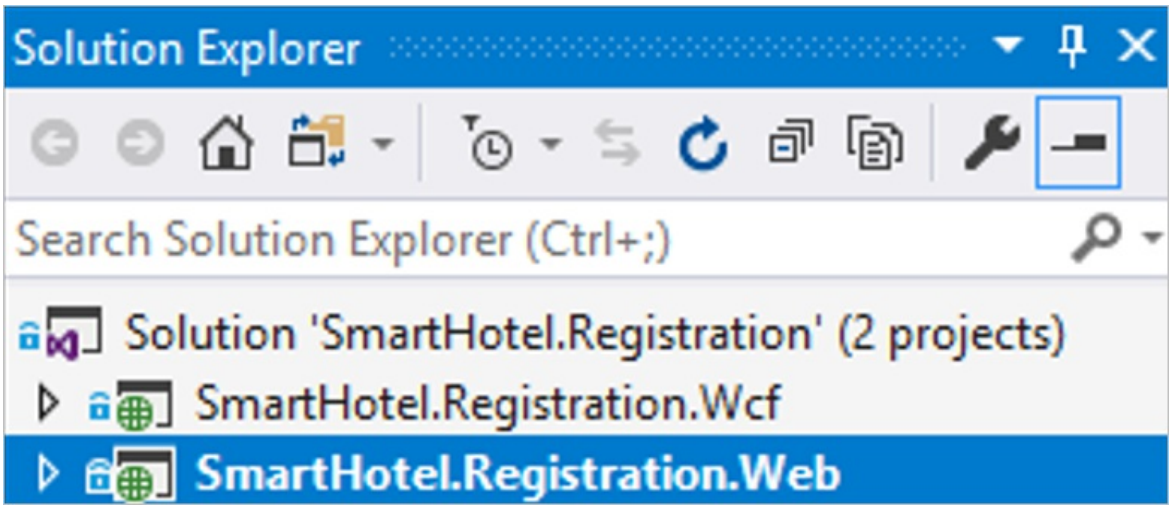
2. They import the Git repo that currently holds their application code. They download it from the public GitHub repository.



3. They connect Visual Studio to the repo and then clone the code to the developer machine by using Team Explorer.



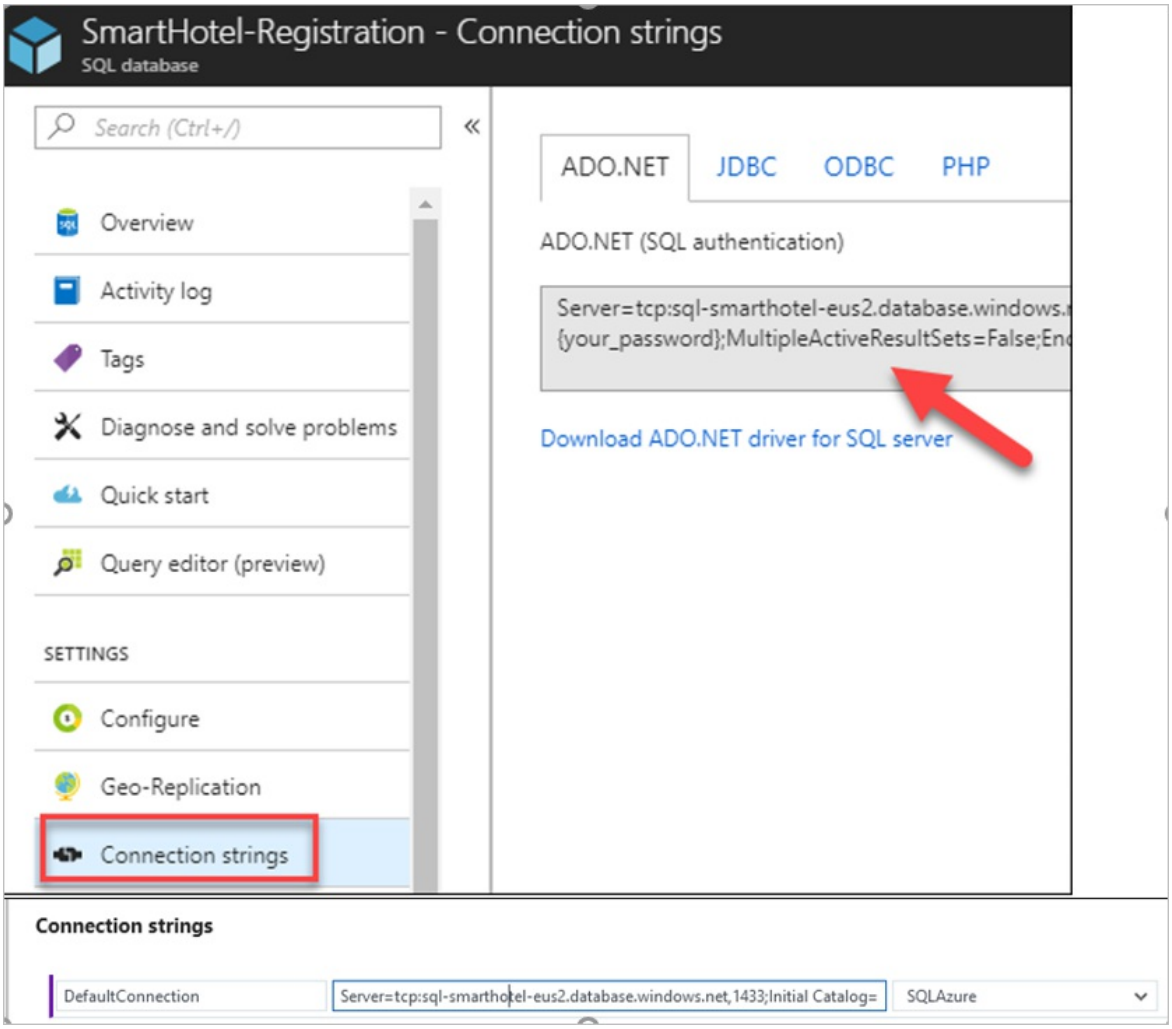
4. They open the solution file for the application. The web app and the WCF service have separate projects within the file.



Step 5: Configure connection strings

The Contoso admins make sure that the web apps and database can communicate with each other. To do this, they configure connection strings in the code and in the web apps.

1. In the web app for the WCF service, `SHWCF-EUS2`, under **Settings** > **Application settings**, they add a new connection string named **DefaultConnection**.
2. They pull the connection string from the `SmartHotel-Registration` database and then update it with the correct credentials.



3. In Visual Studio, the admins open the `SmartHotel.Registration.wcf` project from the solution file. In the project, they update the `connectionStrings` section of the `web.config` file with the connection string.

```
<connectionStrings>
  <add name="DefaultConnection" connectionString="Server=tcp:sql-smarthotel-eus2.
</connectionStrings>
```

4. They change the `client` section of the `web.config` file for `SmartHotel.Registration.Web` to point to the new location of the WCF service. This is the URL of the WCF web app that hosts the service endpoint.

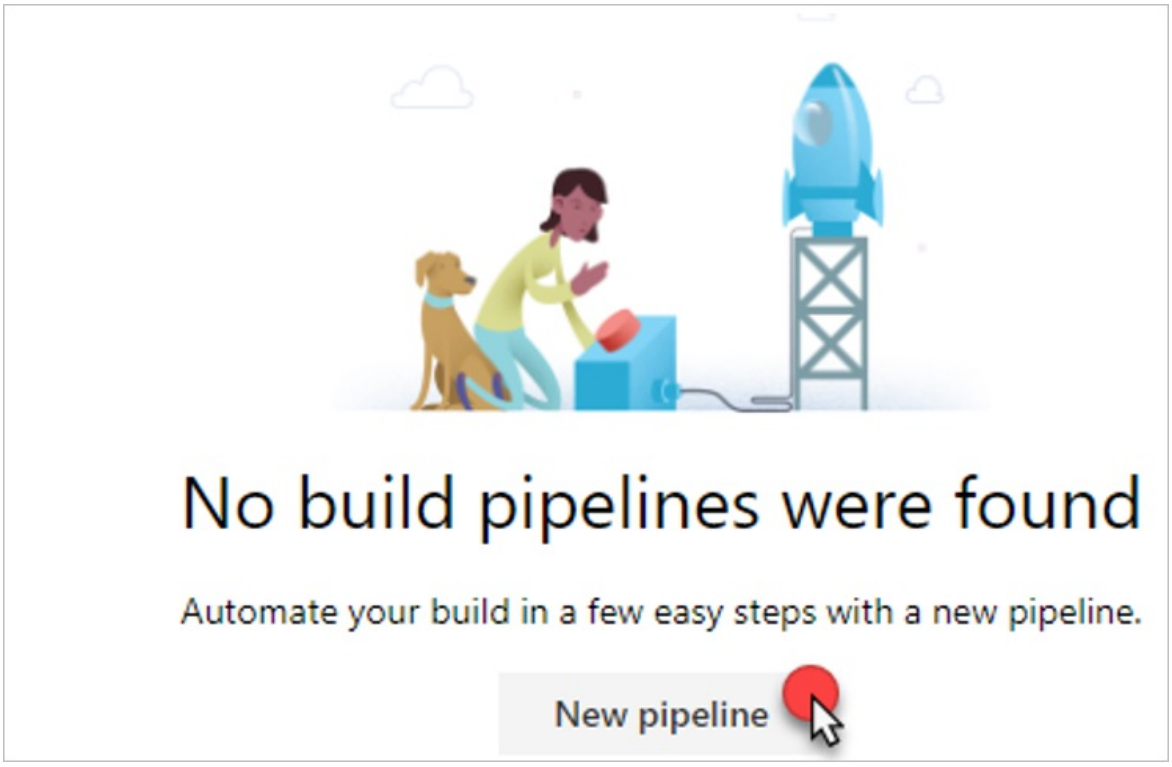
```
<client>
  <endpoint address="http://shwcf-eus2.azurewebsites.net/service.svc" binding="basicHttpBinding"
    bindingConfiguration="BasicHttpBinding_IService" contract="Data.IService"
    name="BasicHttpBinding_IService" />
</client>
```

5. With the code changes now in place, the admins commit and sync them by using Team Explorer in Visual Studio.

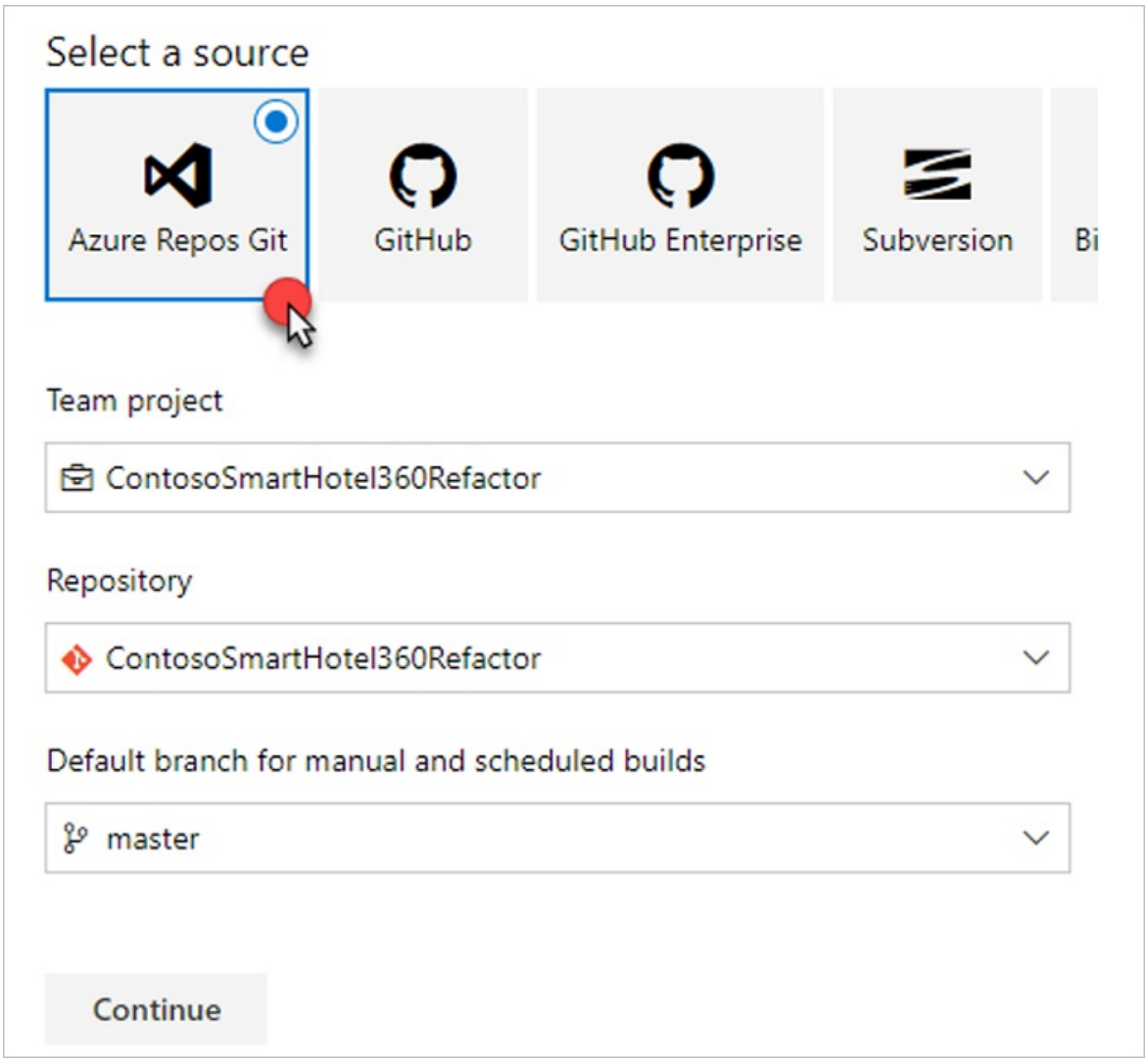
Step 6: Set up build and release pipelines in Azure DevOps

The Contoso admins now configure Azure DevOps to perform the build and release process.

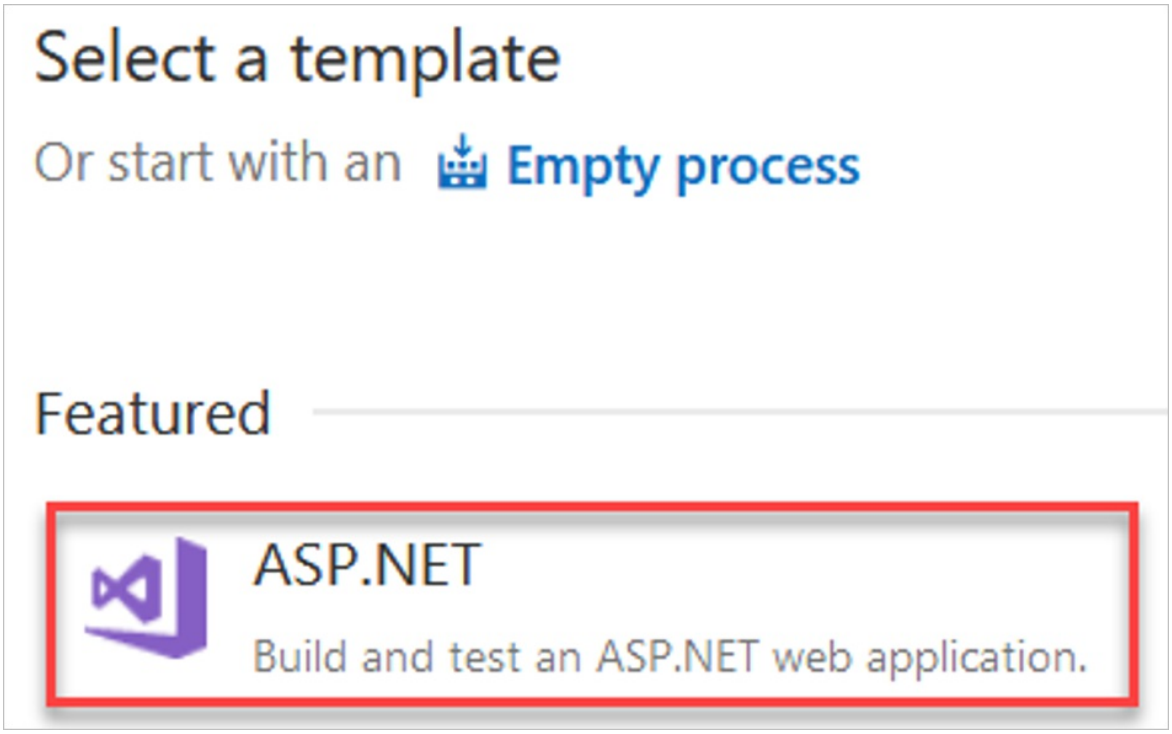
1. In Azure DevOps, they select **Build and release** > **New pipeline**.



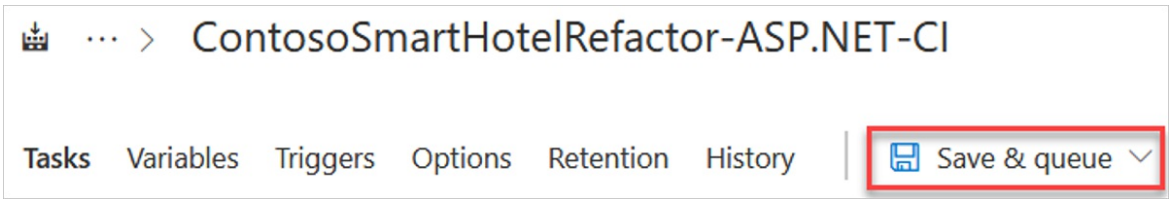
2. They select **Azure Repos Git** and, in the **Repository** drop-down list, they select the relevant repo.



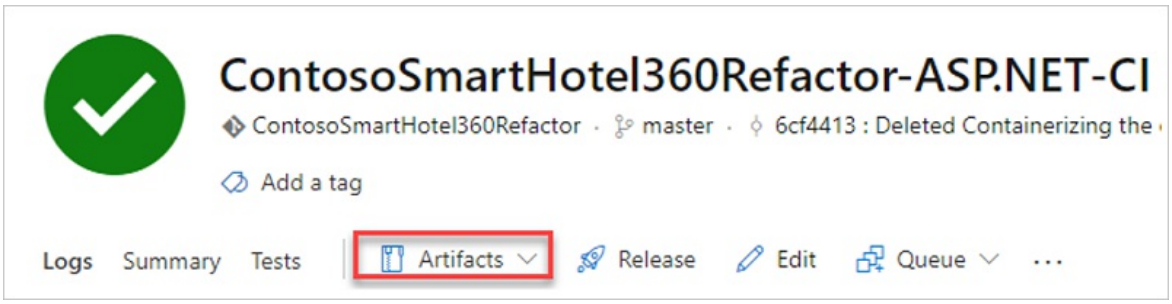
3. Under **Select a template**, they select the `ASP.NET` template for their build.



4. They use the name `ContosoSmartHotelRefactor-ASP.NET-CI` for the build and then select **Save & Queue**, which kicks off the first build.

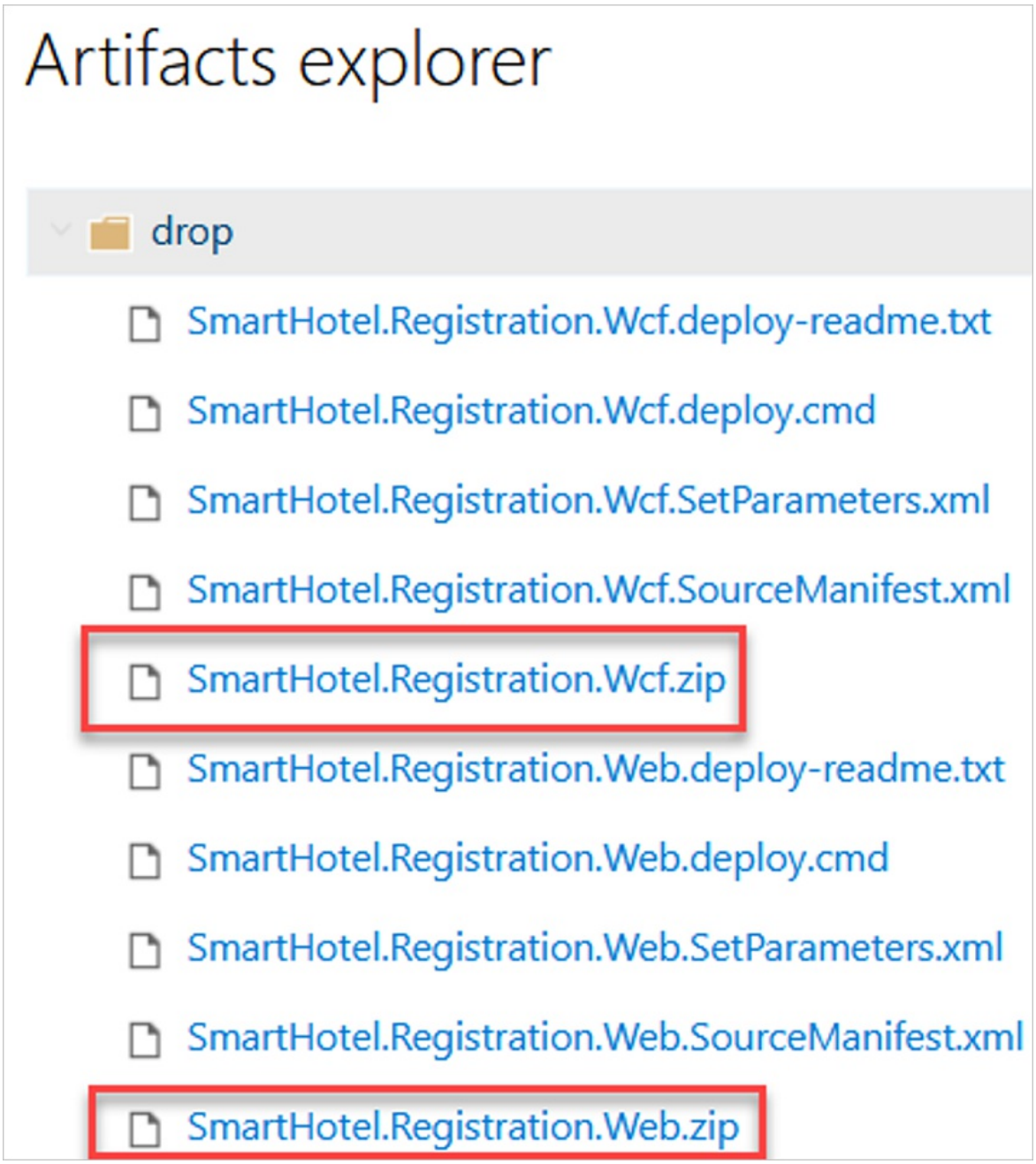


5. They select the build number to watch the process. After it's finished, the admins can see the process feedback, and they select **Artifacts** to review the build results.

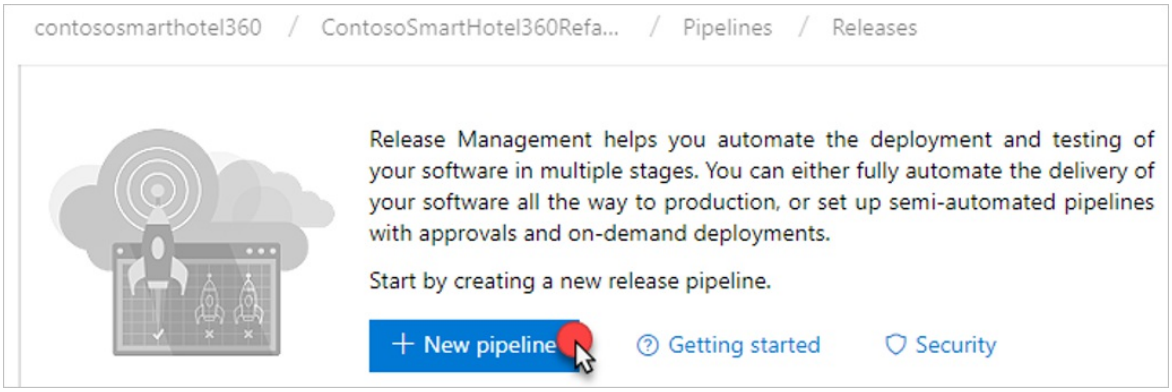


The **Artifacts explorer** pane opens, and the **drop** folder displays the build results.

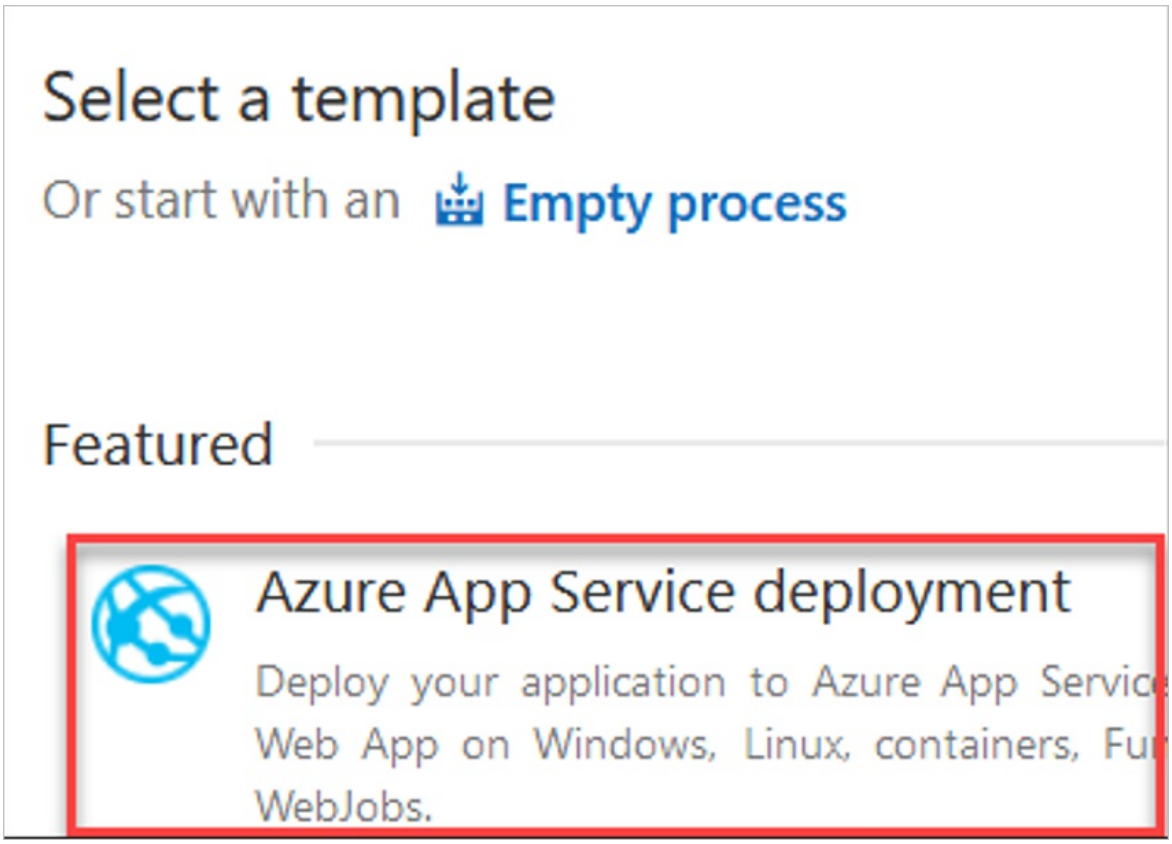
- The two .zip files are the packages that contain the applications.
- These .zip files are used in the release pipeline for deployment to Azure App Service.



6. They select **Releases** > **+ New pipeline**.



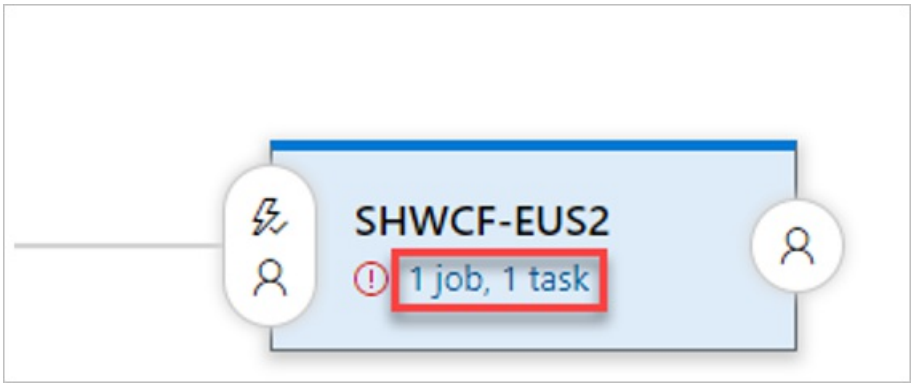
7. They select the deployment template for Azure App Service.



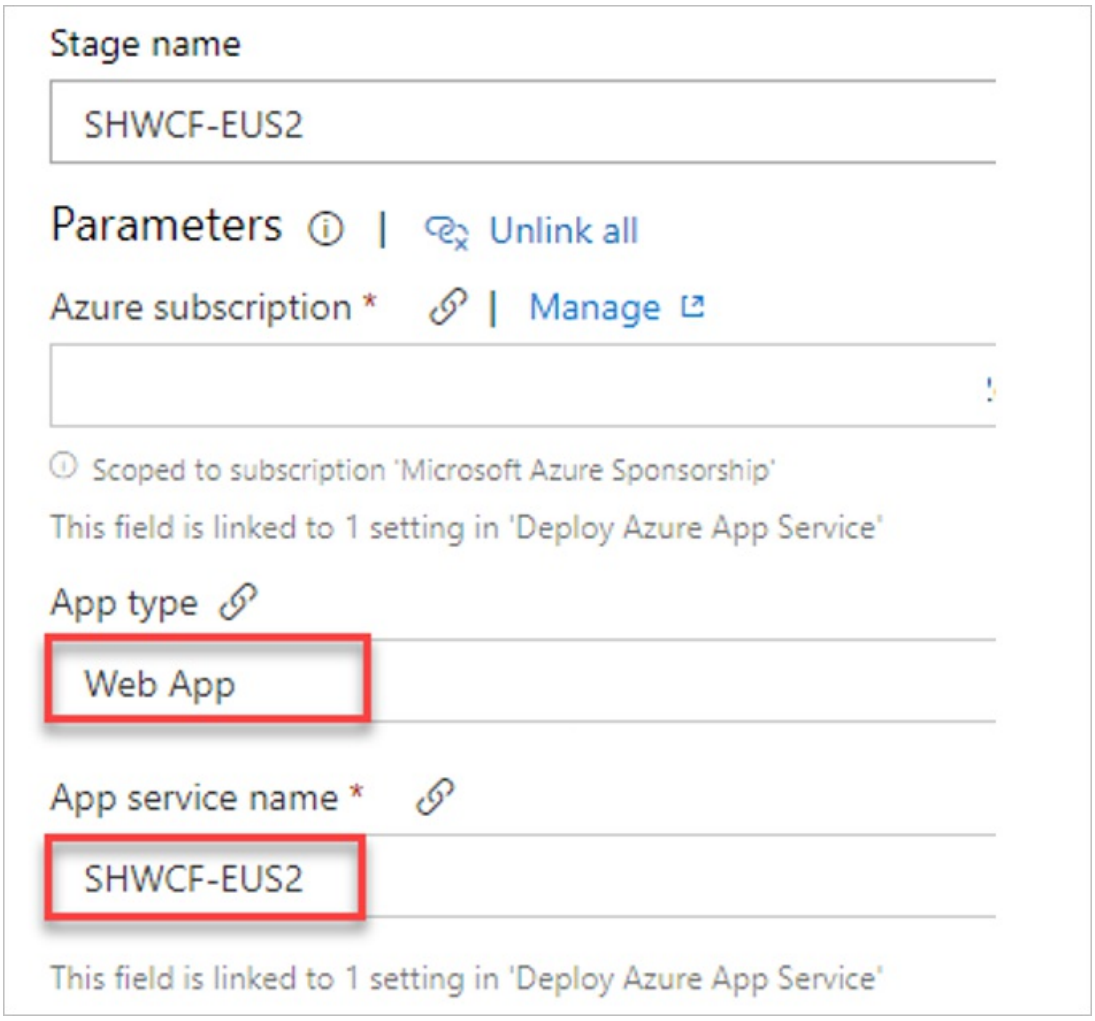
8. They name the release pipeline `ContosoSmartHotel360Refactor` and, in the **Stage name** box, specify `SHWCF-EUS2` as the name of the WCF web app.



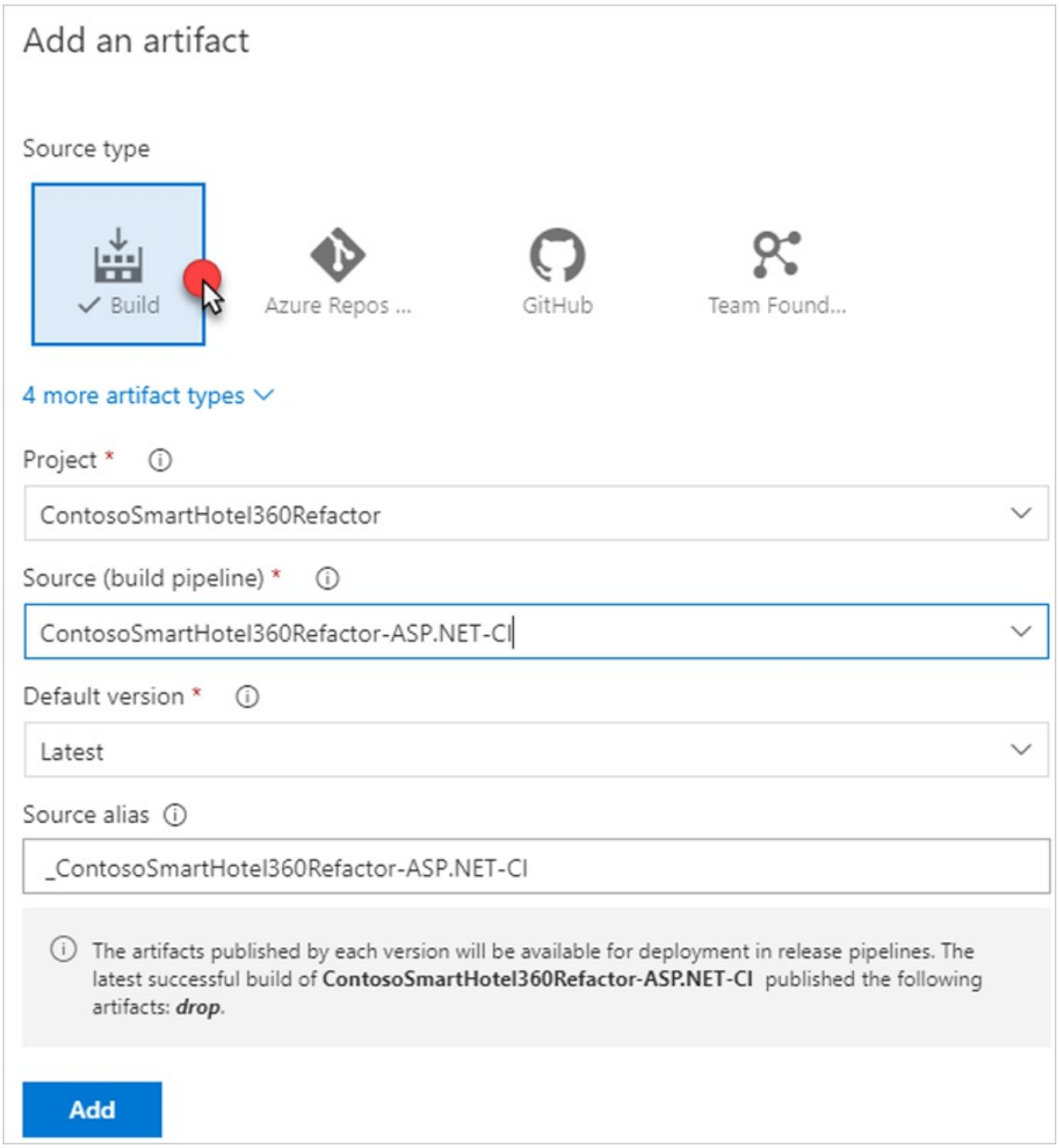
9. Under the stages, they select **1 job, 1 task** to configure deployment of the WCF service.



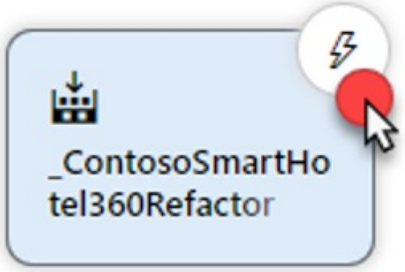
10. They verify that the subscription is selected and authorized, and then they select the **app service name**.



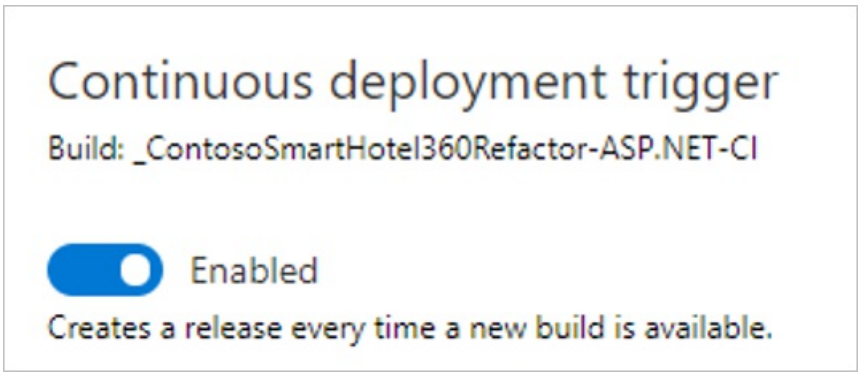
11. On the pipeline > **Artifacts**, they select **+ Add an artifact**, then select to build with the `ContosoSmarthotel360Refactor` pipeline.



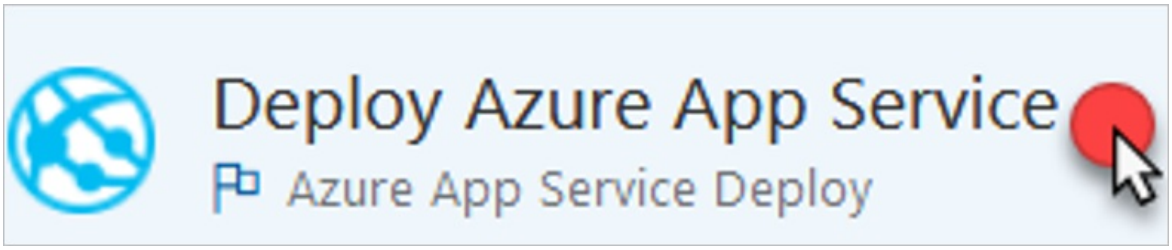
12. To enable the continuous deployment trigger, the admins select the lightning bolt icon on the artifact.



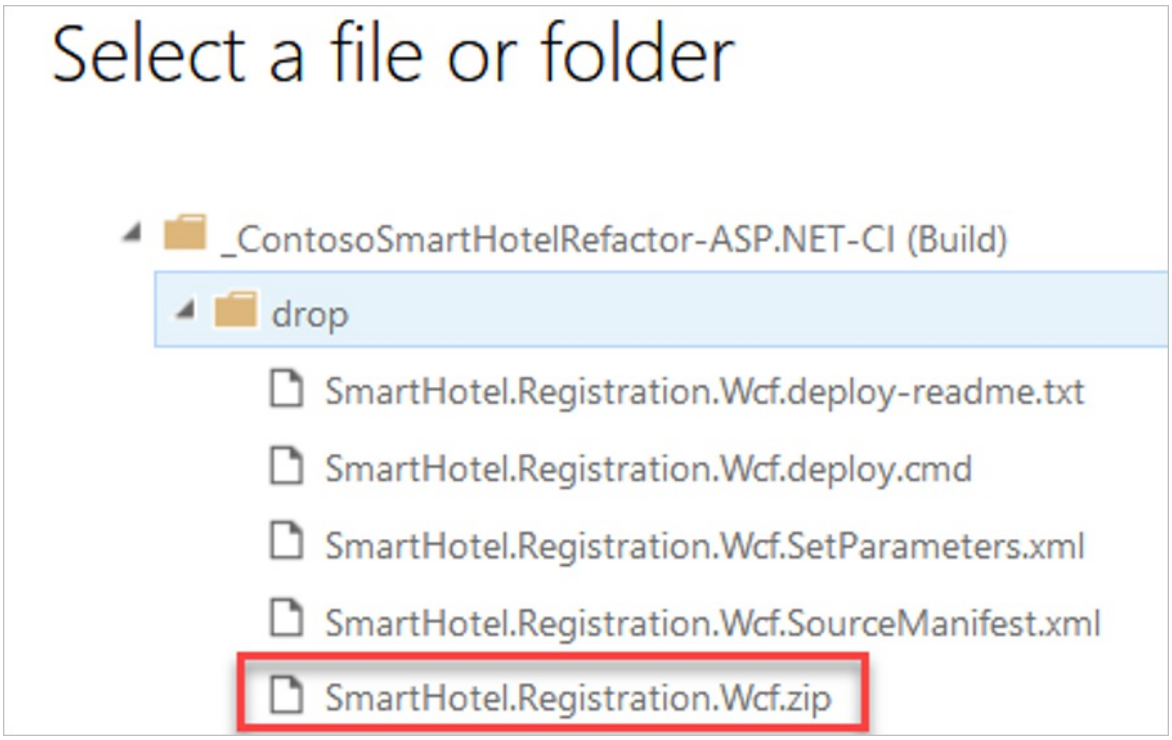
13. They set the continuous deployment trigger to **Enabled**.



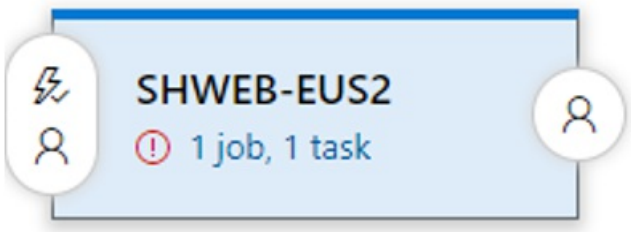
14. The admins go back to the stage **1 job, 1 task** and then select **Deploy Azure App Service**.



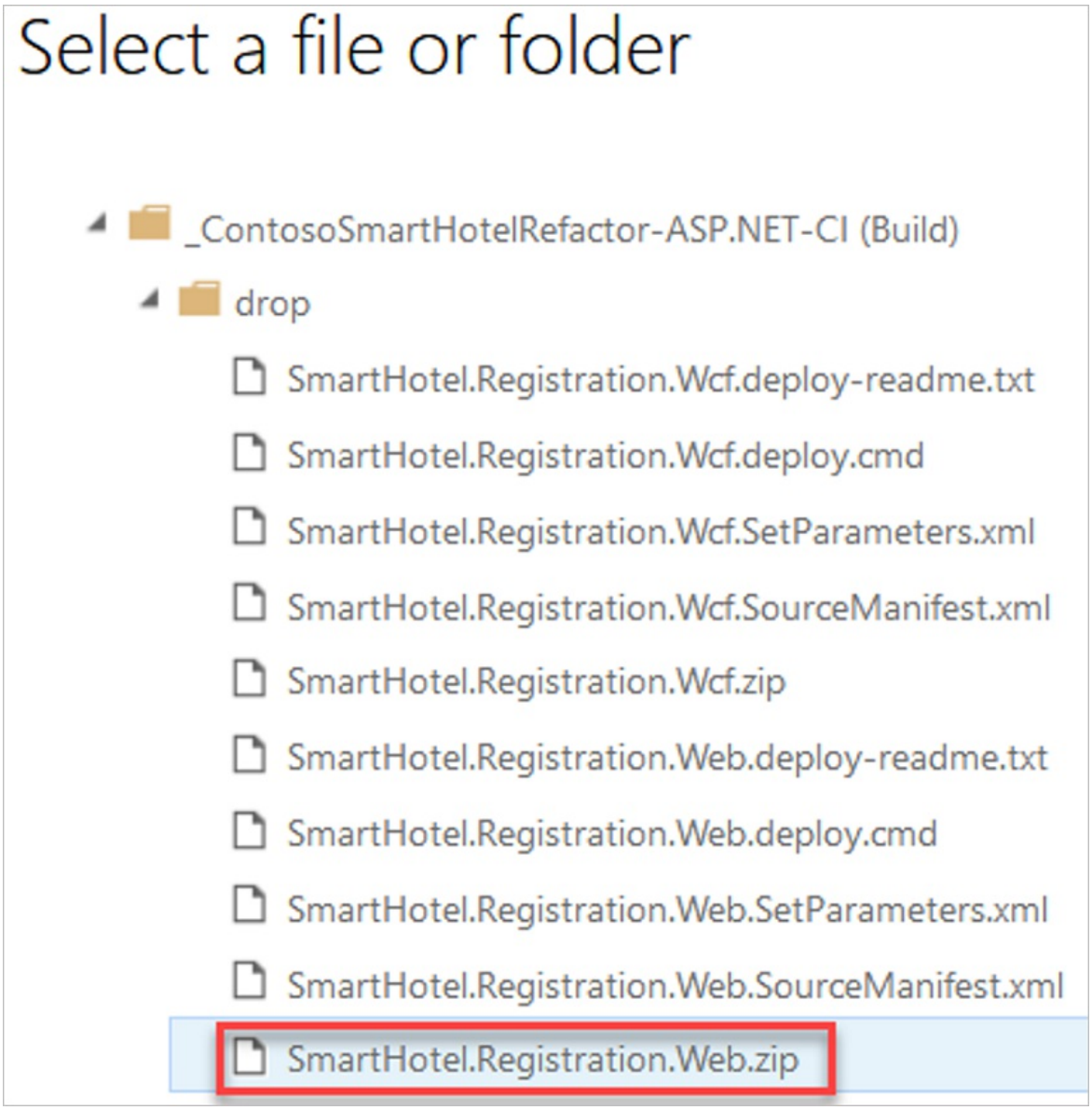
15. In **Select a file or folder**, they expand the **drop** folder, select the `SmartHotel.Registration.Wcf.zip` file that was created during the build, and then select **Save**.



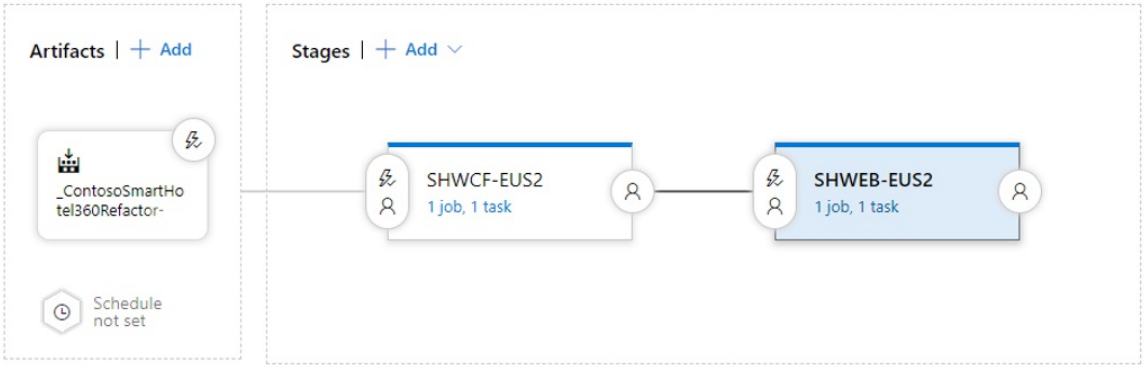
16. They select **Pipeline** > **Stages**, and then select **+ Add** to add an environment for `SHWEB-EUS2`. They select another Azure App Service deployment.



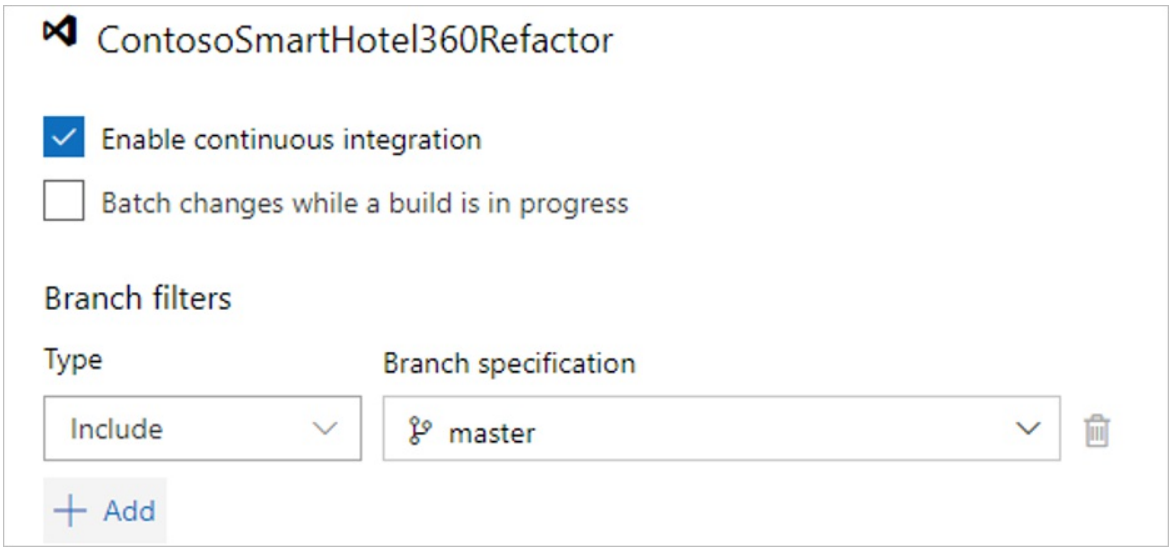
17. They repeat the process to publish the `SmartHotel.Registration.Web.zip` file to the correct web app, and then select **Save**.



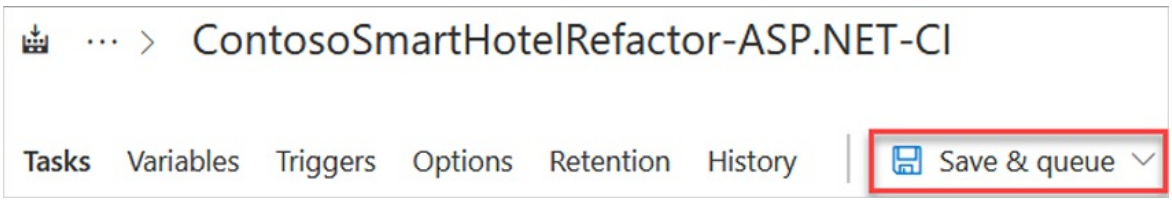
The release pipeline is displayed, as shown here:



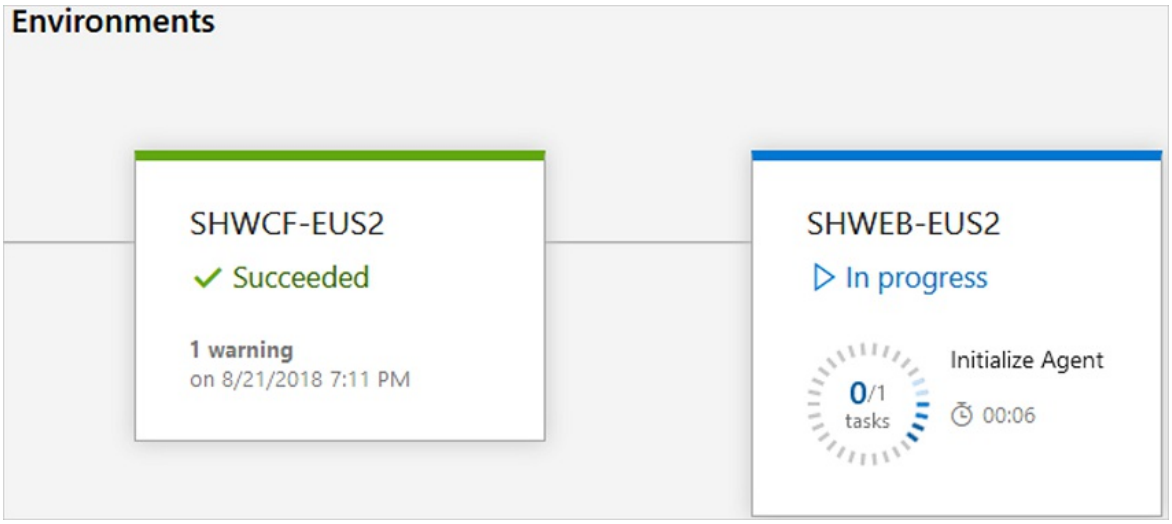
18. They go back to **Build**, select **Triggers**, and then select the **Enable continuous integration** check box. This action enables the pipeline so that when changes are committed to the code, the full build and release occur.



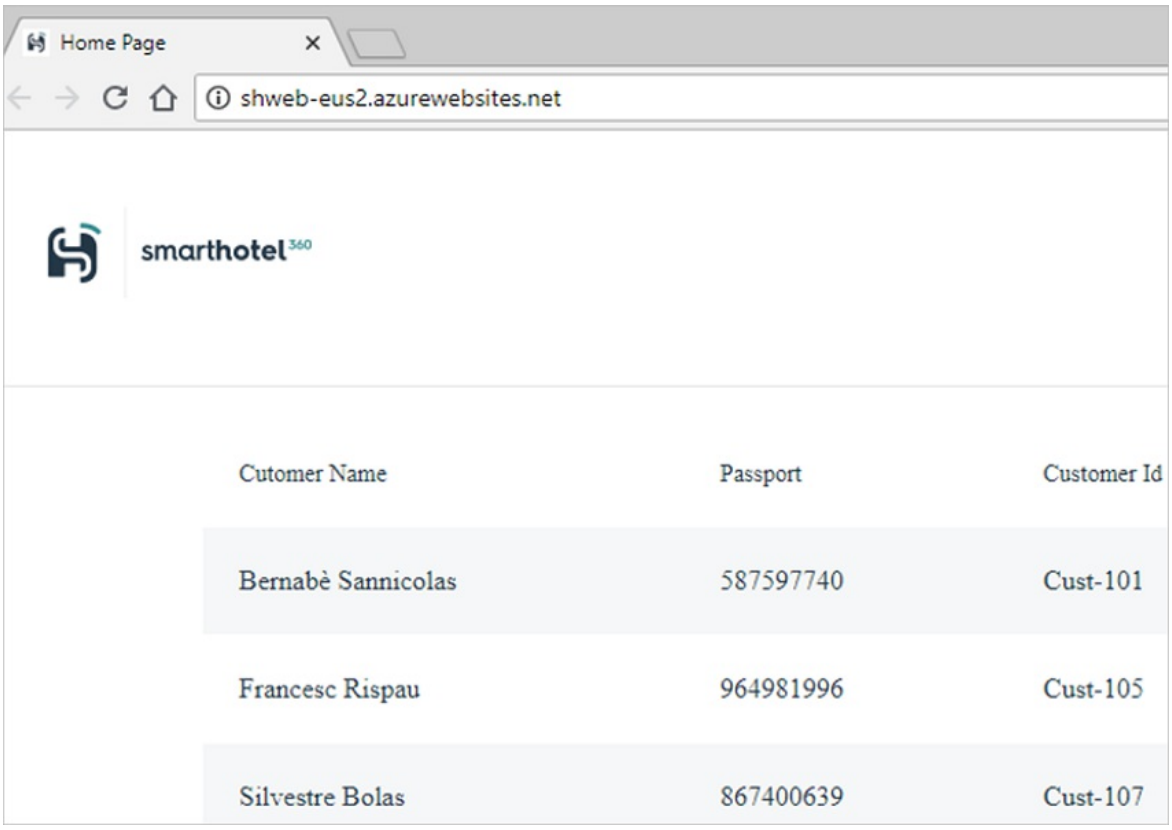
19. They select **Save & Queue** to run the full pipeline. A new build is triggered, which in turn creates the first release of the application to the Azure App Service.



20. Contoso admins can follow the build and release pipeline process from Azure DevOps. After the build finishes, the release starts.



21. After the pipeline finishes, both sites have been deployed, and the application is up and running online.



The application has been successfully migrated to Azure.

Clean up after migration

After migration, Contoso completes these cleanup steps:

- They remove the on-premises VMs from the vCenter inventory.
- They remove the VMs from the local backup jobs.
- They update their internal documentation to show the new locations for the SmartHotel360 application. The documentation shows the database as running in Azure SQL Database and the front end as running in two web apps.
- They review any resources that interact with the decommissioned VMs, and they update any relevant settings or documentation to reflect the new configuration.

Review the deployment

With the resources now migrated to Azure, Contoso needs to fully operationalize and help secure their new infrastructure.

Security

- Contoso helps ensure that their new `SmartHotel-Registration` database is secure. Learn more.
- In particular, Contoso updates the web apps to use SSL with certificates.

Backups

- The Contoso team reviews the backup requirements for the Azure SQL Database. Learn more.
- They also learn about managing SQL Database backups and restores. Learn more about automatic backups.
- They consider implementing failover groups to provide regional failover for the database. Learn more.
- For resilience, they consider deploying the web app in the main region (`East US 2`) and the secondary region (`Central US`). The team could configure Traffic Manager to ensure failover during regional outages.

Licensing and cost optimization

- After all resources are deployed, Contoso assigns Azure tags based on their infrastructure planning.
- All licensing is built into the cost of the PaaS services that Contoso is consuming. This cost is deducted from the Enterprise Agreement.
- Contoso will use Azure Cost Management + Billing to ensure that they stay within the budgets established by their IT leadership.

Conclusion

In this article, Contoso refactored the SmartHotel360 application in Azure by migrating the application front-end VM to two Azure App Service web apps. The application database was migrated to Azure SQL Database.

Refactor an on-premises application to an Azure App Service web app and a SQL managed instance

5/18/2021 • 18 minutes to read • Edit Online

This article demonstrates how the fictional company Contoso refactors a two-tier Windows .NET application that's running on VMware virtual machines (VMs) as part of a migration to Azure. The Contoso team migrates the application front-end VM to an Azure App Service web app. The article also shows how Contoso migrates the application database to an Azure SQL managed instance.

The SmartHotel360 application that we use in this example is provided as open-source software. If you want to use it for your own testing purposes, you can download it from GitHub.

Business drivers

The Contoso IT leadership team has worked closely with business partners to understand what they want to achieve with this migration:

- **Address business growth.** Contoso is growing, and there is pressure on their on-premises systems and infrastructure.
- **Increase efficiency.** Contoso needs to remove unnecessary procedures and streamline processes for developers and users. The business needs IT to be fast and not waste time or money, thus delivering faster on customer requirements.
- **Increase agility.** Contoso IT needs to be more responsive to the needs of the business. It must be able to react faster than the changes in the marketplace, to enable success in a global economy. Reaction time must not get in the way, or become a business blocker.
- **Scale.** As the business grows successfully, Contoso IT must provide systems that are able to grow at the same pace.
- **Reduce costs.** Contoso wants to minimize licensing costs.

Migration goals

To help determine the best migration method, the Contoso cloud team pinned down the following goals:

REQUIREMENTS	DETAILS

REQUIREMENTS	DETAILS
Application	The application in Azure will remain as critical as it is today on-premises. It should have the same performance capabilities as it currently does in VMware. The team doesn't want to invest in the application. For now, admins will simply move the application safely to the cloud. The team wants to stop supporting Windows Server 2008 R2, which the application currently runs on. The team also wants to move away from SQL Server 2008 R2 to a modern platform as a service (PaaS) database, which will minimize the need for management. Contoso wants to take advantage of its investment in SQL Server licensing and Software Assurance where possible. In addition, Contoso wants to mitigate the single point of failure on the web tier.
Limitations	The application consists of an ASP.NET application and a Windows Communication Foundation (WCF) service running on the same VM. They want to spread these components across two web apps using the Azure App Service.
Azure	Contoso wants to move the application to Azure, but they don't want to run it on VMs. Contoso wants to use Azure PaaS services for both the web and data tiers.
DevOps	Contoso wants to move to a DevOps model that uses Azure DevOps for their builds and release pipelines.

Solution design

After pinning down their goals and requirements, Contoso designs and reviews a deployment solution. They also identify the migration process, including the Azure services that they'll use for the migration.

Current application

- The SmartHotel360 on-premises application is tiered across two VMs, `WEBVM` and `SQLVM`.
- The VMs are located on VMware ESXi host contosohost1.contoso.com version 6.5.
- The VMware environment is managed by vCenter Server 6.5 (`vcenter.contoso.com`), which runs on a VM.
- Contoso has an on-premises datacenter (contoso-datacenter), with an on-premises domain controller (contosodc1).
- The on-premises VMs in the Contoso datacenter will be decommissioned after the migration is done.

Proposed solution

- For the application web tier, Contoso has decided to use Azure App Service. This PaaS service enables them to deploy the application with just a few configuration changes. Contoso will use Visual Studio to make the change, and they'll deploy two web apps, one for the website and one for the WCF service.
- To meet requirements for a DevOps pipeline, Contoso will use Azure DevOps for source code management with Git repos. They'll use automated builds and release to build the code and deploy it to the Azure App Service.

Database considerations

As part of the solution design process, Contoso did a feature comparison between Azure SQL Database and SQL Managed Instance. They decided to use SQL Managed Instance based on the following considerations:

- SQL Managed Instance aims to deliver almost 100 percent compatibility with the latest on-premises SQL Server version. Microsoft recommends SQL Managed Instance for customers who are running SQL Server on-premises or on infrastructure as a service (IaaS) VMs who want to migrate their applications to a fully managed service with minimal design changes.
- Contoso is planning to migrate a large number of applications from on-premises to IaaS VMs. Many of these VMs are provided by independent software vendors. Contoso realizes that using SQL Managed Instance will help ensure database compatibility for these applications. They'll use SQL Managed Instance rather than SQL Database, which might not be supported.
- Contoso can simply do a lift and shift migration to SQL Managed Instance by using the fully automated Azure Database Migration Service. With this service in place, Contoso can reuse it for future database migrations.
- SQL Managed Instance supports SQL Server Agent, an important component of the SmartHotel360 application. Contoso needs this compatibility; otherwise, they'll have to redesign the maintenance plans required by the application.
- With Software Assurance, Contoso can exchange their existing licenses for discounted rates on a SQL managed instance by using the Azure Hybrid Benefit for SQL Server. This allows Contoso to save up to 30 percent by using SQL Managed Instance.
- The SQL managed instance is fully contained in the virtual network, so it provides greater isolation and security for Contoso's data. Contoso can get the benefits of the public cloud, while keeping the environment isolated from the public internet.
- SQL Managed Instance supports many security features, including always-encrypted, dynamic data masking, row-level security, and threat detection.

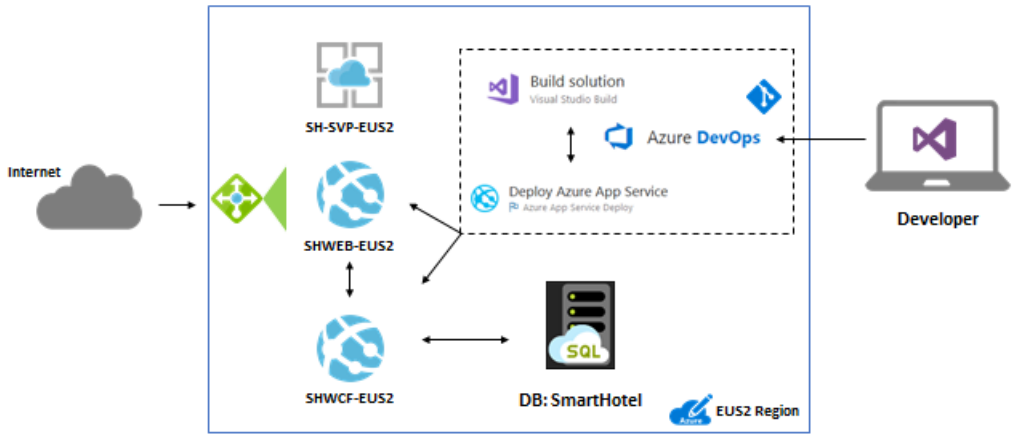
Solution review

Contoso evaluates their proposed design by putting together a pros and cons list, as shown in the following table:

CONSIDERATION	DETAILS

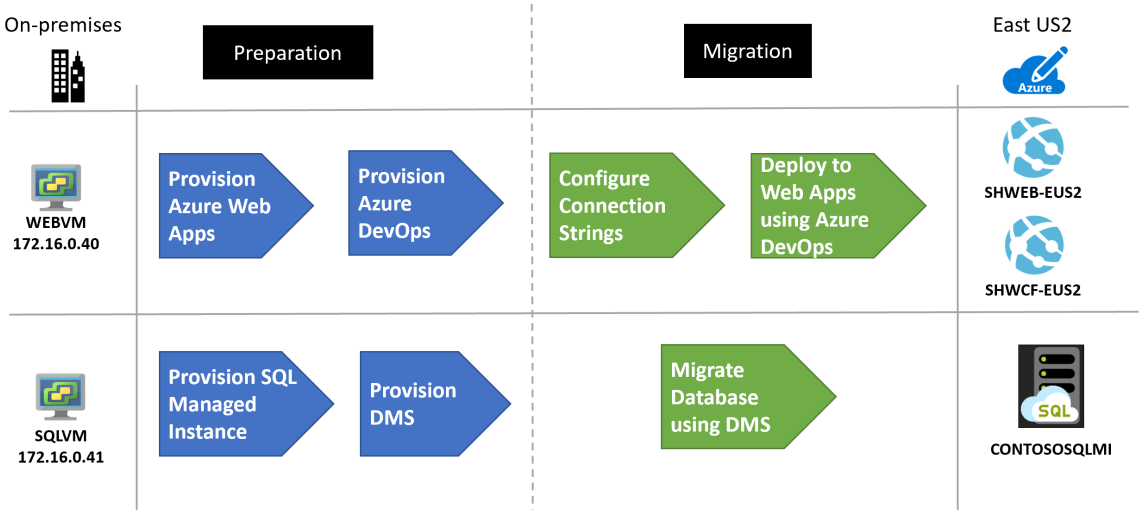
CONSIDERATION	DETAILS
Pros	The SmartHotel360 application code doesn't require changes for migration to Azure. Contoso can take advantage of their investment in Software Assurance by using the Azure Hybrid Benefit for both SQL Server and Windows Server. After the migration, Windows Server 2008 R2 won't need to be supported. For more information, see the Microsoft Lifecycle Policy. Contoso can configure the web tier of the application with multiple instances, so that the web tier is no longer a single point of failure. The database will no longer depend on the aging SQL Server 2008 R2. SQL Managed Instance supports Contoso's technical requirements and goals. The SQL managed instance will provide 100 percent compatibility with their current deployment, while moving them away from SQL Server 2008 R2. Contoso can take advantage of their investment in Software Assurance and using the Azure Hybrid Benefit for SQL Server and Windows Server. They can reuse Azure Database Migration Service for additional future migrations. The SQL managed instance has built-in fault tolerance that Contoso doesn't need to configure. This ensures that the data tier is no longer a single point of failover.
Cons	Azure App Service supports only one application deployment for each web app. This means that two web apps must be provisioned, one for the website and one for the WCF service. For the data tier, SQL Managed Instance might not be the best solution if Contoso wants to customize the operating system or the database server, or if they want to run third-party applications along with SQL Server. Running SQL Server on an IaaS VM could provide this flexibility.

Proposed architecture



Migration process

1. Contoso provisions an Azure SQL managed instance and then migrates the SmartHotel360 database to it by using Azure Database Migration Service.
2. Contoso provisions and configures web apps and deploys the SmartHotel360 application to them.



Azure services

SERVICE	DESCRIPTION	COST
Azure App Service Migration Assistant	A free and simple path to seamlessly migrate .NET web applications from on-premises to the cloud with minimal to no code changes.	It's a downloadable tool, free of charge.
Azure Database Migration Service	Azure Database Migration Service enables seamless migration from multiple database sources to Azure data platforms with minimal downtime.	Learn about supported regions and Azure Database Migration Service pricing.

SERVICE	DESCRIPTION	COST
Azure SQL Managed Instance	SQL Managed Instance is a managed database service that represents a fully managed SQL Server instance in Azure. It uses the same code as the latest version of SQL Server Database Engine, and has the latest features, performance improvements, and security patches.	Using a SQL managed instance that runs in Azure incurs charges based on capacity. Learn more about SQL Managed Instance pricing.
Azure App Service	Helps create powerful cloud applications that use a fully managed platform.	Pricing is based on size, location, and usage duration. Learn more.
Azure Pipelines	Provides a continuous integration and continuous deployment (CI/CD) pipeline for application development. The pipeline starts with a Git repository for managing application code, a build system for producing packages and other build artifacts, and a release management system to deploy changes in dev, test, and production environments.	

Prerequisites

To run this scenario, Contoso must meet the following prerequisites:

REQUIREMENTS	DETAILS
Azure subscription	Contoso created subscriptions earlier in this article series. If you don't have an Azure subscription, create a free account. If you create a free account, you're the administrator of your subscription and can perform all actions. If you use an existing subscription and you're not the administrator, you need to work with the admin to assign you Owner or Contributor permissions.
Azure infrastructure	Contoso set up their Azure infrastructure as described in Azure infrastructure for migration.

Scenario steps

Here's how Contoso will run the migration:

- **Step 1: Assess and migrate the web apps.**. Contoso uses the Azure App Service Migration Assistant tool to run pre-migration compatibility checks and migrate their web apps to Azure App Service.
- **Step 2: Set up a SQL managed instance**. Contoso needs an existing managed instance to which the on-premises SQL Server database will migrate.
- **Step 3: Migrate via Azure Database Migration Service**. Contoso migrates the application database via Azure Database Migration Service.
- **Step 4: Set up Azure DevOps**. Contoso creates a new Azure DevOps project and imports the Git repo.
- **Step 5: Configure connection strings**. Contoso configures connection strings so that the web tier web

app, the WCF service web app, and the SQL managed instance can communicate.

- **Step 6: Set up build and release pipelines in Azure DevOps**. As a final step, Contoso sets up build and release pipelines in Azure DevOps to create the application. The team then deploys the pipelines to two separate web apps.

Step 1: Assess and migrate the web apps

Contoso admins assess and migrate their web app using the Azure App Service Migration Assistant tool. They use the Microsoft Learn path as a guide during the process. In brief, the admins perform the following actions:

- They use the Azure App Service migration assessment tool to evaluate any dependencies between their web apps and to determine whether there are any incompatibilities between their on-premises web apps and what's supported on Azure App Service.
- They download the Azure App Service Migration Assistant and sign in to their Azure account.
- They choose a subscription, a resource group, and the website's domain name.

Step 2: Set up a SQL managed instance

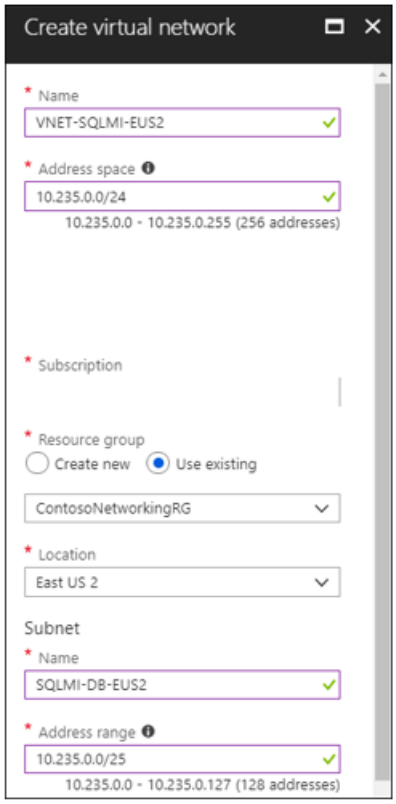
To set up an Azure SQL managed instance, Contoso needs a subnet that meets the following requirements:

- The subnet must be dedicated. It must be empty, and it can't contain any other cloud service. The subnet can't be a gateway subnet.
- After the managed instance is created, Contoso should not add resources to the subnet.
- The subnet can't have a network security group associated with it.
- The subnet must have a user-defined route table. The only route assigned should be `0.0.0.0/0` next-hop internet.
- If an optional custom DNS is specified for the virtual network, the virtual IP address `168.63.129.16` for the recursive resolvers in Azure must be added to the list. Learn how to configure a custom DNS for an Azure SQL managed instance.
- The subnet must not have a service endpoint (storage or SQL) associated with it. Service endpoints should be disabled on the virtual network.
- The subnet must have a minimum of 16 IP addresses. Learn how to size the managed instance subnet.
- In Contoso's hybrid environment, custom DNS settings are required. Contoso configures DNS settings to use one or more of the company's Azure DNS servers. Learn more about DNS customization.

Set up a virtual network for the managed instance

Contoso admins set up the virtual network as follows:

1. They create a new virtual network (`VNET-SQLMI-EU2`) in the primary region (`East US 2`). It adds the virtual network to the `ContosoNetworkingRG` resource group.
2. They assign an address space of `10.235.0.0/24` . They ensure that the range doesn't overlap with any other networks in its enterprise.
3. They add two subnets to the network:
   - `SQLMI-DS-EUS2` (`10.235.0.0/25`).
   - `SQLMI-SAW-EUS2` (`10.235.0.128/29`). This subnet is used to attach a directory to the managed instance.



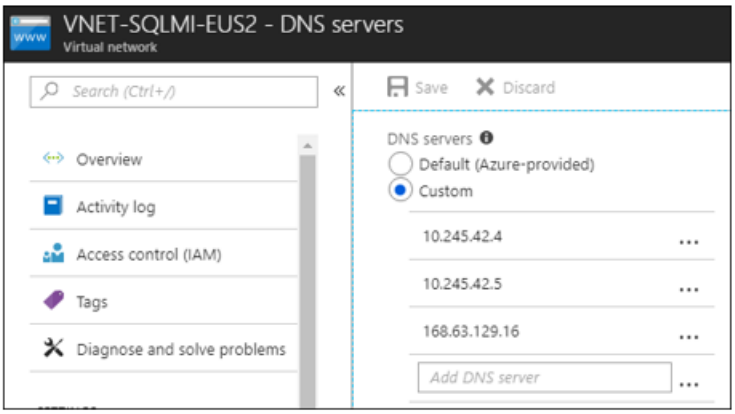
4. After the virtual network and subnets are deployed, they peer networks as follows:

   - Peers `VNET-SQLMI-EUS2` with `VNET-HUB-EUS2` (the hub virtual network for `East US 2`).
   - Peers `VNET-SQLMI-EUS2` with `VNET-PROD-EUS2` (the production network).

NAME	PEERING STATUS	PEER
VNET-SQLMI-EUS2-to-VNET-PROD-EUS2	Connected	VNET-PROD-EUS2
VNET-SQLMI-EUS2-to-VNET-HUB-EUS2	Connected	VNET-HUB-EUS2

5. They set custom DNS settings. The DNS settings point first to Contoso's Azure domain controllers. Azure DNS is secondary. The Contoso Azure domain controllers are located as follows:

   - Located in the `PROD-DC-EUS2` subnet of the production network (`VNET-PROD-EUS2`) in the `East US 2` region.
   - `CONTOSODC3` address: `10.245.42.4`
   - `CONTOSODC4` address: `10.245.42.5`
   - Azure DNS resolver: `168.63.129.16`



Need more help?

- Read the SQL Managed Instance overview.
- Learn how to create a virtual network for a SQL managed instance.
- Learn how to set up peering.
- Learn how to update Azure Active Directory DNS settings.

Set up routing

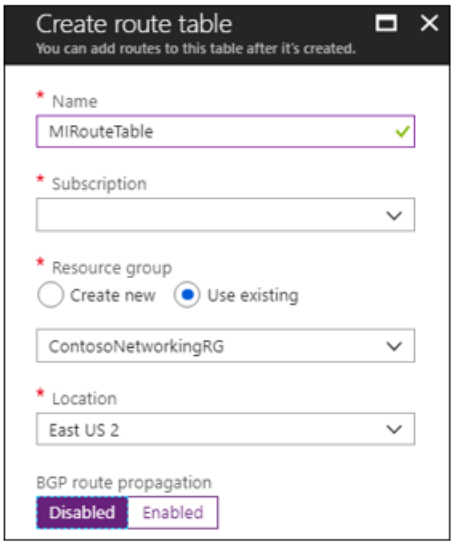
The managed instance is placed in a private virtual network. Contoso needs a route table for the virtual network to communicate with the Azure management service. If the virtual network can't communicate with the service that manages it, the virtual network becomes inaccessible.

Contoso considers these factors:

- The route table contains a set of rules (routes) that specify how packets that are sent from the managed instance should be routed in the virtual network.
- The route table is associated with subnets where managed instances are deployed. Each packet that leaves a subnet is handled based on the associated route table.
- A subnet can be associated with only one route table.
- There are no additional charges for creating route tables in Microsoft Azure.

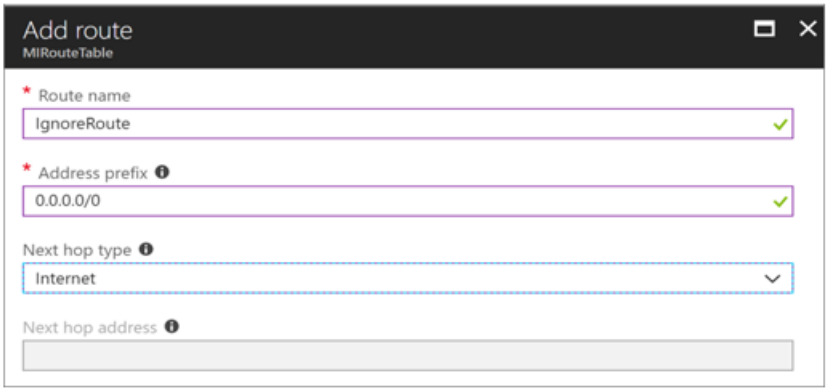
To set up routing, Contoso admins do the following:

1. They create a user-defined route table in the `ContosoNetworkingRG` resource group.

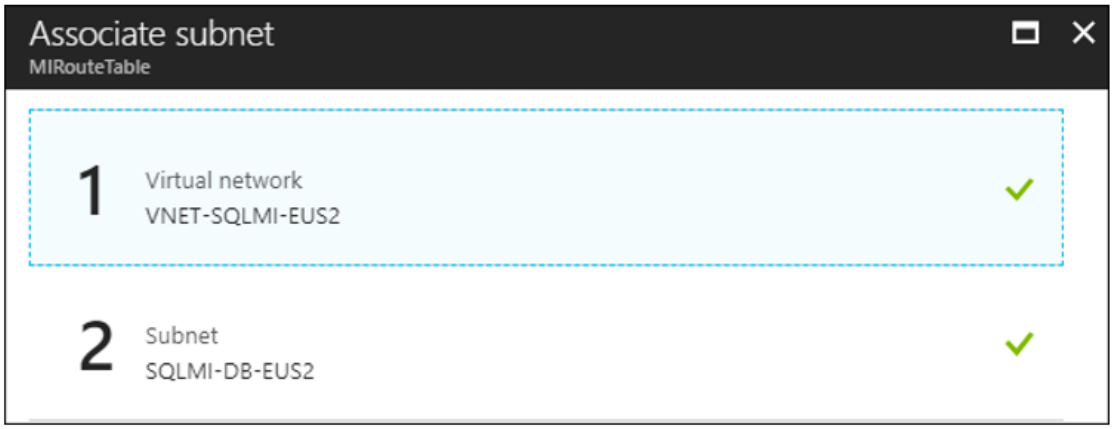


2. To comply with SQL Managed Instance requirements, after the route table (`MIRouteTable`) is deployed, the admins add a route with an address prefix of `0.0.0.0/0`. The **Next hop type** option is set to

Internet.



3. They associate the route table with the `SQLMI-DB-EUS2` subnet (in the `VNET-SQLMI-EUS2` network).



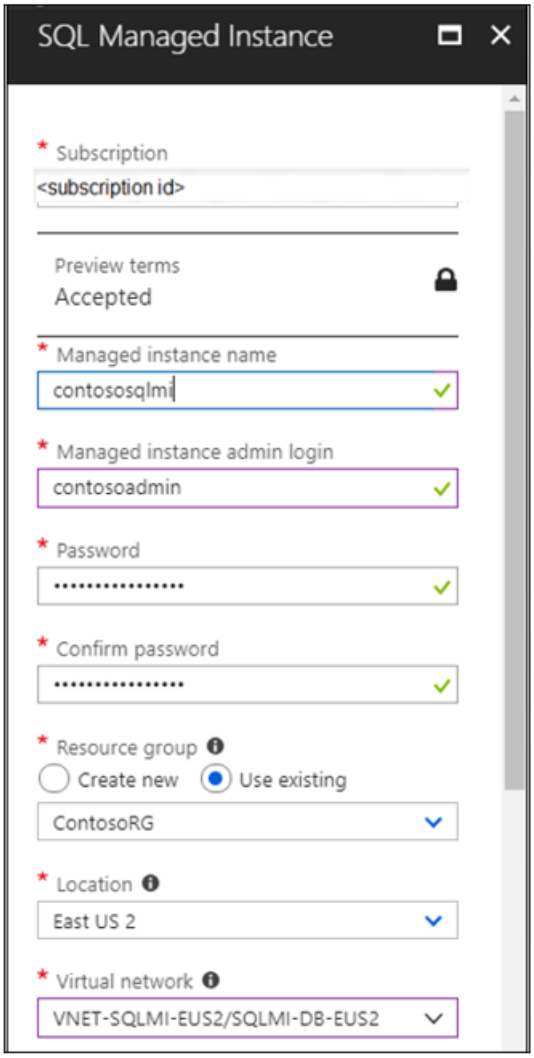
Need more help?

Learn how to set up routes for a managed instance.

Create a managed instance

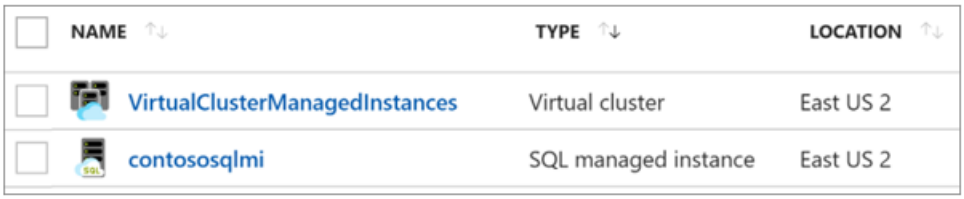
Now, Contoso admins provision a SQL managed instance by doing the following:

1. Because the managed instance serves a business application, the admins deploy the managed instance in the company's primary region (East US 2). They add the managed instance to the `ContosoRG` resource group.
2. They select a pricing tier, size compute, and storage for the instance. Learn more about SQL Managed Instance pricing.



After the managed instance is deployed, two new resources appear in the `ContosoRG` resource group:

- The SQL managed instance.
- A virtual cluster, in case Contoso has multiple managed instances.



Need more help?

Learn how to provision a managed instance.

Step 3: Migrate via Azure Database Migration Service

Contoso admins migrate the managed instance via Azure Database Migration Service by following the instructions in the step-by-step migration tutorial. They can perform online, offline, and hybrid (preview) migrations.

In brief, Contoso admins do the following:

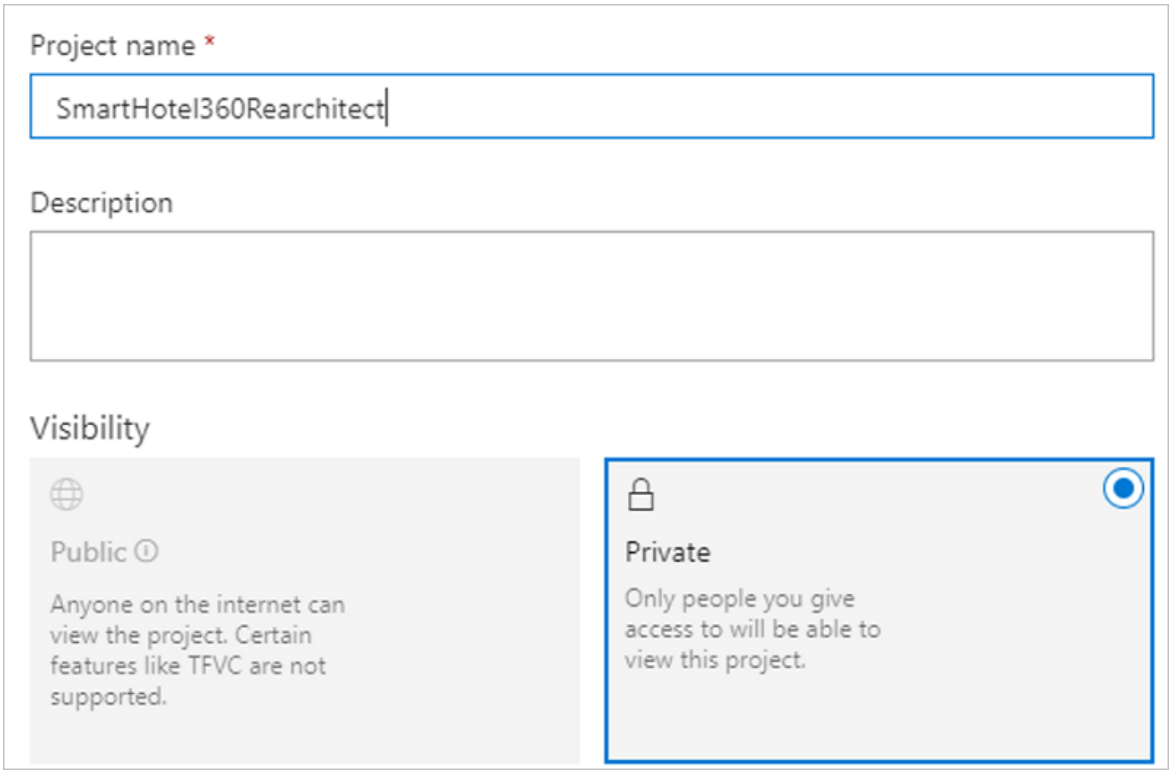
- They create an Azure Database Migration Service instance with a Premium SKU that's connected to the virtual network.

- They ensure that Database Migration Service can access the remote SQL Server via the virtual network. This would entail ensuring that all incoming ports are allowed from Azure to SQL Server at the virtual network level, the network VPN, and the machine that hosts SQL Server.
- They configure Azure Database Migration Service:
  - Create a migration project.
  - Add a source (on-premises database).
  - Select a target.
  - Select the databases to migrate.
  - Configure advanced settings.
  - Start the replication.
  - Resolve any errors.
  - Perform the final cutover.

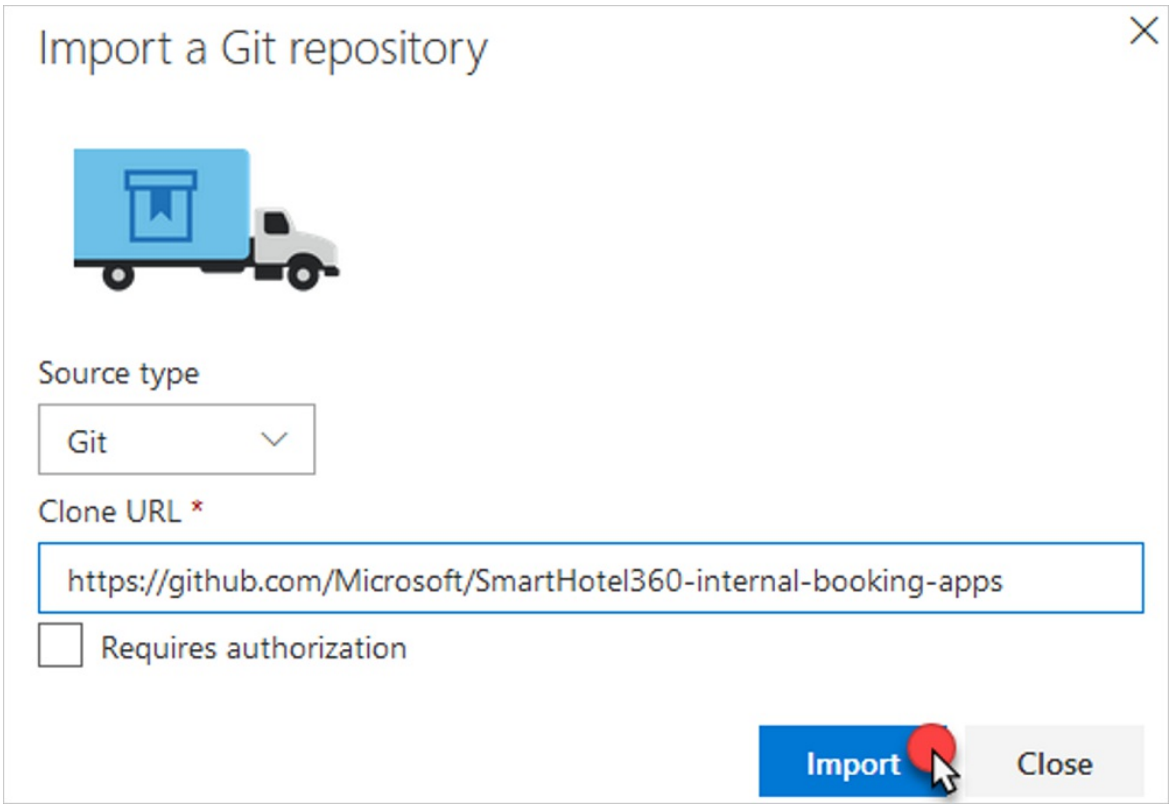
Step 4: Set up Azure DevOps

Contoso needs to build the DevOps infrastructure and pipelines for the application. To do this, the Contoso admins create a new DevOps project, import the code, and then set up build and release pipelines.

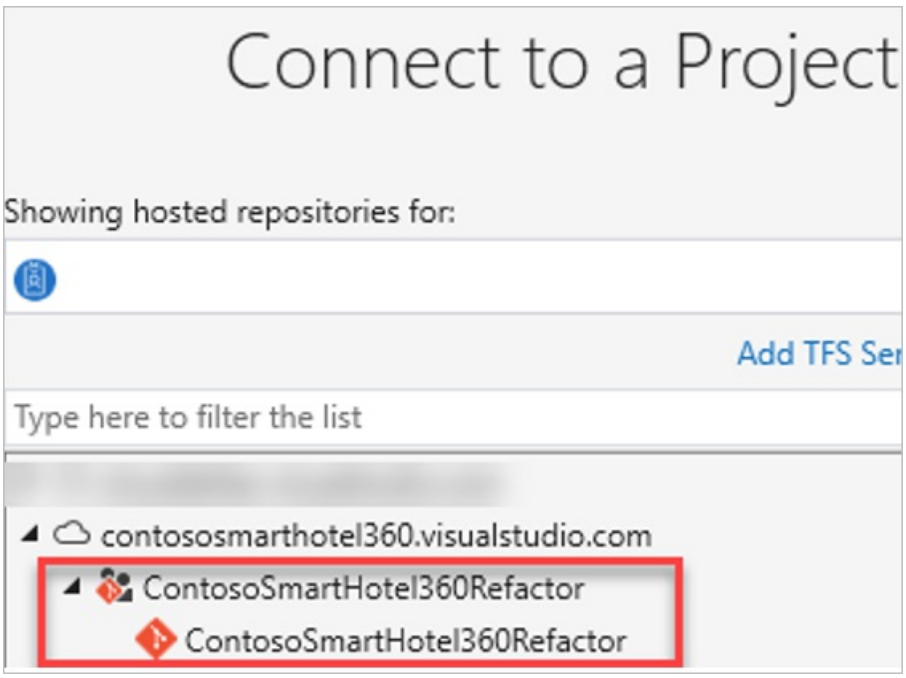
1. In the Contoso Azure DevOps account, they create a new project, `ContosoSmartHotelRefactor`, and then select **Git** for version control.



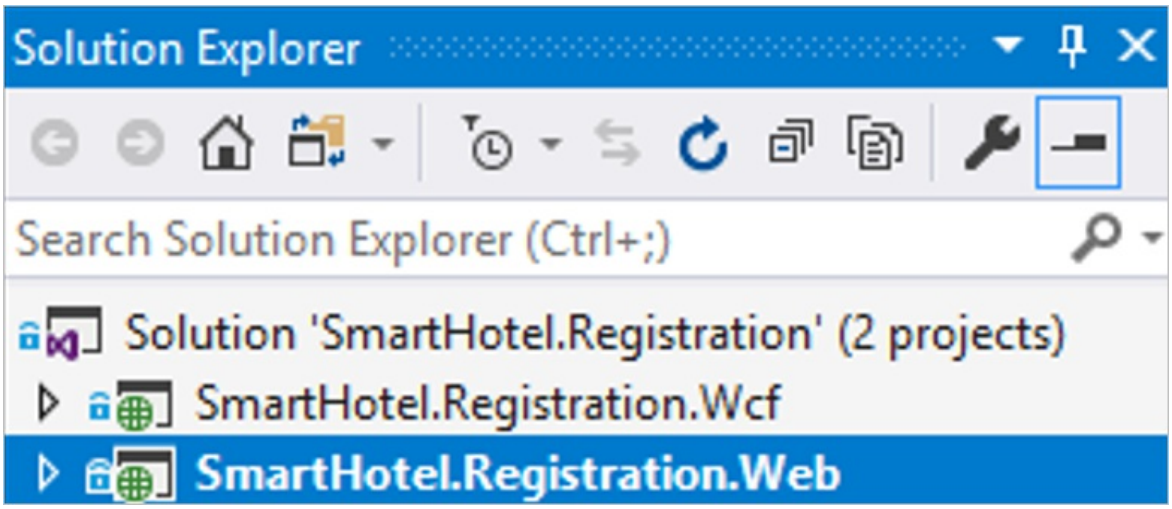
2. They import the Git repo that currently holds their application code. They download it from the public GitHub repository.



3. They connect Visual Studio to the repo and then clone the code to the developer machine by using Team Explorer.



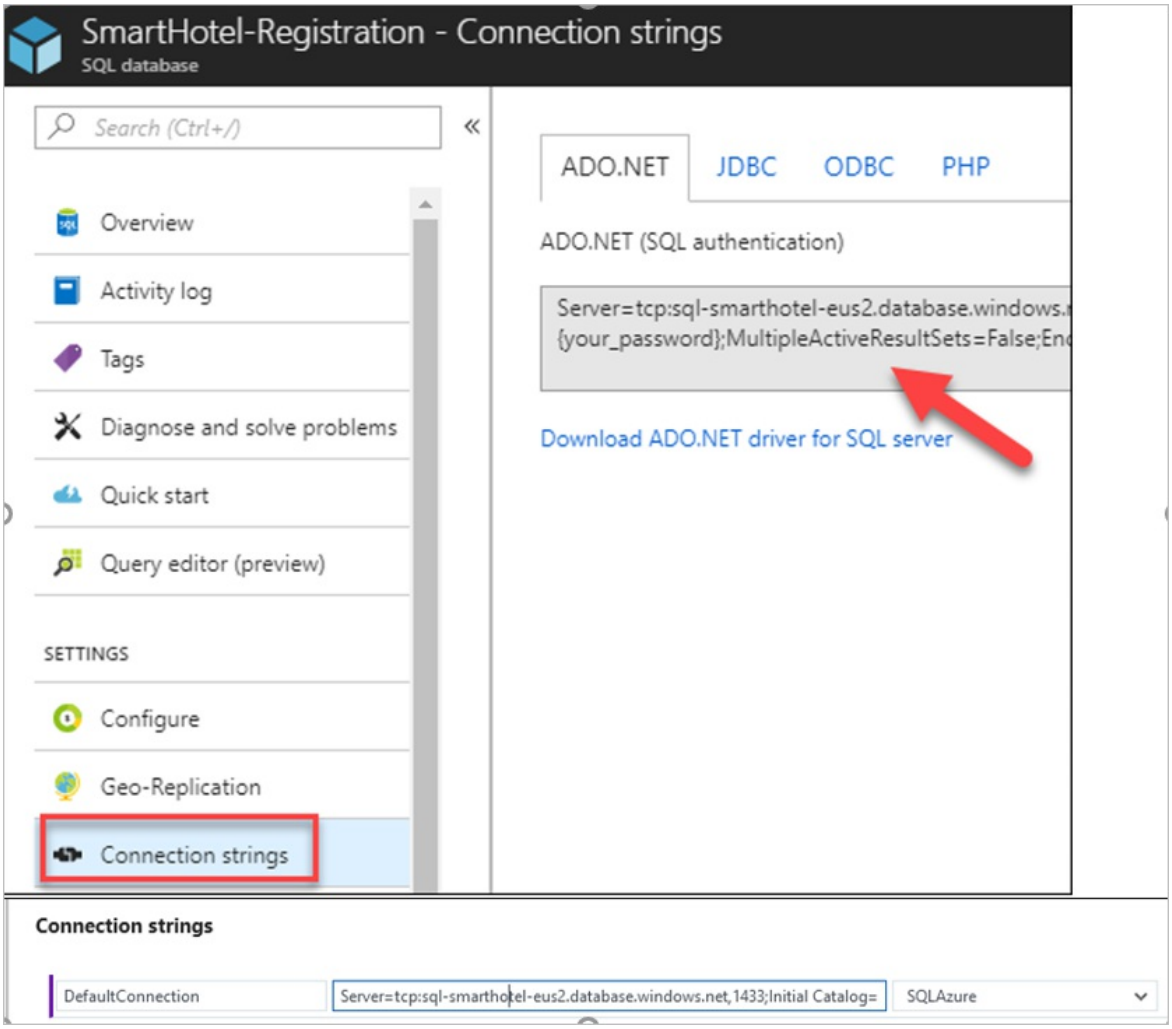
4. They open the solution file for the application. The web app and WCF service have separate projects within the file.



Step 5: Configure connection strings

The Contoso admins make sure that the web apps and database can communicate with each other. To do this, they configure connection strings in the code and in the web apps.

1. In the web app for the WCF service, `SHWCF-EUS2`, under **Settings** > **Application settings**, they add a new connection string named `DefaultConnection`.
2. They pull the connection string from the `SmartHotel-Registration` database and then update it with the correct credentials.



3. In Visual Studio, the admins open the `SmartHotel.Registration.wcf` project from the solution file. In the project, they update the `connectionStrings` section of the `web.config` file with the connection string.

```
<connectionStrings>
  <add name="DefaultConnection" connectionString="Server=tcp:sql-smarthotel-eus2.
</connectionStrings>
```

4. They change the `client` section of the `web.config` file for `SmartHotel.Registration.Web` to point to the new location of the WCF service. This is the URL of the WCF web app that hosts the service endpoint.

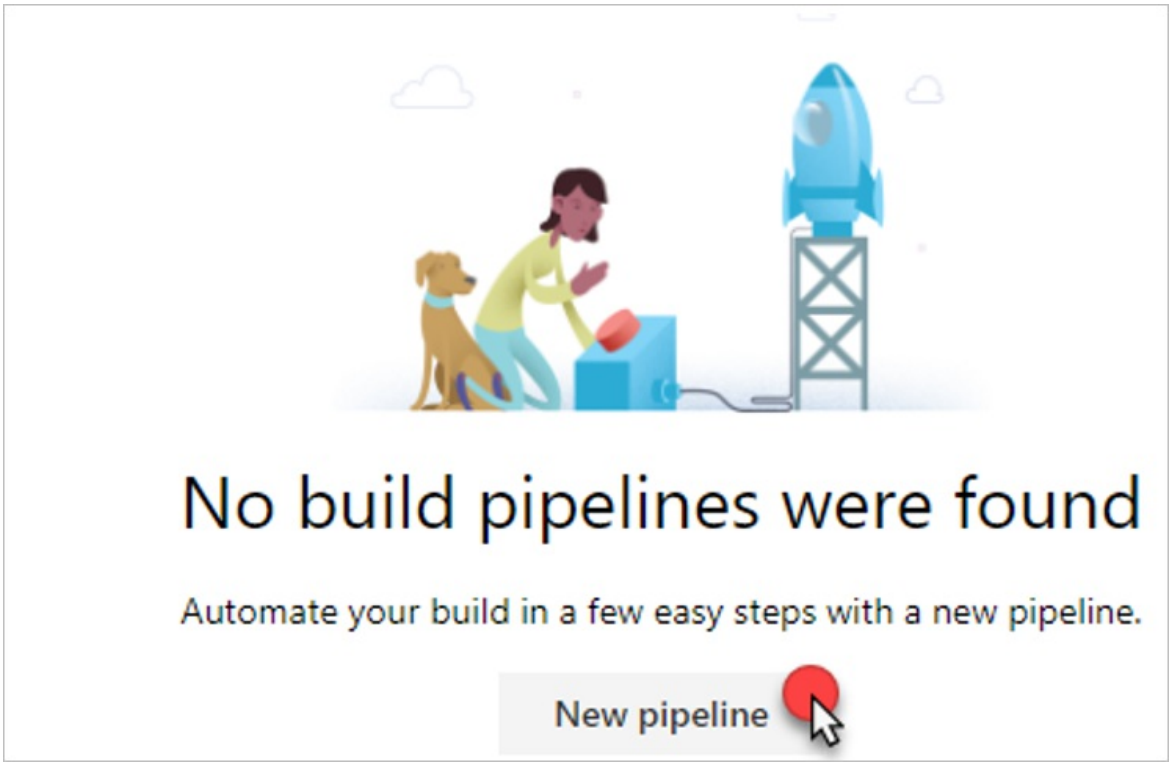
```
<client>
  <endpoint address="http://shwcf-eus2.azurewebsites.net/service.svc" binding="basicHttpBinding"
    bindingConfiguration="BasicHttpBinding_IService" contract="Data.IService"
    name="BasicHttpBinding_IService" />
</client>
```

5. With the code changes now in place, the admins commit and sync them by using Team Explorer in Visual Studio.

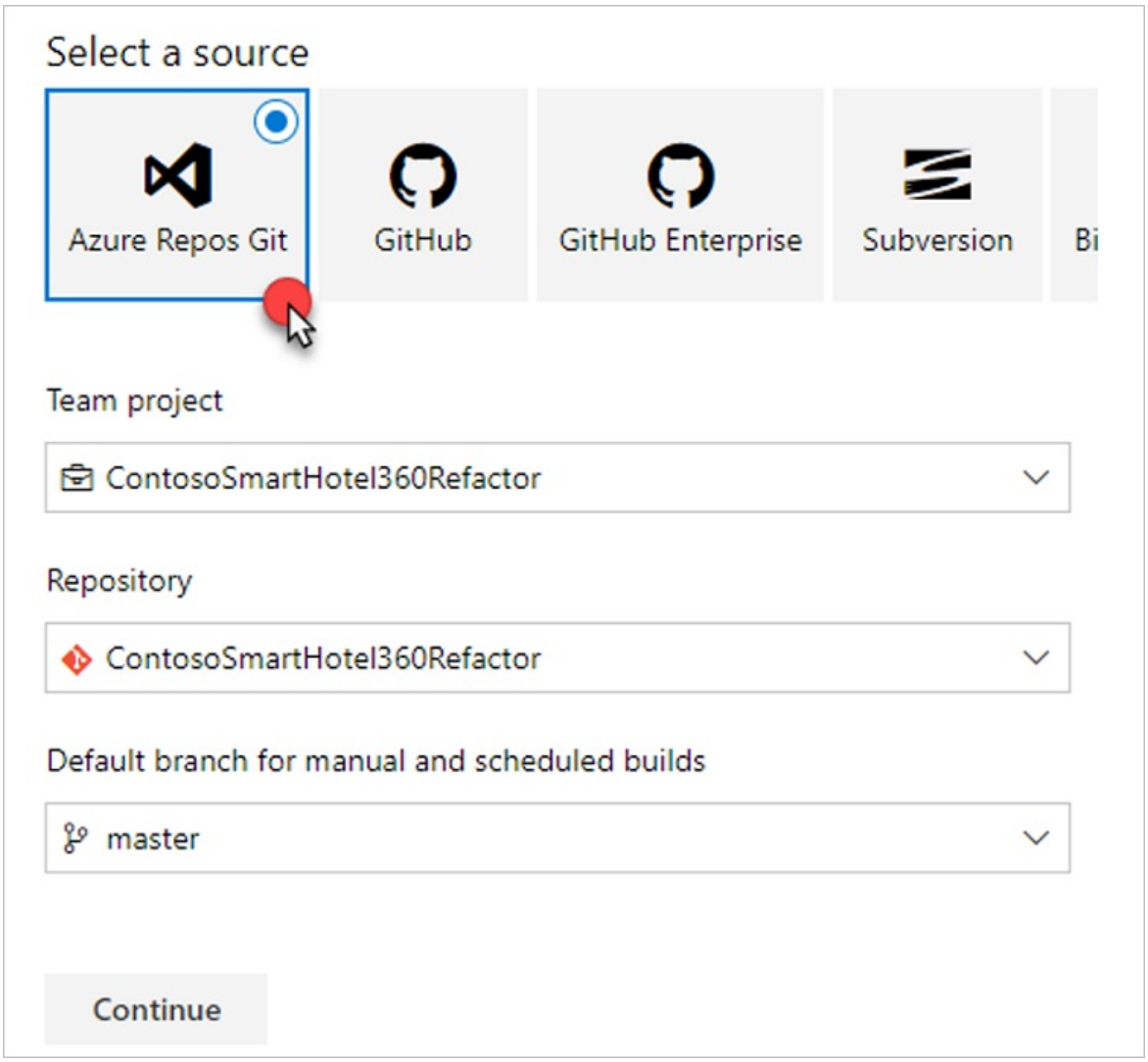
Step 6: Set up build and release pipelines in Azure DevOps

The Contoso admins now configure Azure DevOps to perform the build and release process.

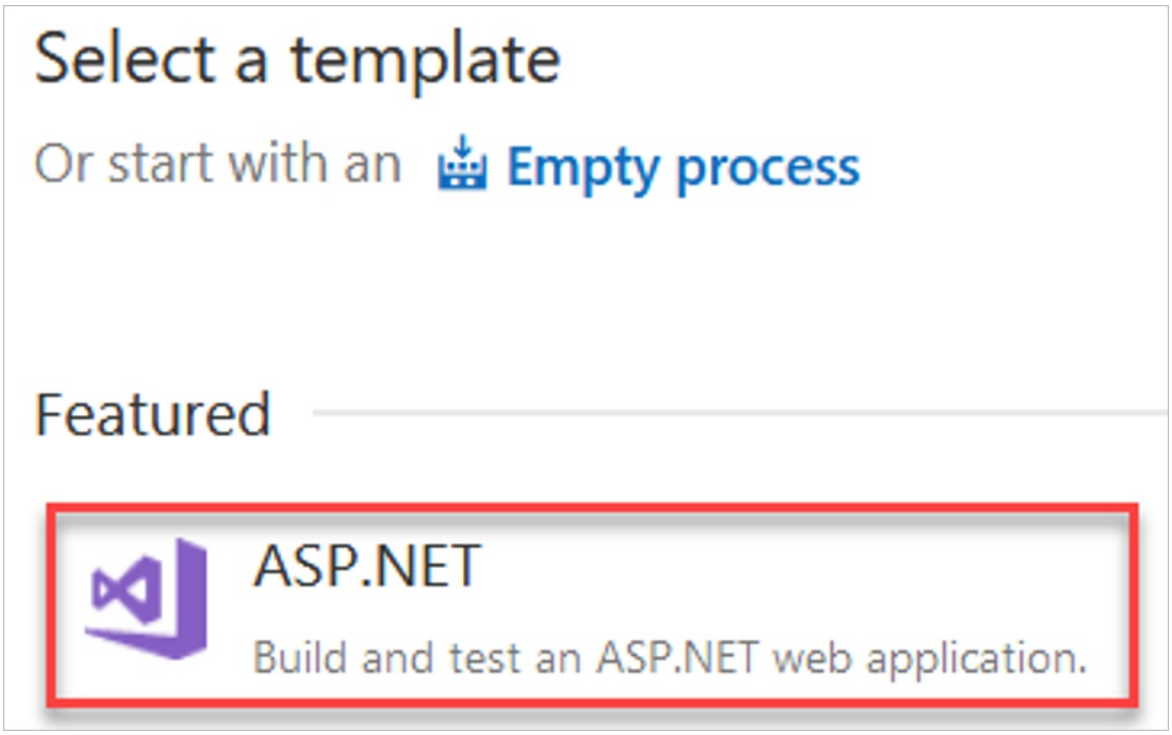
1. In Azure DevOps, they select **Build and release** > **New pipeline**.



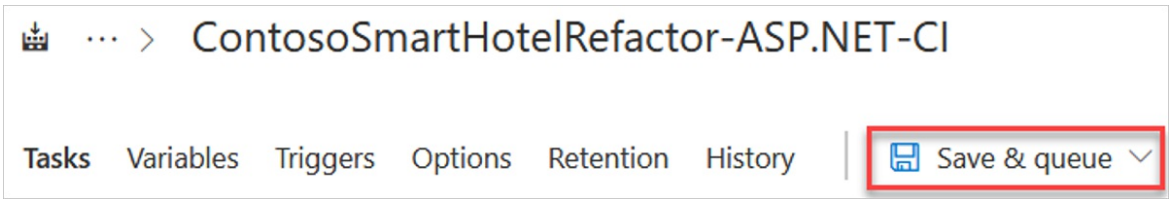
2. They select **Azure Repos Git** and the relevant repo.



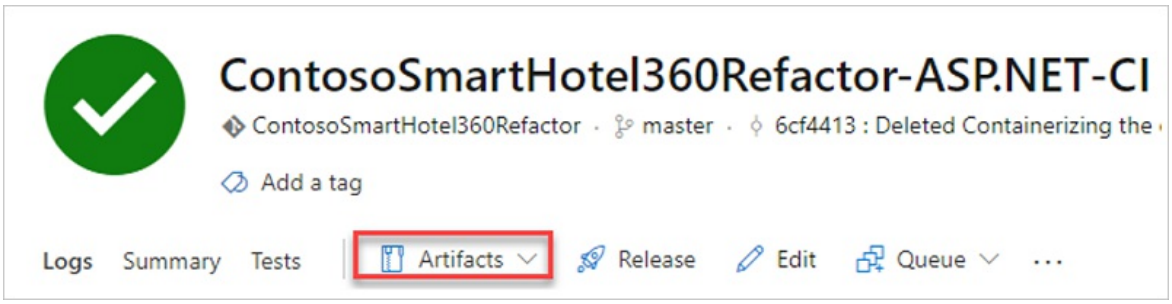
3. In **Select a template**, they select the ASP.NET template for their build.



4. They use the name `ContosoSmartHotelRefactor-ASP.NET-CI` for the build and then select **Save & Queue**, which kicks off the first build.

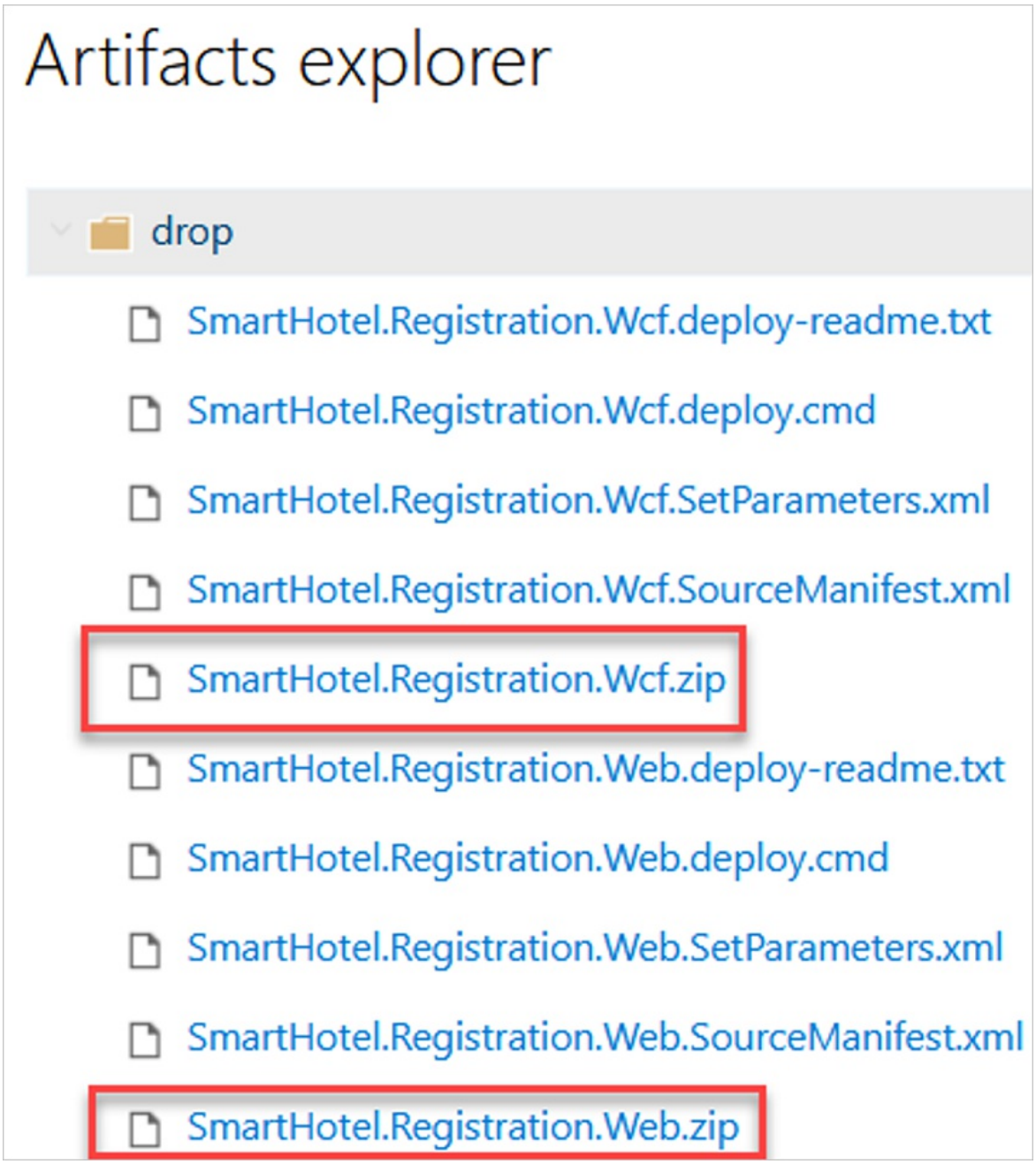


5. They select the build number to watch the process. After it's finished, the admins can see the process feedback, and they select **Artifacts** to review the build results.

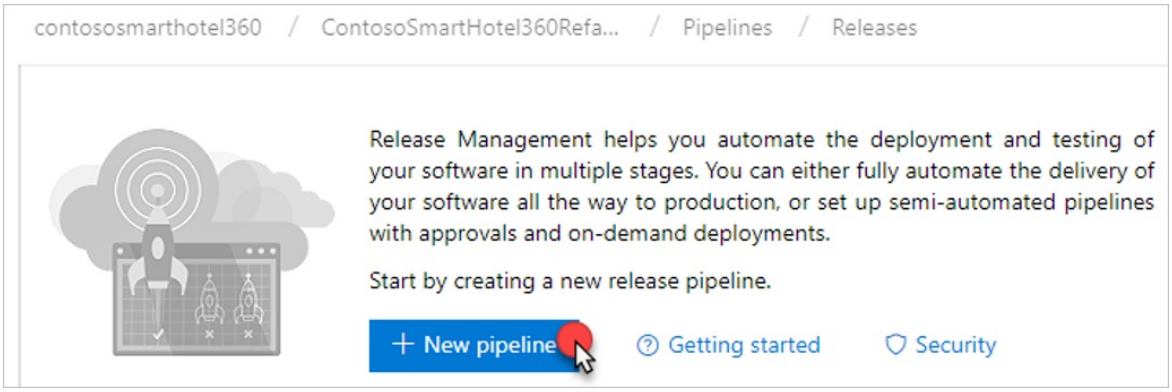


The **Artifacts explorer** pane opens, and the **drop** folder displays the build results.

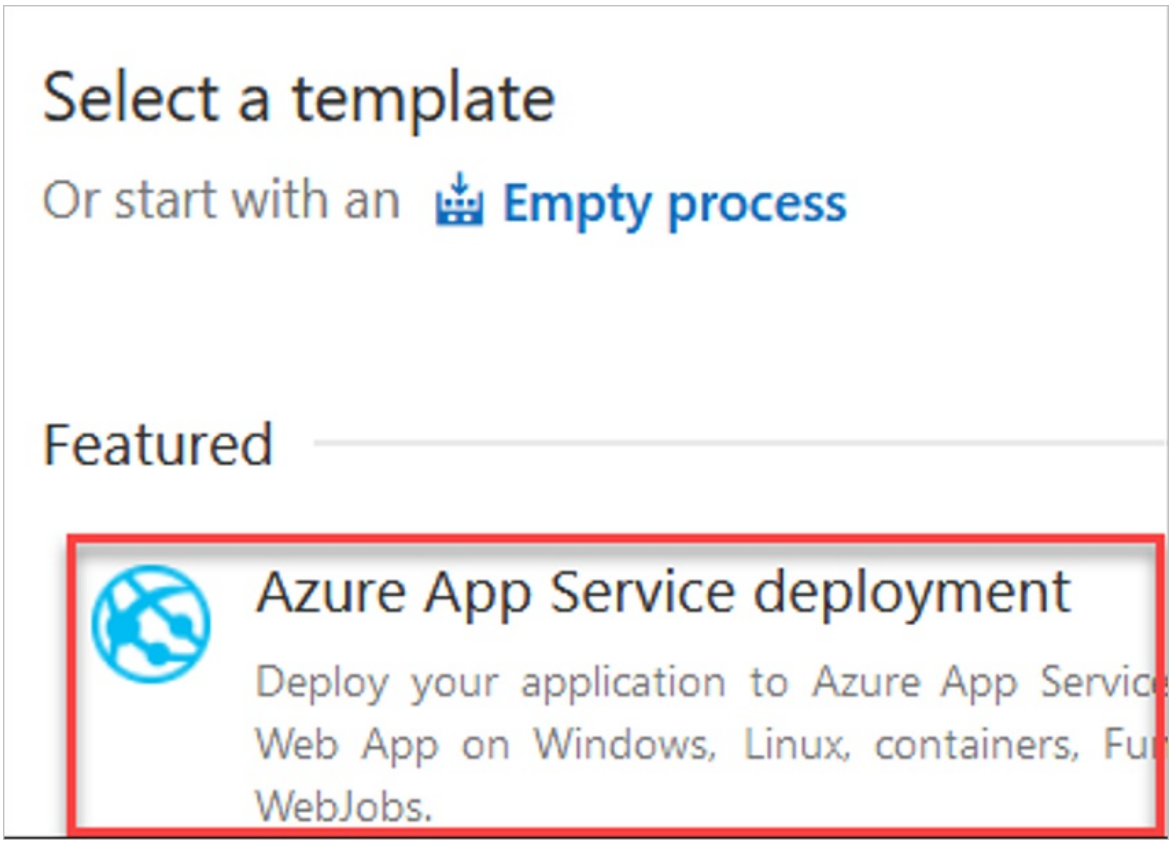
- The two .zip files are the packages that contain the applications.
- These .zip files are used in the release pipeline for deployment to Azure App Service.



6. They select **Releases** > **+ New pipeline**.



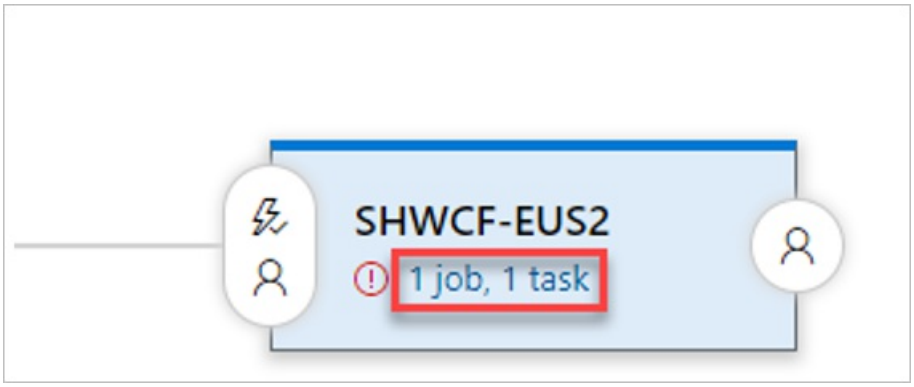
7. They select the deployment template for Azure App Service.



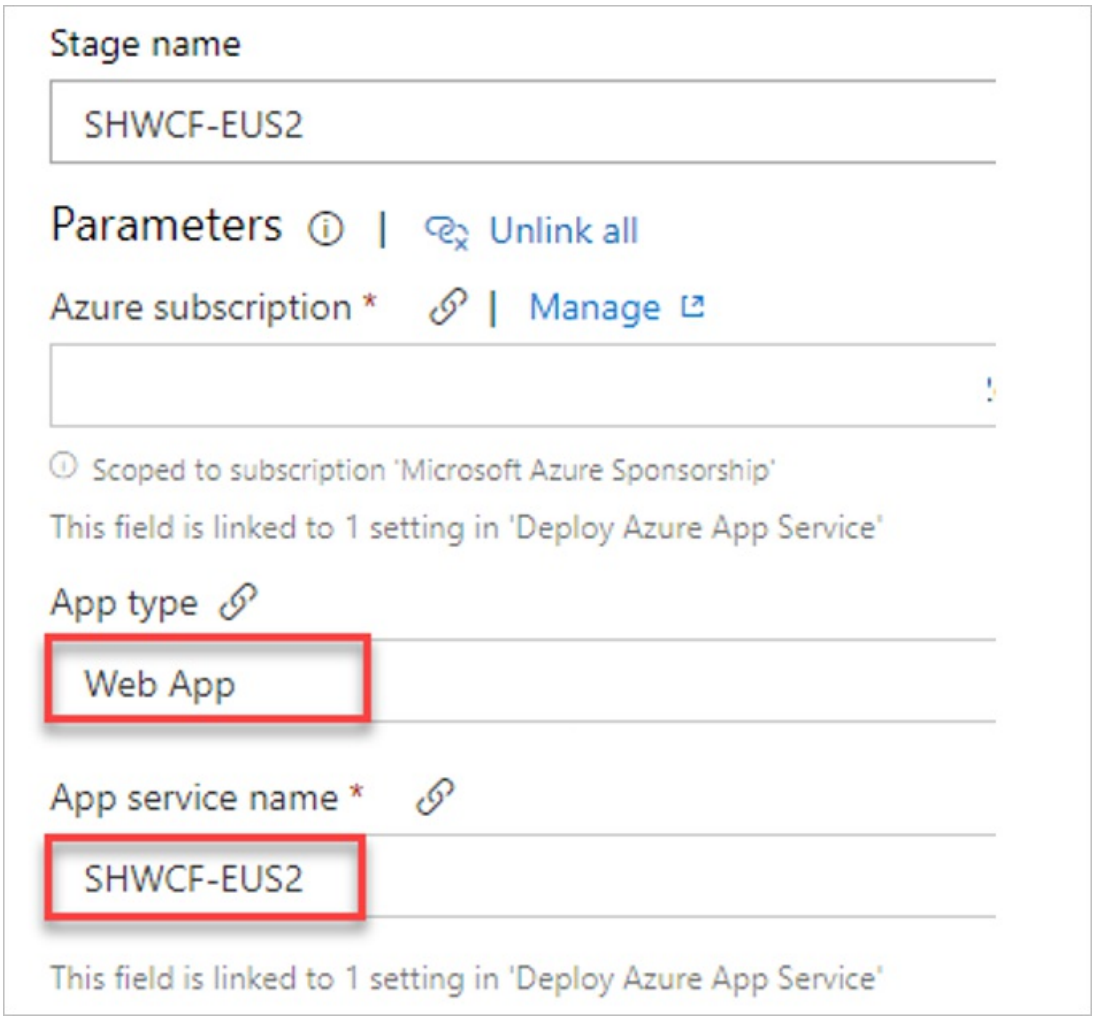
8. They name the release pipeline `ContosoSmartHotel360Refactor` and, in the **Stage name** box, specify `SHWCF-EUS2` as the name of the WCF web app.



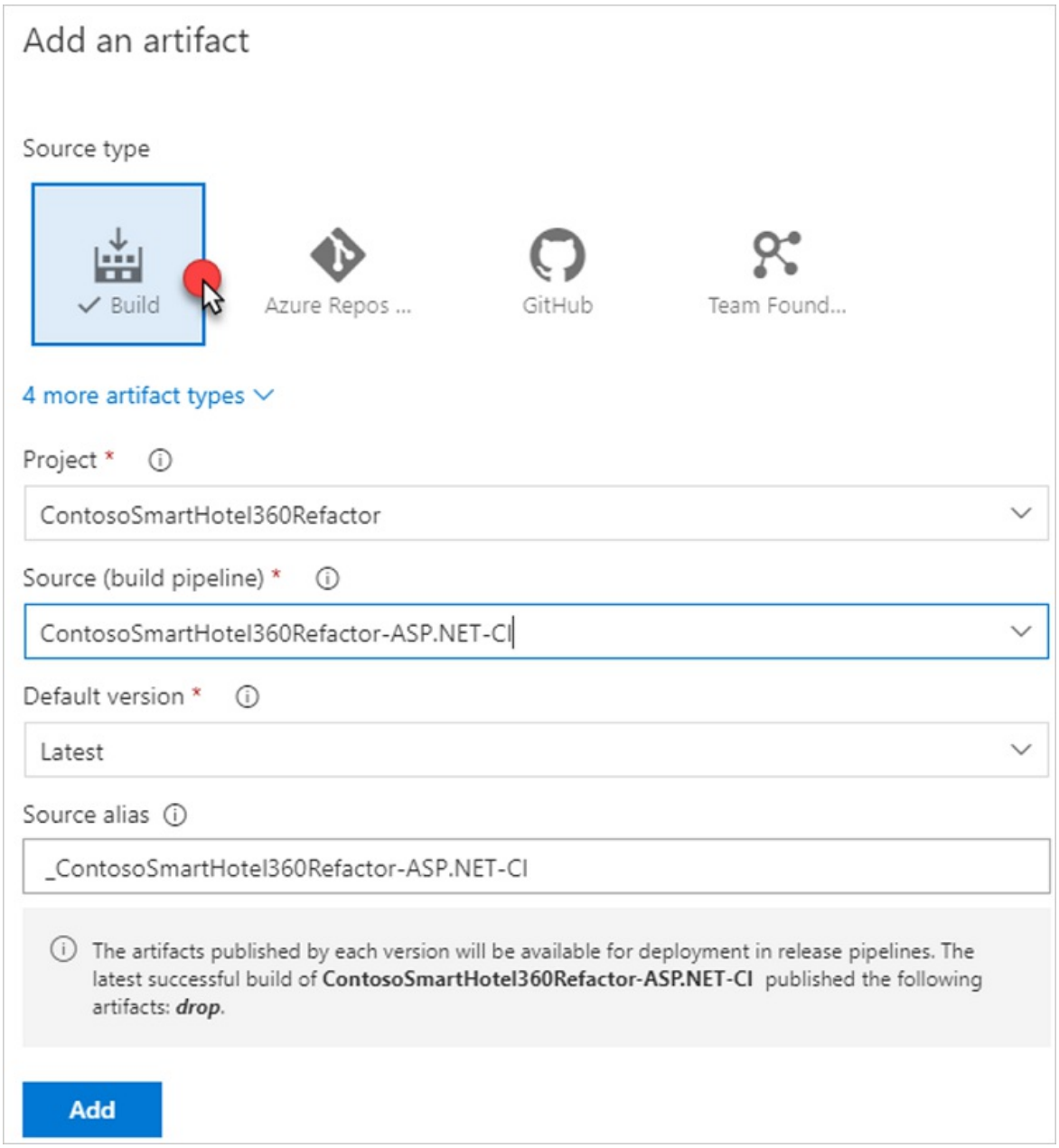
9. Under the stages, they select **1 job, 1 task** to configure deployment of the WCF service.



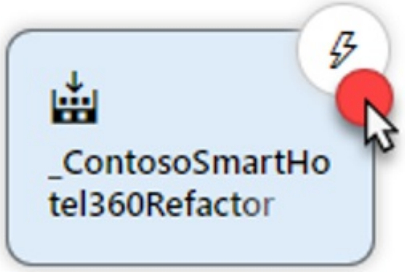
10. They verify that the subscription is selected and authorized, and then they select the **app service name**.



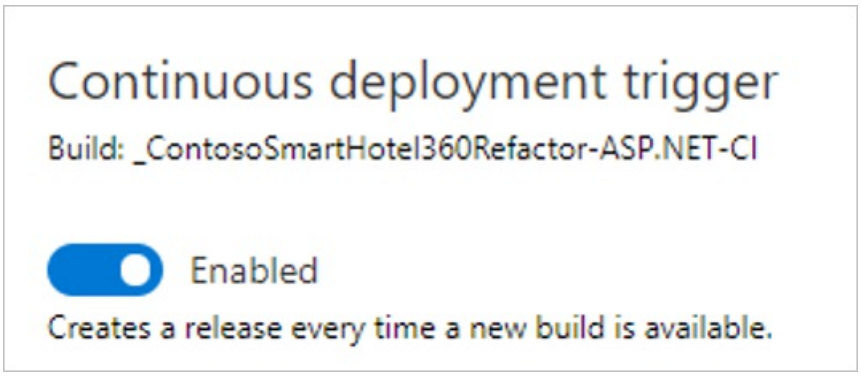
11. On the pipeline, they select **Artifacts**, select **+ Add an artifact**, select **Build** as the source type, and then build with the `ContosoSmarthotel360Refactor` pipeline.



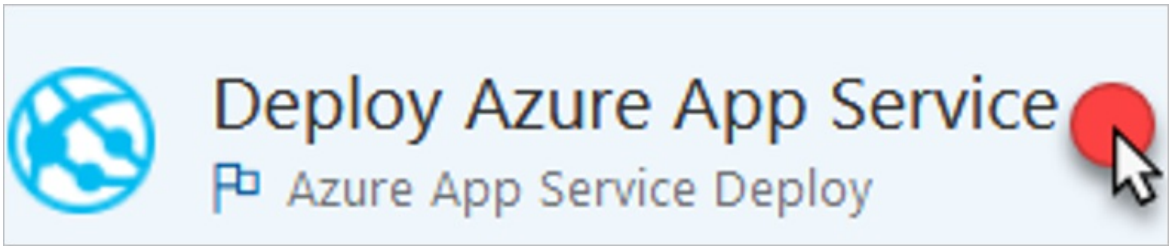
12. To enable the continuous deployment trigger, the admins select the lightning bolt icon on the artifact.



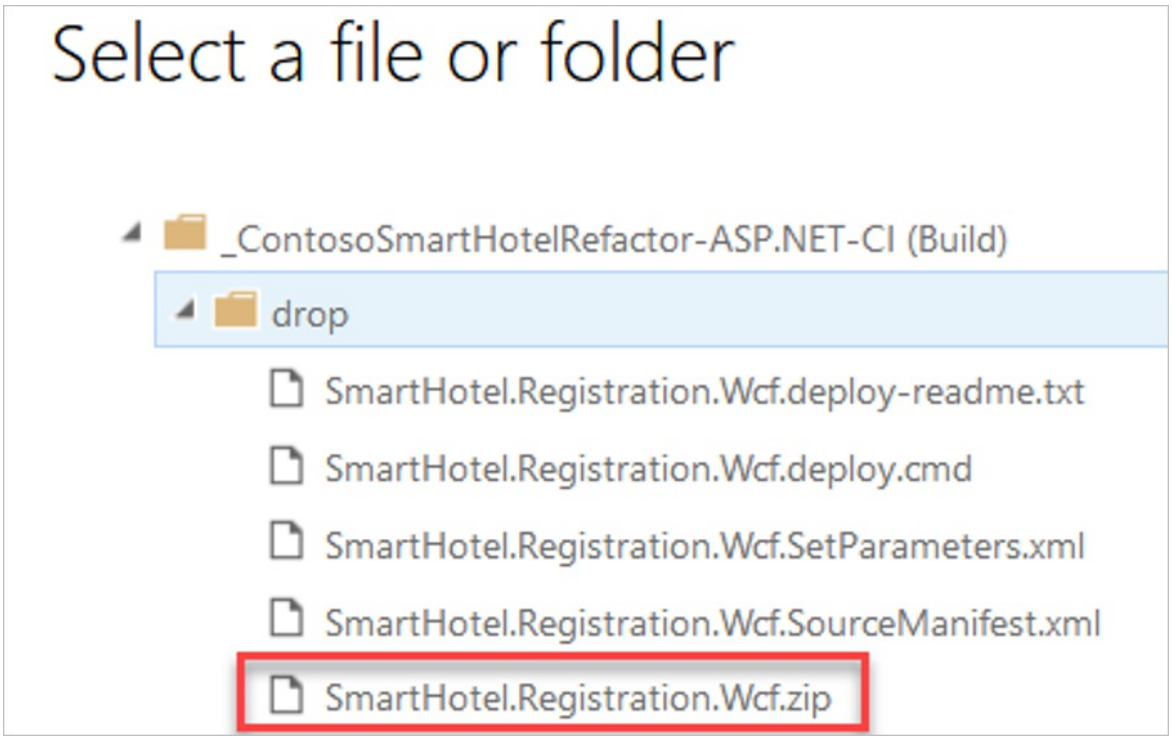
13. They set the continuous deployment trigger to **Enabled**.



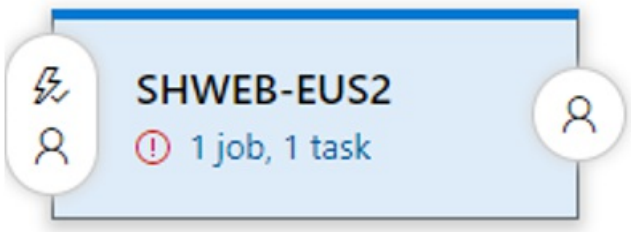
14. The admins go back to the stage **1 job, 1 task** and then select **Deploy Azure App Service**.



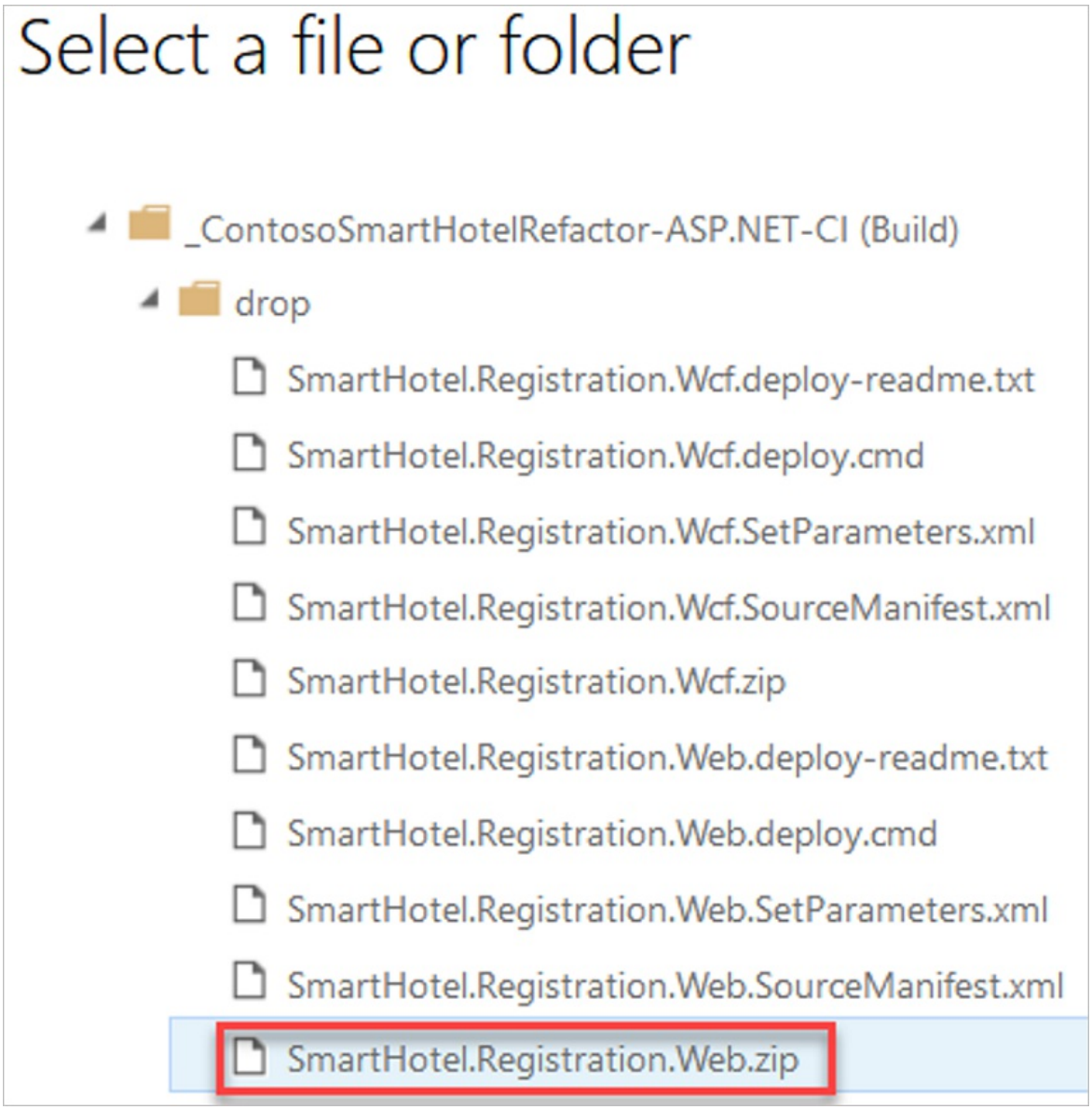
15. In **Select a file or folder**, they expand the **drop** folder, select the `SmartHotel.Registration.Wcf.zip` file that was created during the build, and then select **Save**.



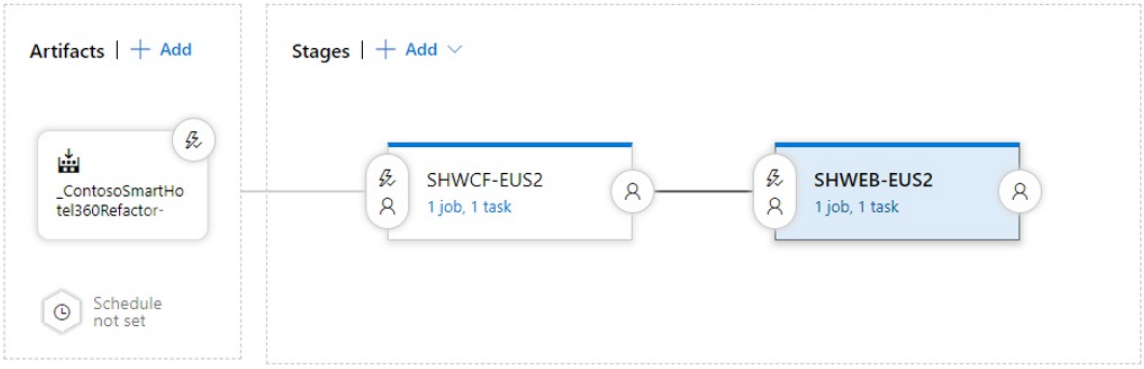
16. They select **Pipeline** > **Stages**, and then select **+ Add** to add an environment for `SHWEB-EUS2`. They select another Azure App Service deployment.



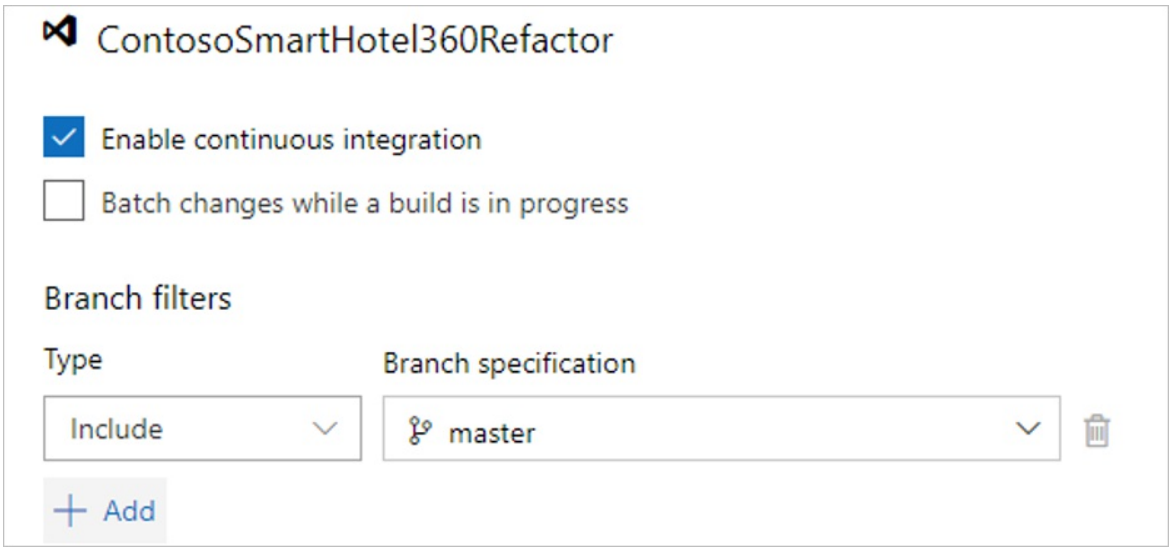
17. They repeat the process to publish the web app `SmartHotel.Registration.Web.zip` file to the correct web app, and then select **Save**.



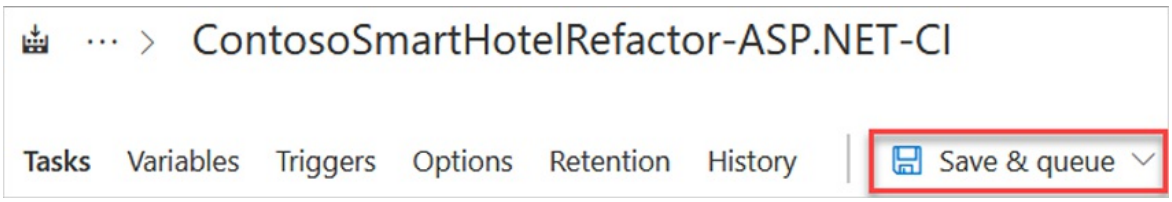
The release pipeline is displayed, as shown here:



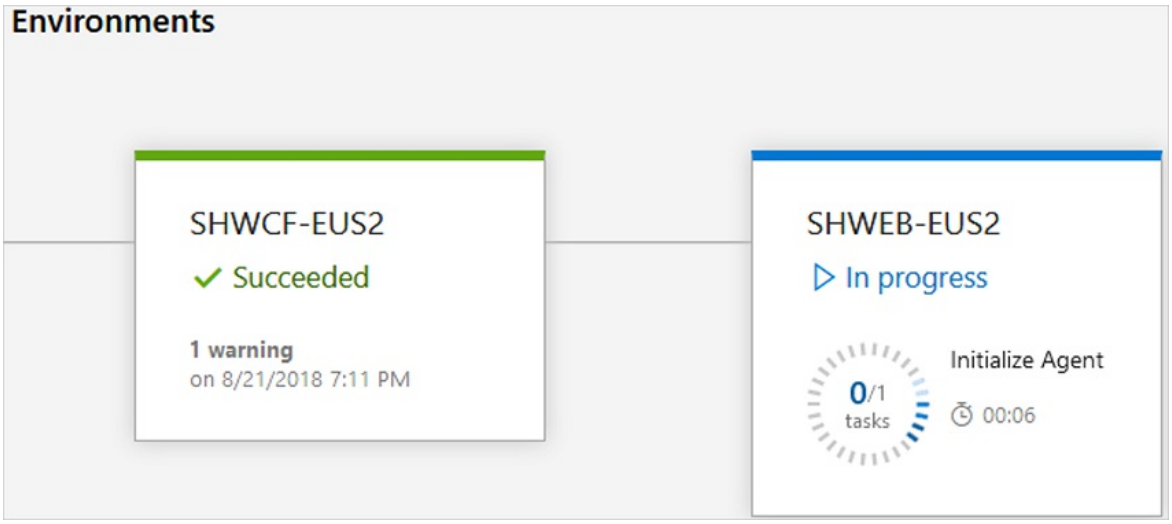
18. They go back to **Build**, select **Triggers**, and then select the **Enable continuous integration** check box. This action enables the pipeline so that when changes are committed to the code, the full build and release occur.



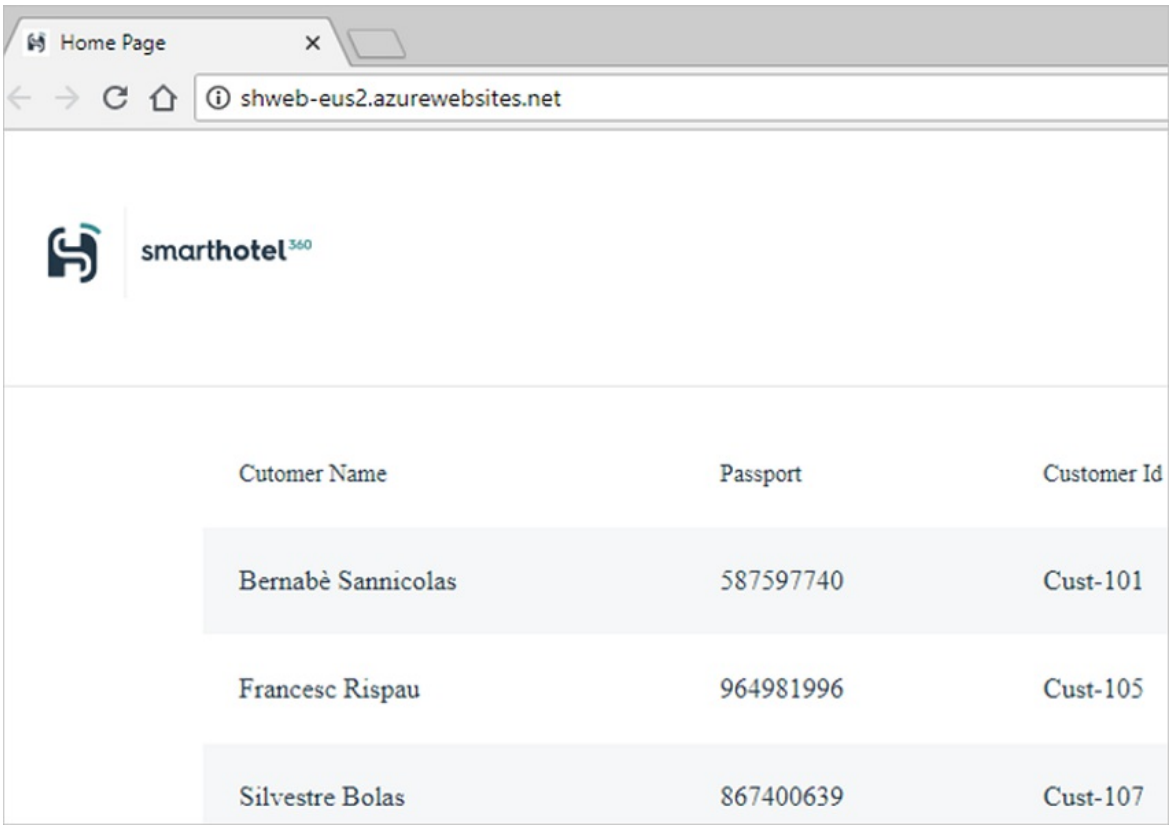
19. They select **Save & Queue** to run the full pipeline. A new build is triggered, which in turn creates the first release of the application to the Azure App Service.



20. Contoso admins can follow the build and release pipeline process from Azure DevOps. After the build finishes, the release starts.



21. After the pipeline finishes, both sites have been deployed and the application is up and running online.



The application has been successfully migrated to Azure.

Clean up after the migration

After the migration, the Contoso team completes the following cleanup steps:

- They remove the on-premises VMs from the vCenter inventory.
- They remove the VMs from the local backup jobs.
- They update their internal documentation to show the new locations for the SmartHotel360 application. The documentation shows the database as running in the SQL managed instance, and the front end as running in two web apps.
- They review any resources that interact with the decommissioned VMs, and they update any relevant settings or documentation to reflect the new configuration.

Review the deployment

With the resources now migrated to Azure, Contoso needs to fully operationalize and help secure their new infrastructure.

Security

- Contoso helps ensure that their new `SmartHotel-Registration` database is secure. Learn more.
- In particular, Contoso updates the web apps to use SSL with certificates.

Backups

- The Contoso team reviews the backup requirements for the database in Azure SQL Managed Instance. Learn more.
- They also learn about managing SQL Database backups and restores. Learn more about automatic backups.
- They consider implementing failover groups to provide regional failover for the database. Learn more.
- They consider deploying the web app in the main region (`East US 2`) and the secondary region (`Central US`) for resilience. The team could configure Traffic Manager to ensure failover during regional outages.

Licensing and cost optimization

- After all resources are deployed, Contoso assigns Azure tags based on their infrastructure planning.
- All licensing is built into the cost of the PaaS services that Contoso is consuming. This cost is deducted from the Enterprise Agreement.
- Contoso will use Azure Cost Management + Billing to ensure that they stay within the budgets established by their IT leadership.

Conclusion

In this article, Contoso refactored the SmartHotel360 application in Azure by migrating the application front-end VM to two Azure App Service web apps. The application database was migrated to an Azure SQL managed instance.

Refactor a Linux application by using Azure App Service, Traffic Manager, and Azure Database for MySQL

5/4/2021 • 14 minutes to read • Edit Online

This article shows how the fictional company Contoso refactors a two-tier LAMP-based application, migrating it from on-premises to Azure by using Azure App Service with GitHub integration and Azure Database for MySQL.

osTicket, the service desk application that we use in this example, is provided as open-source software. If you want to use it for your own testing purposes, you can download it from the osTicket repo in GitHub.

Business drivers

The IT leadership team has worked closely with business partners to understand what they want to achieve:

- **Address business growth.** Contoso is growing and moving into new markets. It needs additional customer service agents.
- **Scale.** The solution should be built so that Contoso can add more customer service agents as the business scales.
- **Improve resiliency.** In the past, issues with the system affected internal users only. With the new business model, external users will be affected, and Contoso needs the application up and running at all times.

Migration goals

To determine the best migration method, the Contoso cloud team has pinned down their goals for this migration:

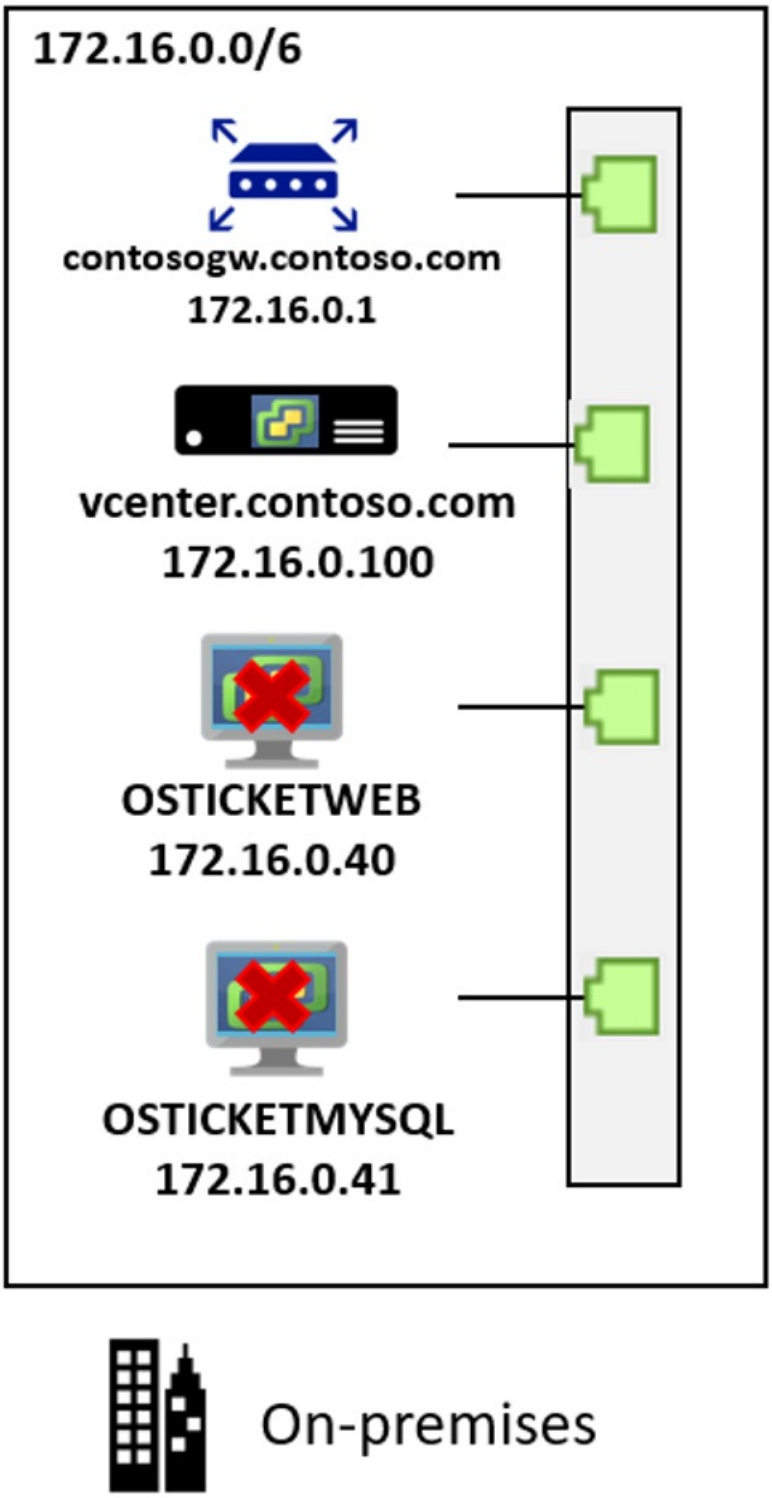
- The application should scale beyond current on-premises capacity and performance. Contoso is moving the application to take advantage of Azure's on-demand scaling.
- Contoso wants to move the application code base to a continuous delivery pipeline. As application changes are pushed to GitHub, Contoso wants to deploy those changes without tasks for operations staff.
- The application must be resilient, with capabilities for growth and failover. Contoso wants to deploy the application in two different Azure regions and set it up to scale automatically.
- Contoso wants to minimize database admin tasks after the application is moved to the cloud.

Solution design

After pinning down their goals and requirements, Contoso designs and reviews a deployment solution, and identifies the migration process, including the Azure services that will be used for the migration.

Current architecture

- The application is tiered across two virtual machines (VMs) (`OSTICKETWEB` and `OSTICKETMYSQL`).
- The VMs are located on VMware ESXi host `contosohost1.contoso.com` (version 6.5).
- The VMware environment is managed by vCenter Server 6.5 (`vcenter.contoso.com`), running on a VM.
- Contoso has an on-premises datacenter (`contoso-datacenter`), with an on-premises domain controller (`contosodc1`).

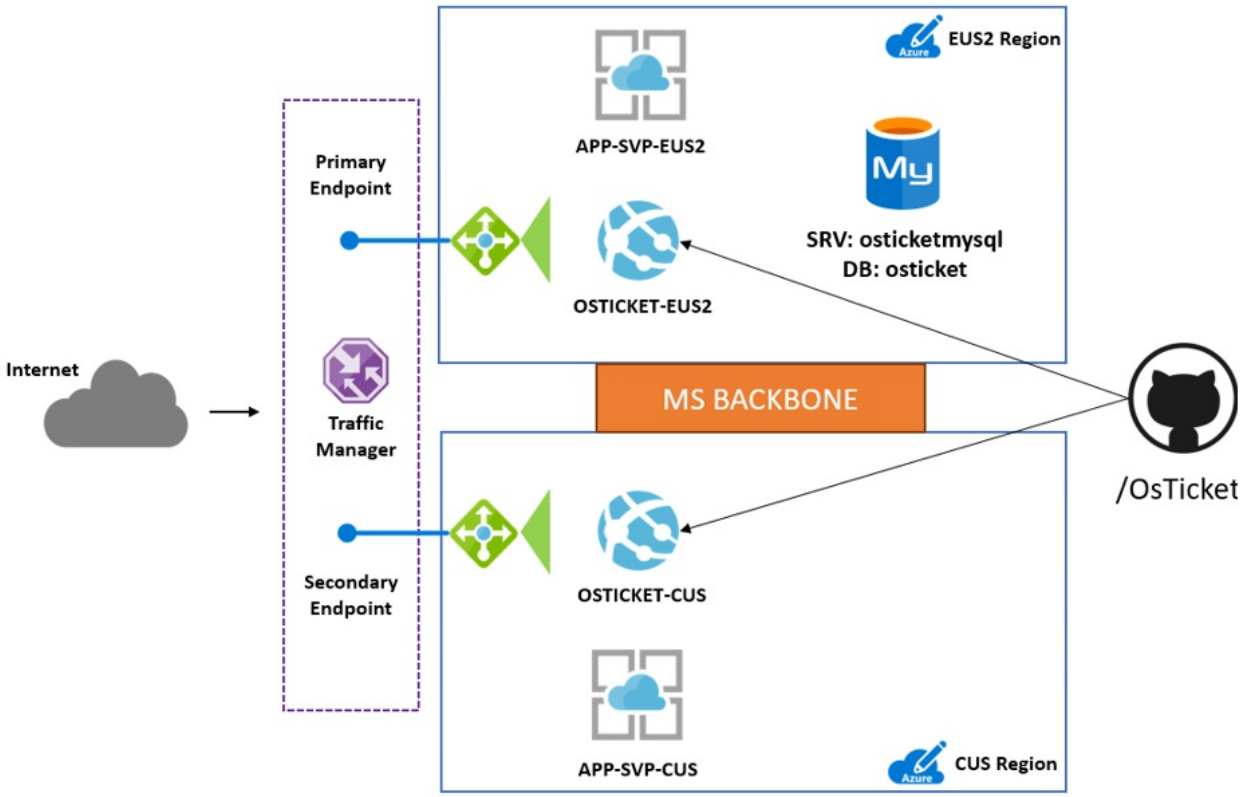


Proposed architecture

Here's the proposed architecture:

- The web tier application on `OSTICKETWEB` will be migrated by building an Azure App Service web app in two Azure regions. The Contoso team will implement Azure App Service for Linux by using the PHP 7.0 Docker container.
- The application code will be moved to GitHub, and the Azure App Service web app will be configured for continuous delivery with GitHub.
- Azure App Service will be deployed in both the primary region (`East US 2`) and secondary region (`Central US`).
- Azure Traffic Manager will be set up in front of the two web apps in both regions.
- Traffic Manager will be configured in priority mode to force the traffic through `East US 2`.

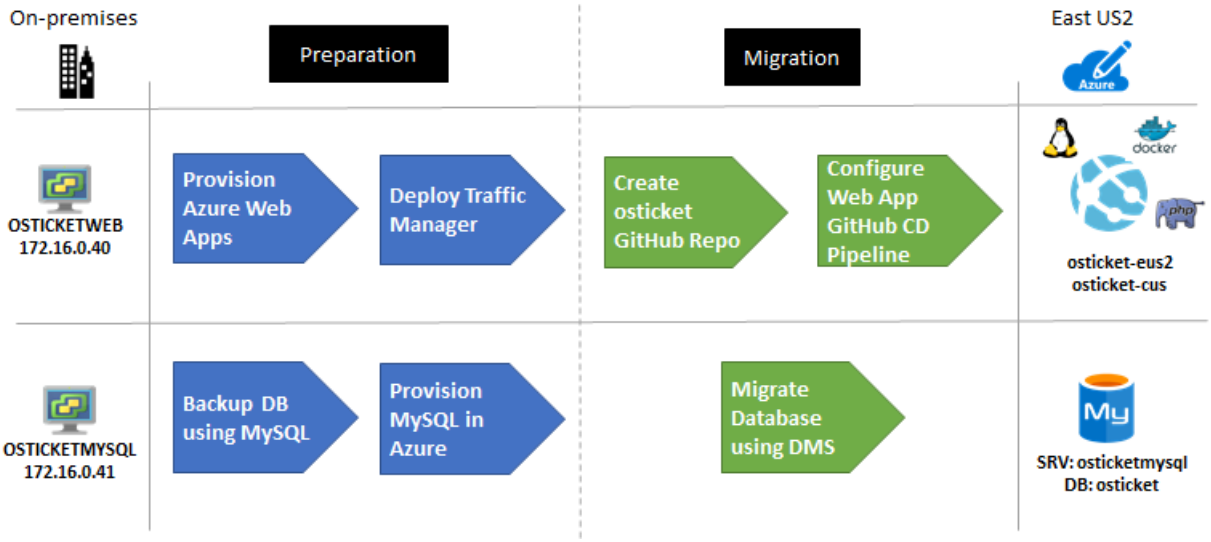
- If the Azure app server in `East US 2` goes offline, users can access the failed over application in `Central US`.
- The application database will be migrated to the Azure Database for MySQL service by using Azure Database Migration Service. The on-premises database will be backed up locally, and restored directly to Azure Database for MySQL.
- The database will reside in the primary region (`East US 2`) in the database subnet (`PROD-DB-EUS2`) of the production network (`VNET-PROD-EUS2`).
- Since they're migrating a production workload, Azure resources for the application will reside in the production resource group `ContosoRG`.
- The Traffic Manager resource will be deployed in Contoso's infrastructure resource group `ContosoInfraRG`.
- The on-premises VMs in the Contoso datacenter will be decommissioned after the migration is done.



Migration process

Contoso completes the migration process as follows:

1. As a first step, Contoso admins set up the Azure infrastructure, including provisioning Azure App Service, setting up Traffic Manager, and provisioning an Azure Database for MySQL instance.
2. After preparing the Azure infrastructure, they migrate the database by using Azure Database Migration Service.
3. After the database is running in Azure, they upload a GitHub private repository for Azure App Service with continuous delivery, and load it with the osTicket application.
4. In the Azure portal, they load the application from GitHub to the Docker container by running Azure App Service.
5. They tweak DNS settings and configure autoscaling for the application.



Azure services

SERVICE	DESCRIPTION	COST	
Azure App Service	The service runs and scales applications by using Azure platform as a service (PaaS) for websites.	Pricing is based on the size of the instances and the features required. Learn more.	
Azure Traffic Manager	A load balancer that uses Domain Name System (DNS) to direct users to Azure or to external websites and services.	Pricing is based on the number of received DNS queries and the number of monitored endpoints.	Learn more.
Azure Database Migration Service	Azure Database Migration Service enables seamless migration from multiple database sources to Azure data platforms, with minimal downtime.	Learn about supported regions and Database Migration Service pricing.	
Azure Database for MySQL	The database is based on the open-source MySQL database engine. It provides a fully managed, enterprise-ready community MySQL database for application development and deployment.	Pricing is based on compute, storage, and backup requirements. Learn more.	

Prerequisites

To run this scenario, Contoso must meet the following prerequisites:

REQUIREMENTS	DETAILS

REQUIREMENTS	DETAILS
Azure subscription	Contoso created subscriptions earlier in this article series. If you don't have an Azure subscription, create a free account. If you create a free account, you're the administrator of your subscription and can perform all actions. If you use an existing subscription and you're not the administrator, you need to work with the admin to assign you Owner or Contributor permissions.
Azure infrastructure	Contoso set up their Azure infrastructure as described in Azure infrastructure for migration.

Scenario steps

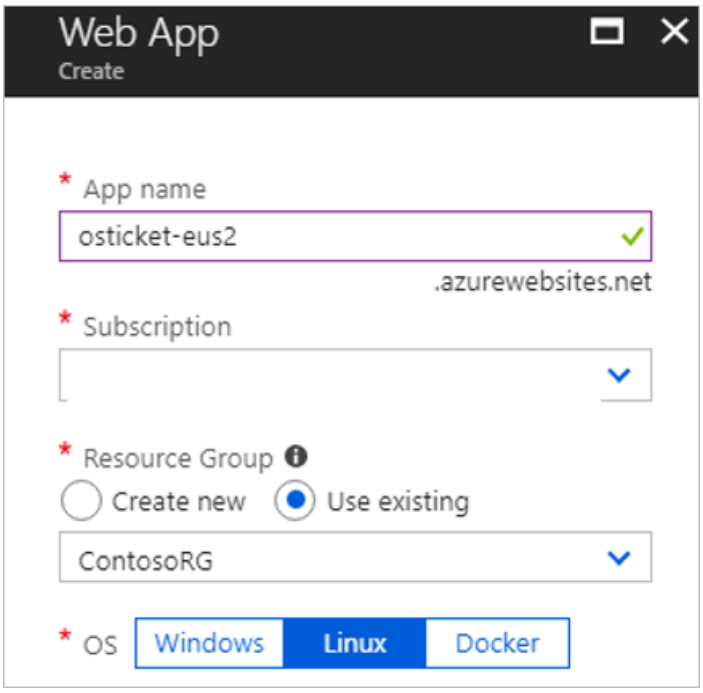
Here's the Contoso plan for completing the migration:

- **Step 1: Provision Azure App Service**. Contoso admins will provision web apps in the primary and secondary regions.
- **Step 2: Set up Traffic Manager**. They set up Traffic Manager in front of the web apps, for routing and load balancing traffic.
- **Step 3: Provision Azure Database for MySQL**. In Azure, they provision an instance of Azure Database for MySQL.
- **Step 4: Migrate the database**. They migrate the database by using Azure Database Migration Service.
- **Step 5: Set up GitHub**. They set up a local GitHub repository for the application web sites and code.
- **Step 6: Configure the web apps**. They configure the web apps with the osTicket websites.

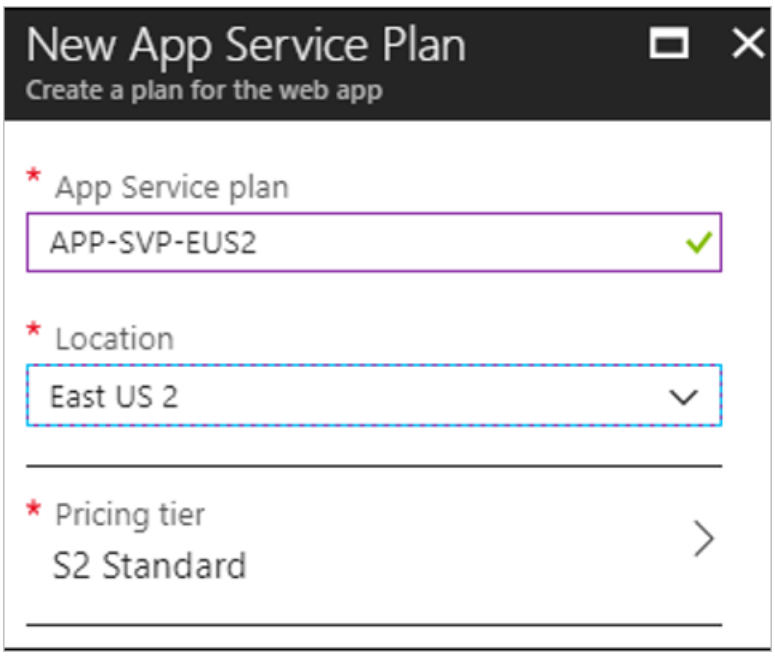
Step 1: Provision Azure App Service

Contoso admins provision two web apps (one in each region) by using Azure App Service.

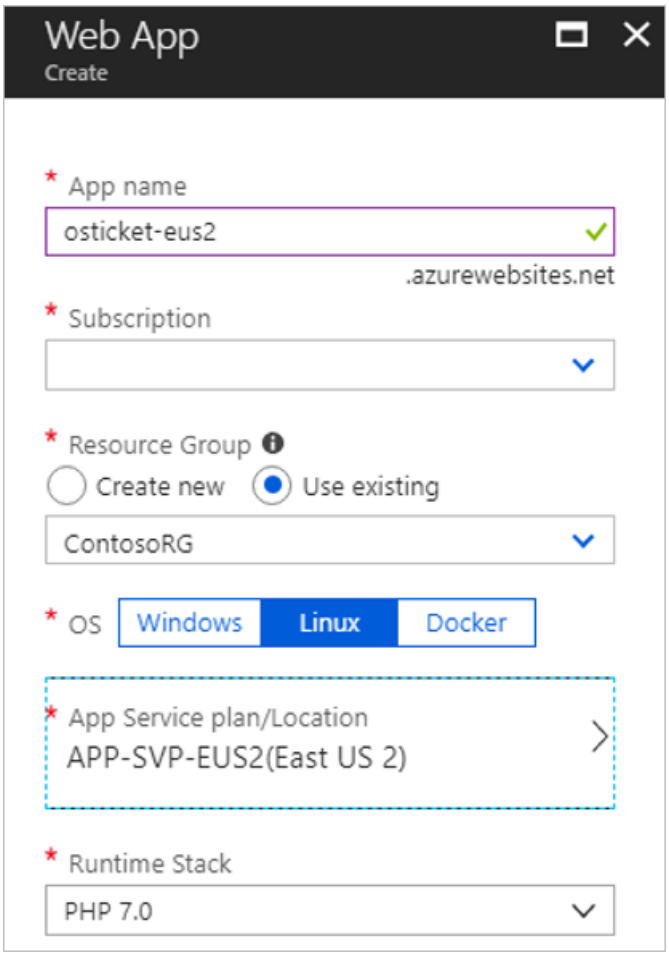
1. They create a web app resource (`osticket-eus2`) in the primary region (`East US 2`) via Azure Marketplace.
2. They put the resource in the production resource group `ContosoRG` .



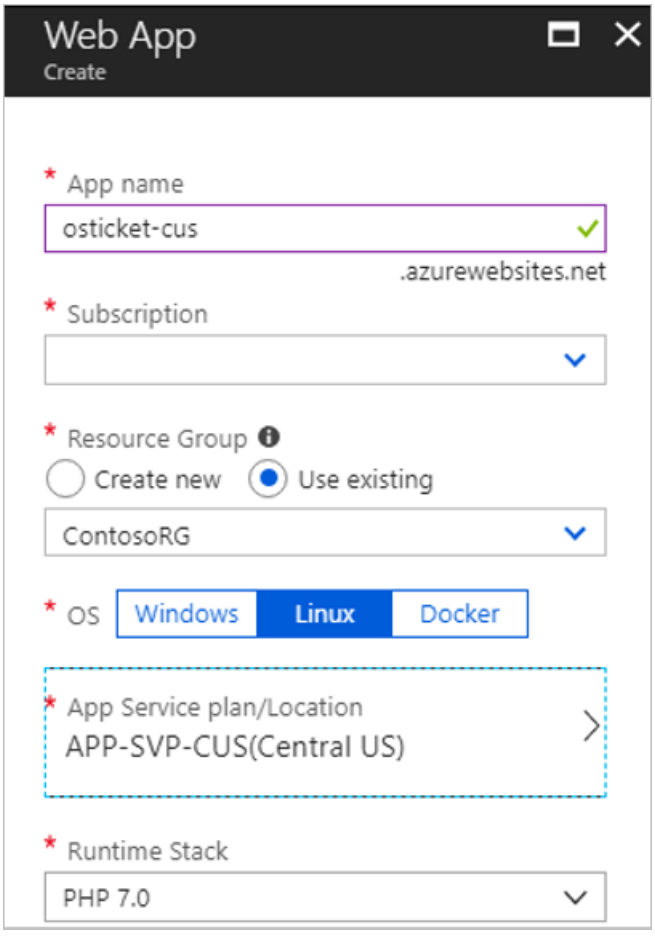
3. They create an App Service plan, `APP-SVP-EUS2`, in the primary region, and they use the standard size.



4. They select a Linux OS with PHP 7.0 runtime stack, which is a Docker container.



5. They create a second web app, `osticket-cus`, and an Azure App Service plan for **Central US**.



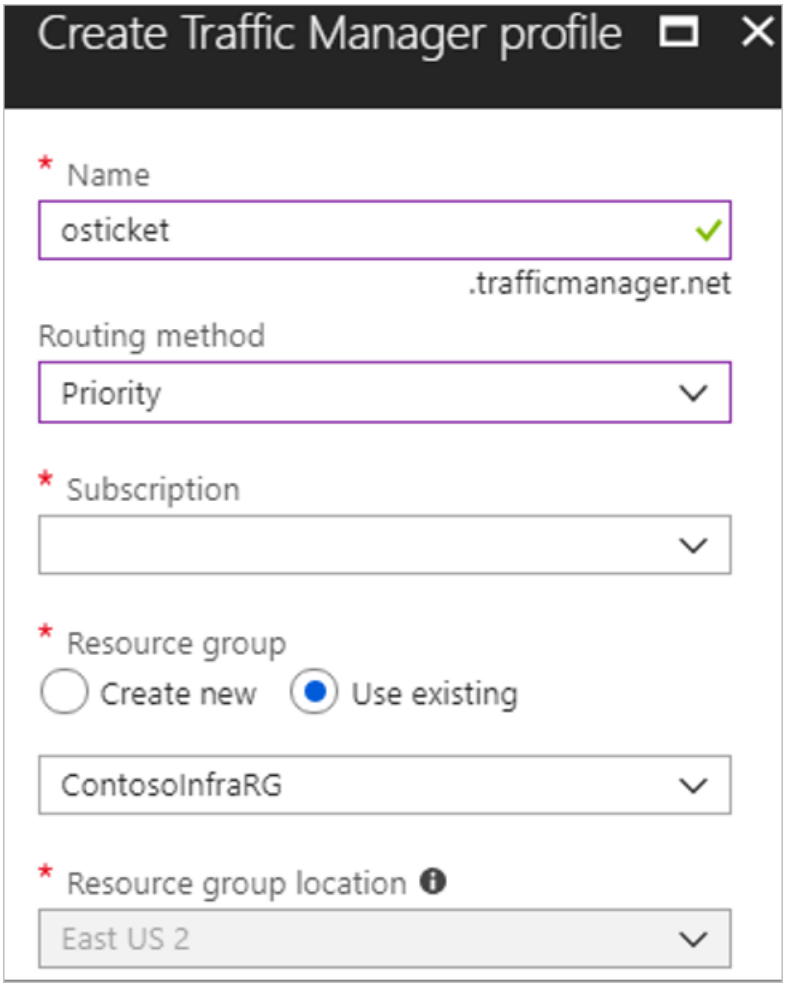
Need more help?

- Learn about Azure App Service web apps.
- Learn about Azure App Service on Linux.

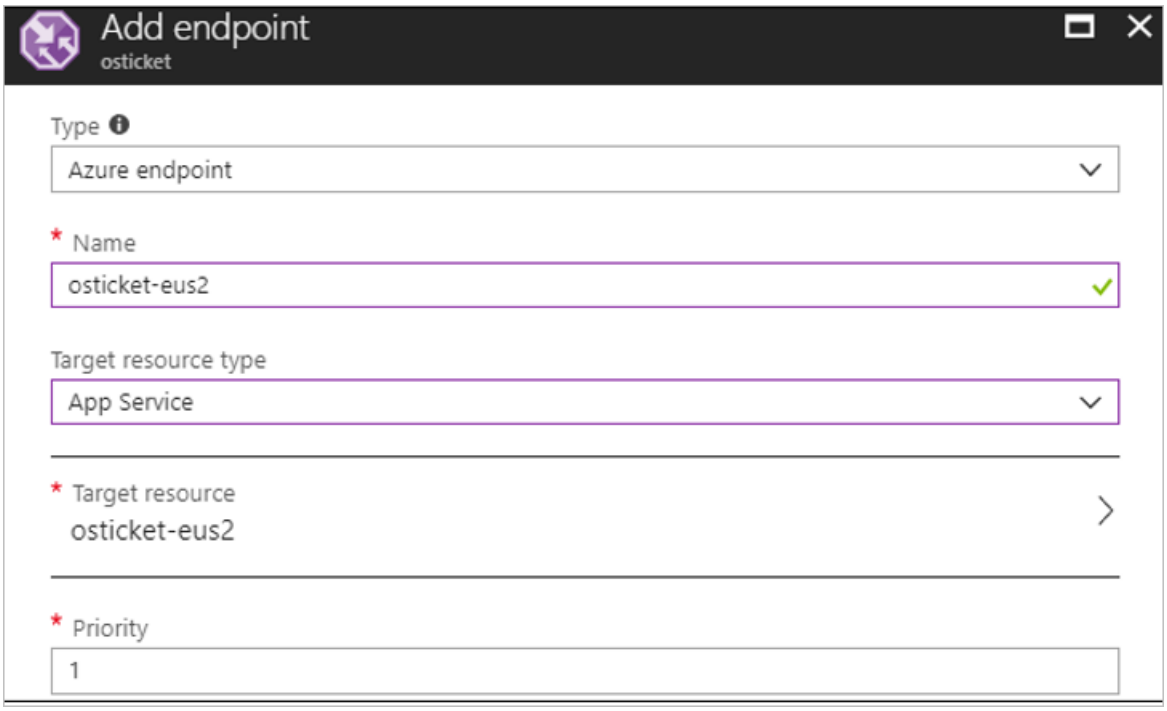
Step 2: Set up Traffic Manager

Contoso admins set up Traffic Manager to direct inbound web requests to the web apps that are running on the osTicket web tier.

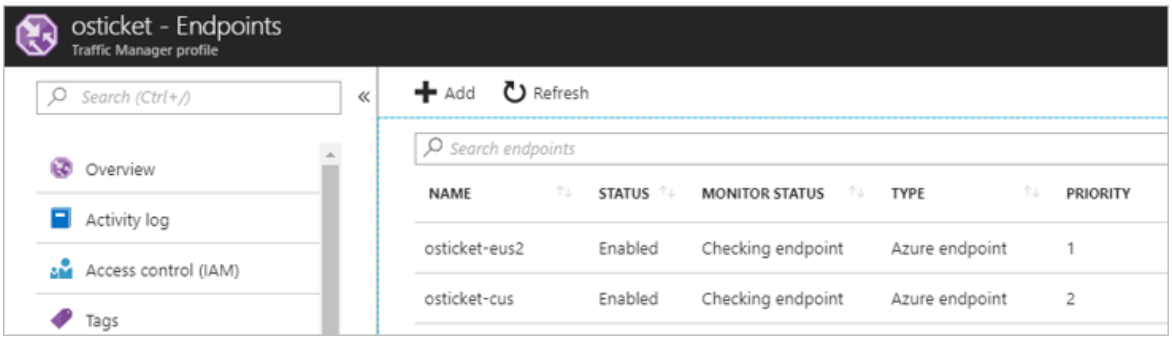
1. In Azure Marketplace, they create a Traffic Manager resource, `osticket.trafficmanager.net`. They use priority routing so that **East US 2** is the primary site. They place the resource in their existing infrastructure resource group, `ContosoInfraRG`. Note that Traffic Manager is global and not bound to a specific location.



2. They configure Traffic Manager with endpoints. They add the web app in East US 2 as the primary site, `osticket-eus2`, and the web app in Central US as the secondary site, `osticket-cus`.



3. After they add the endpoints, the admins can monitor them.



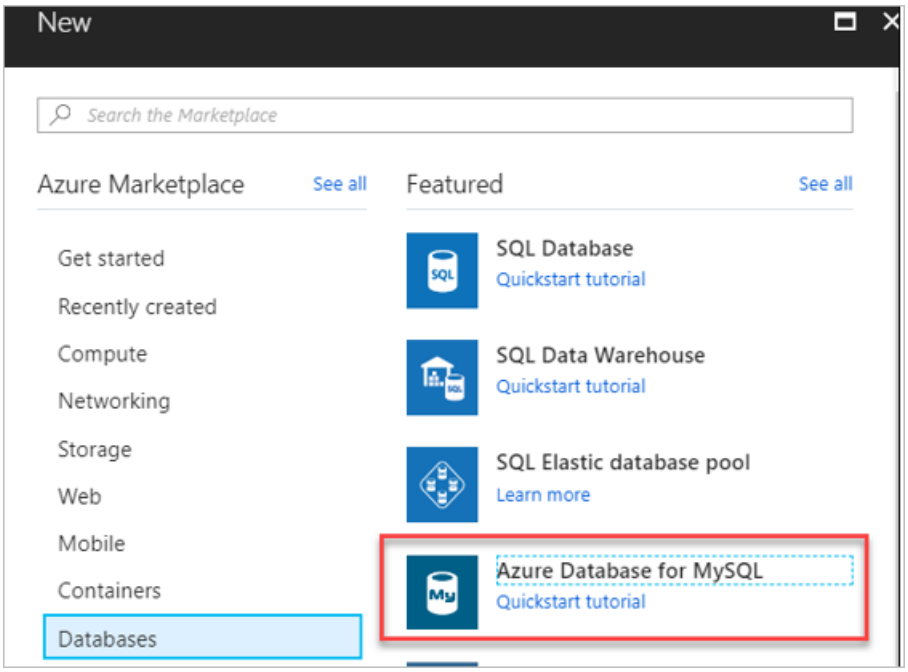
Need more help?

- Learn about Traffic Manager.
- Learn about routing traffic to a priority endpoint.

Step 3: Provision Azure Database for MySQL

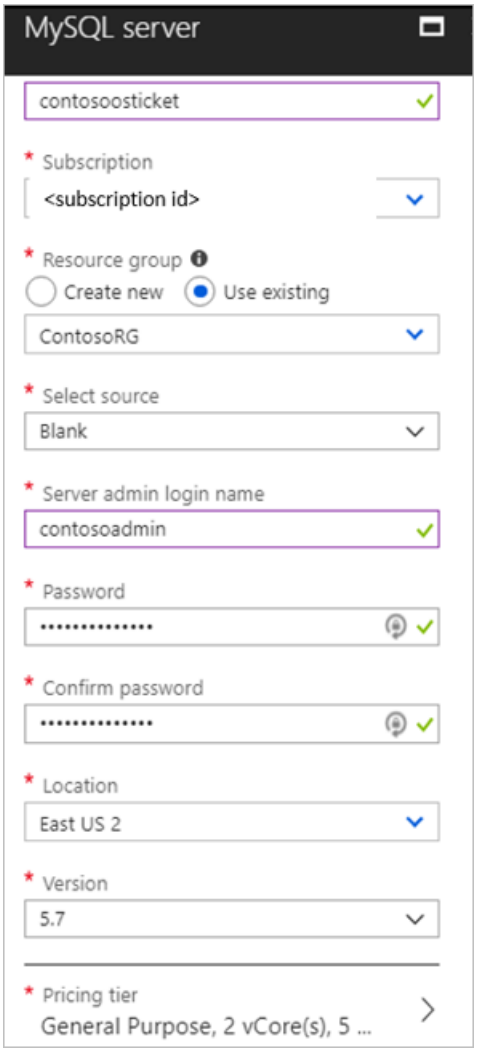
Contoso admins provision a MySQL database instance in the primary region, East US 2.

1. In the Azure portal, they create an Azure Database for MySQL resource.

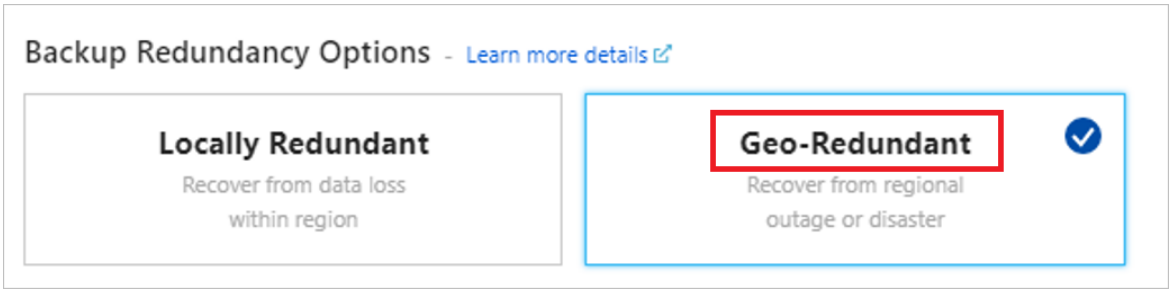


2. They add the name `contosoosticket` for the Azure database. They add the database to the production resource group `ContosoRG` and then specify credentials for it.

3. The on-premises MySQL database is version 5.7, so they select this version for compatibility. They use the default sizes, which match their database requirements.

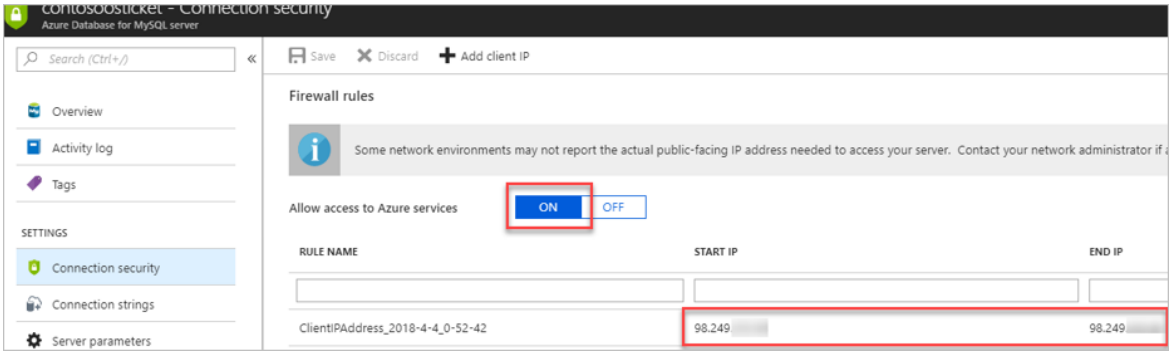


4. For **Backup Redundancy Options**, they select **Geo-Redundant**. This option allows them to restore the database in their secondary region (Central US) if an outage occurs. They can configure this option only when they provision the database.



5. They set up connection security. In the database, they select **Connection security** and then set up firewall rules to allow the database to access Azure services.

6. They add the local workstation client IP address to the start and end IP addresses. This allows the web apps to access the MySQL database, along with the database client that's performing the migration.



Step 4: Migrate the database

There are several ways to move the MySQL database. Each option requires Contoso admins to create an Azure Database for MySQL instance for the target. After they create the instance, they can migrate the database by using either of two paths:

- Step 4a: Azure Database Migration Service
- Step 4b: MySQL Workbench backup and restore

Step 4a: Migrate the database via Azure Database Migration Service

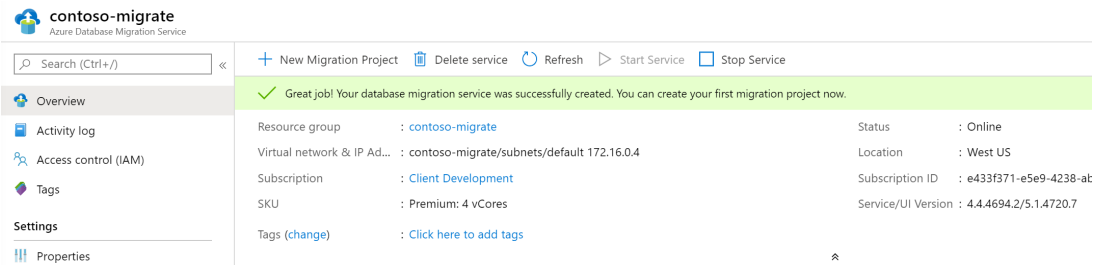
Contoso admins migrate the database via Azure Database Migration Service by following the step-by-step migration tutorial. They can perform online, offline, and hybrid (preview) migrations by using MySQL 5.6 or 5.7.

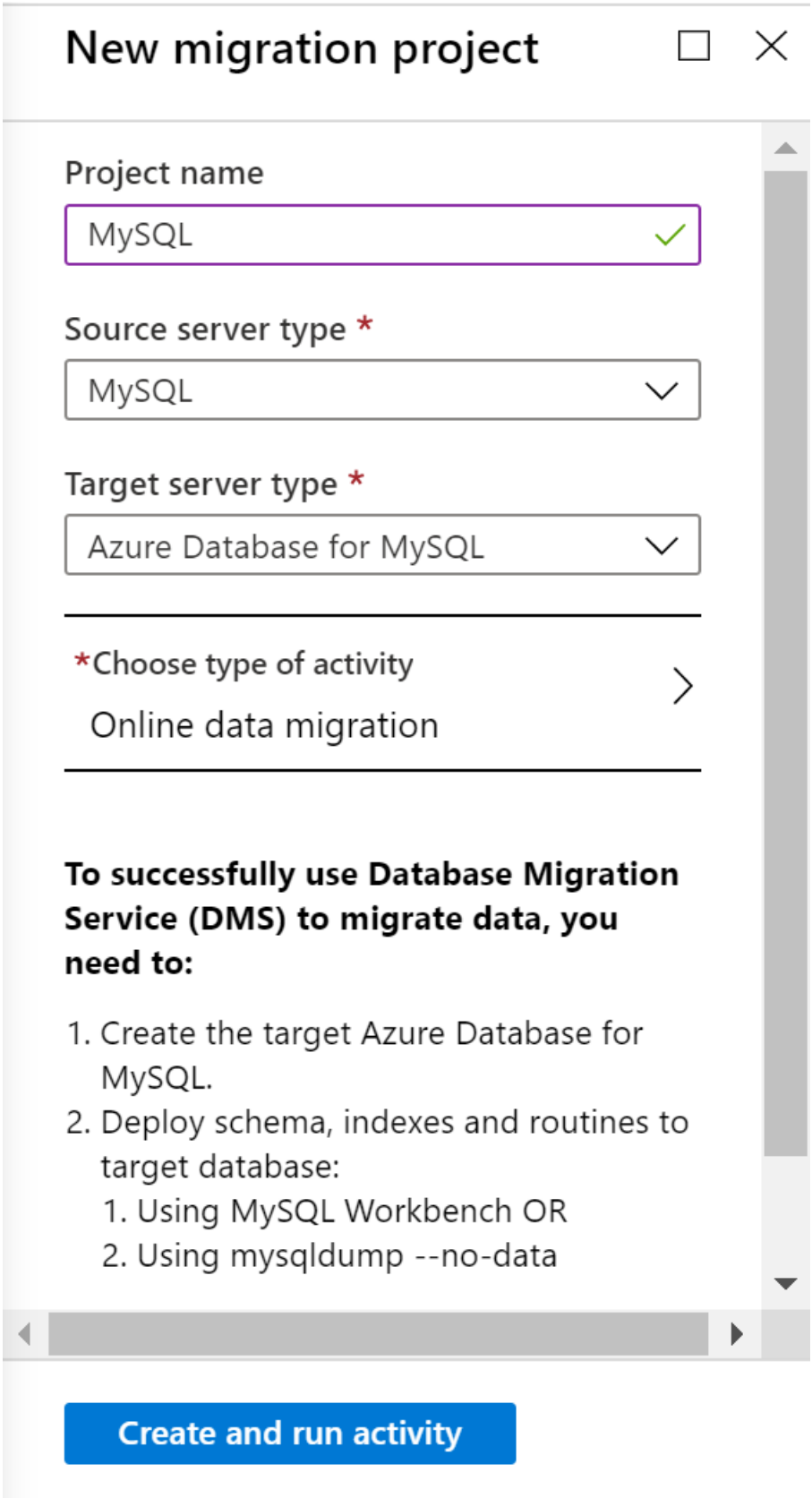
> **NOTE**
> MySQL 8.0 is supported in Azure Database for MySQL, but the Database Migration Service tool does not yet support this version.

In brief, Contoso does the following:

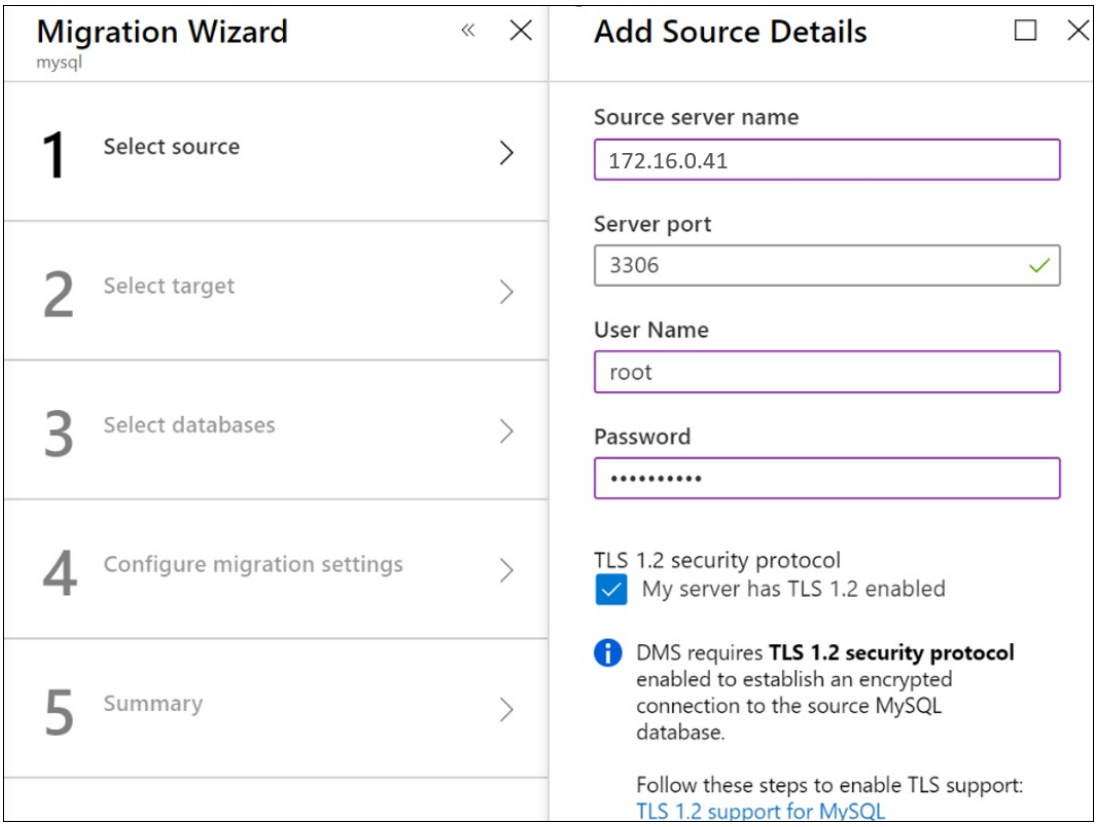
- They ensure that all migration prerequisites are met:
  - The MySQL database server source must match the version that Azure Database for MySQL supports. Azure Database for MySQL supports MySQL Community Edition, the InnoDB storage engine, and migration across source and target with the same versions.
  - They enable binary logging in `my.ini` (Windows) or `my.cnf` (Unix). Failure to do this will cause the following error in the Migration Wizard:
    ```
    Error in binary logging. Variable binlog_row_image has value 'minimal'. Please change it to 'full'.
    ```
    For more information, see the MySQL documentation.

    - The user must have the `ReplicationAdmin` role.
    - Migrate the database schemas without foreign keys and triggers.
- They create a virtual private network (VPN) that connects via ExpressRoute or VPN to the on-premises network.
- They create an Azure Database Migration Service instance with a Premium SKU that's connected to the virtual network.
- They ensure that Azure Database Migration Service can access the MySQL database via the virtual network. This entails ensuring that all incoming ports are allowed from Azure to MySQL at the virtual network level, the network VPN, and the machine that hosts MySQL.
- They run the Database Migration Service tool and then do the following:
    1. Create a migration project that's based on the Premium SKU.

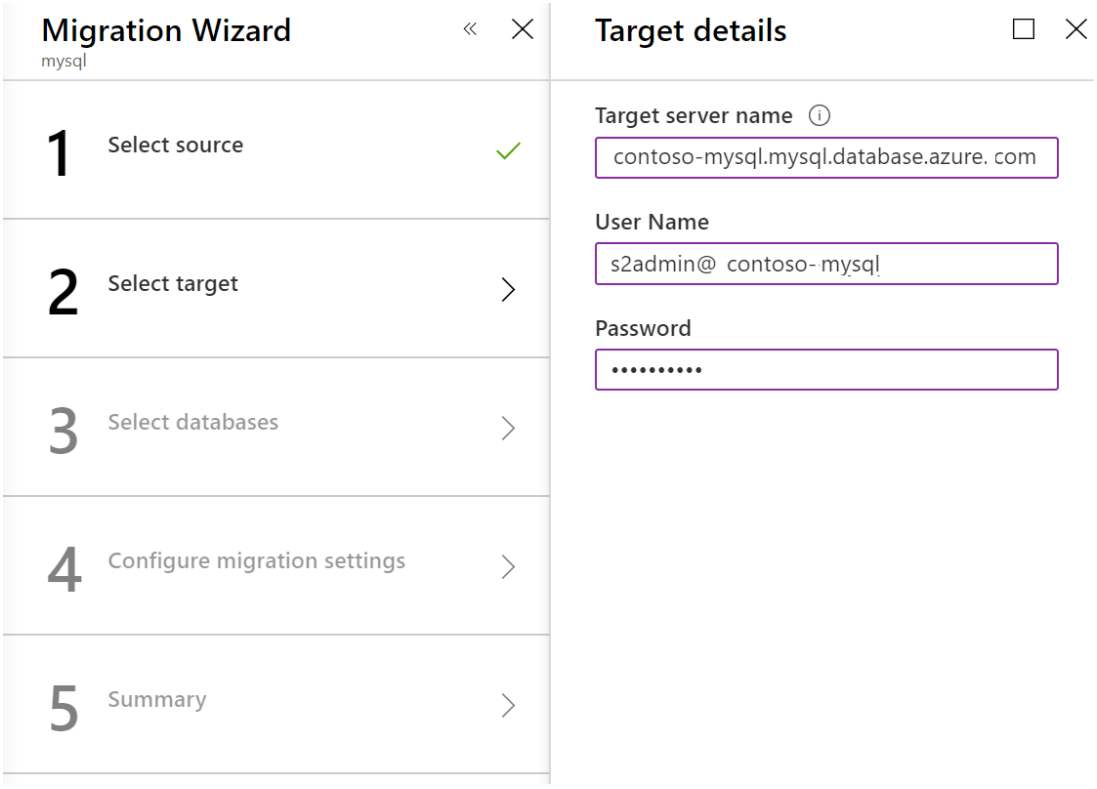




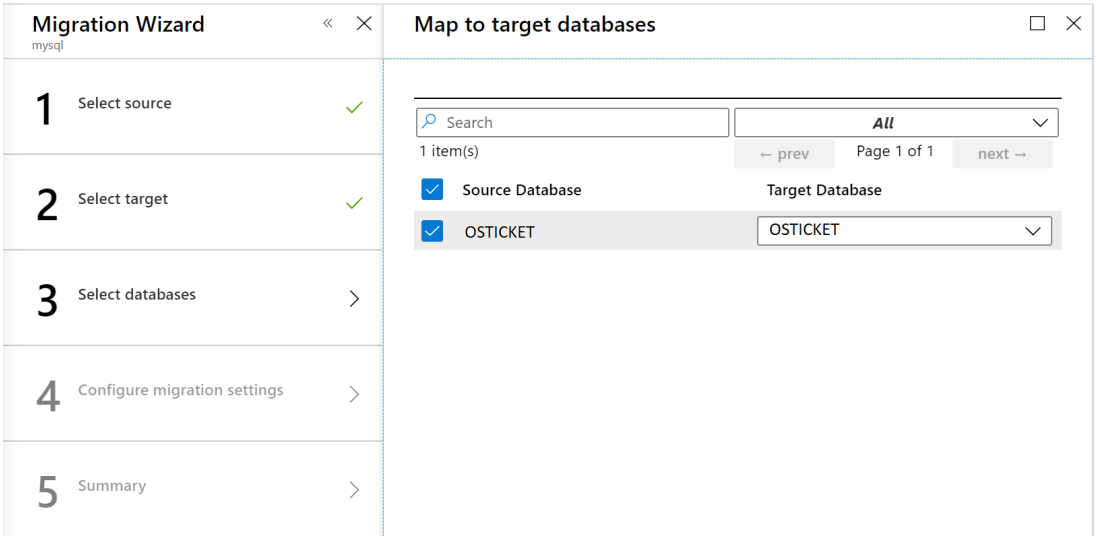
2. Add a source (on-premises database).



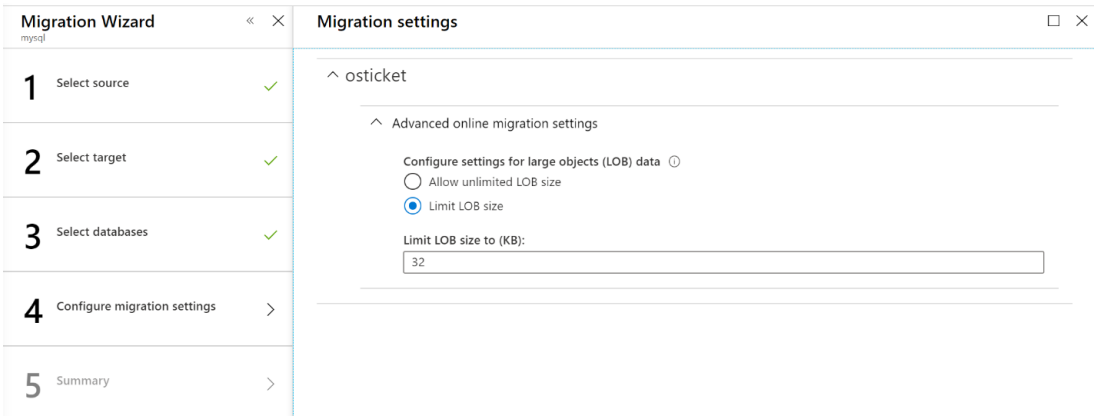
3. Select a target.



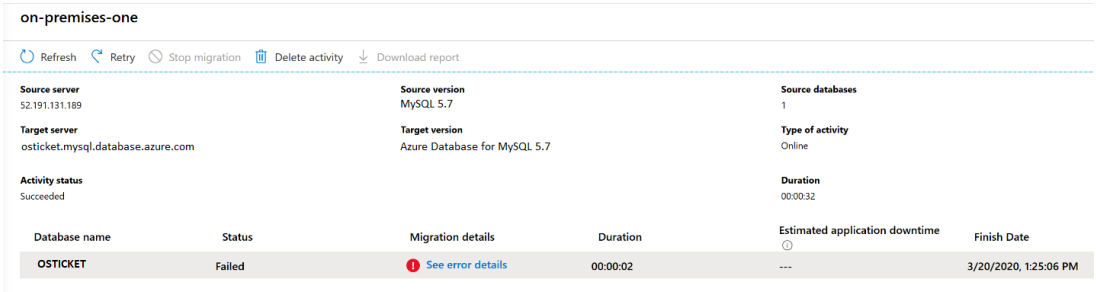
4. Select the databases to migrate.



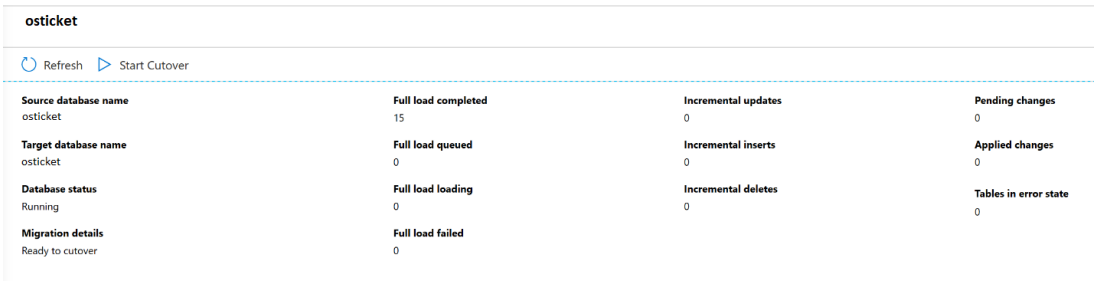
5. Configure advanced settings.

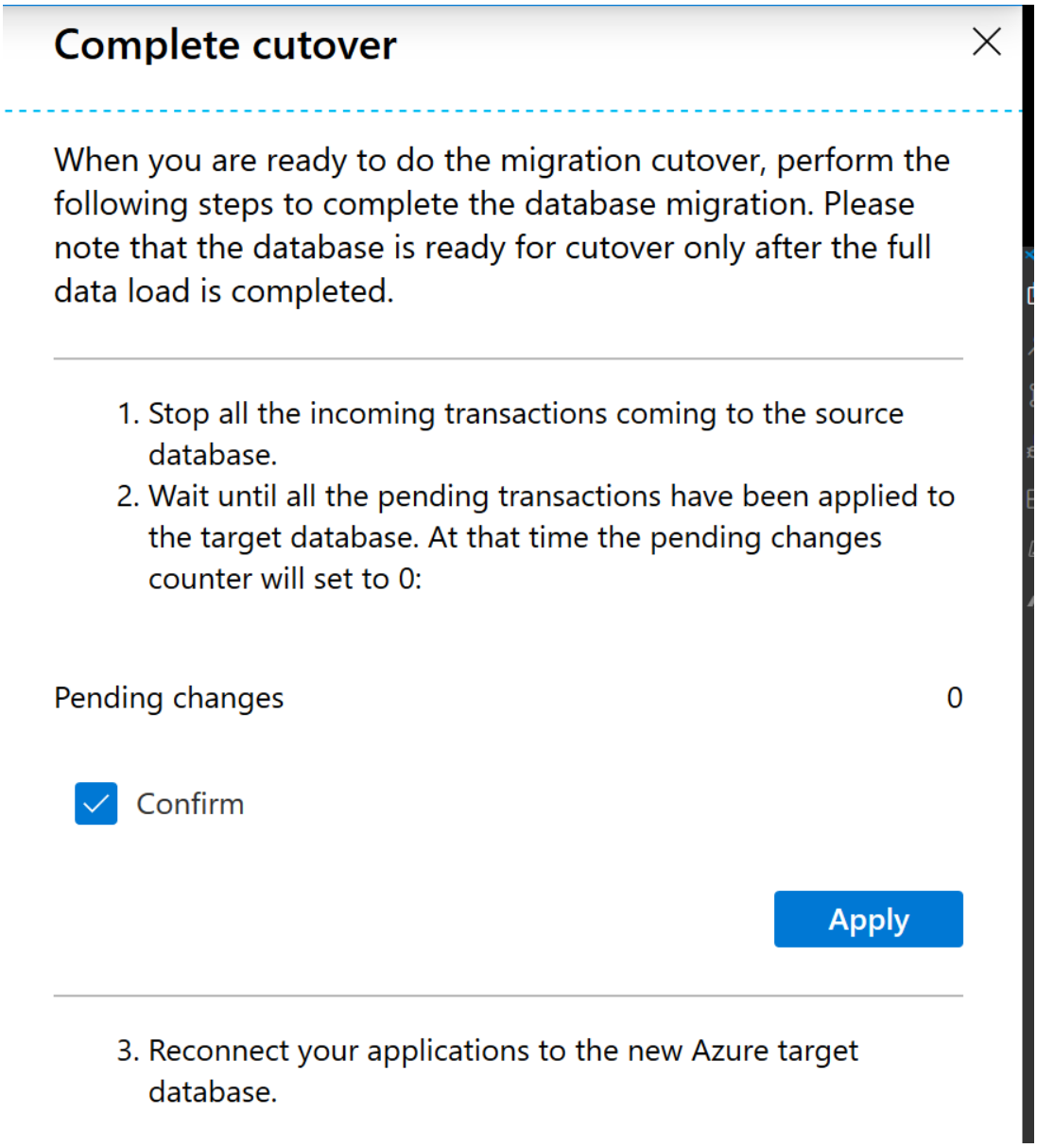


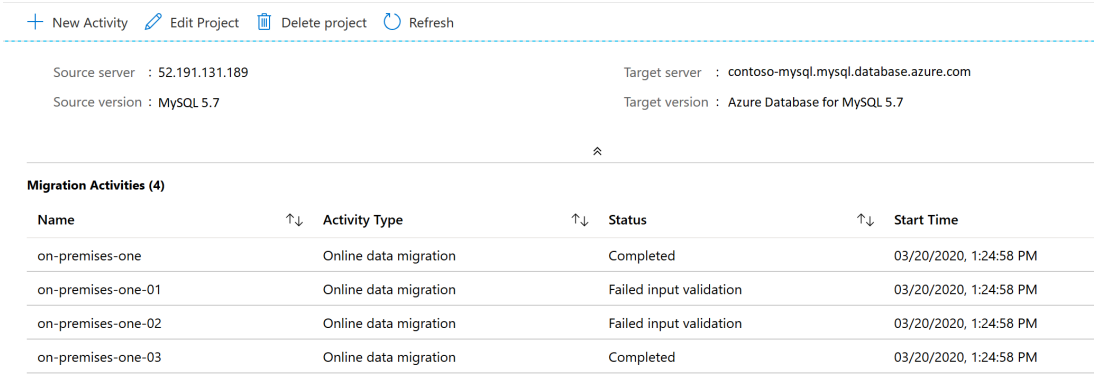
6. Start the replication and resolve any errors.



7. Perform the final cutover.







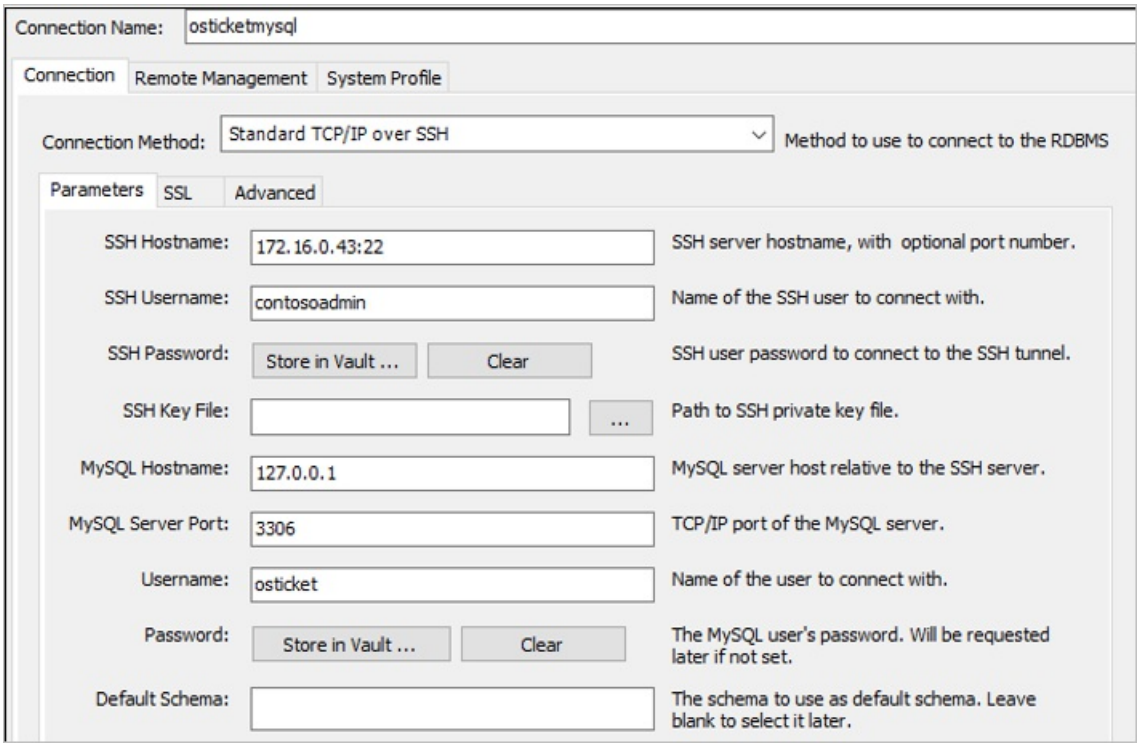
8. Reinstate any foreign keys and triggers.
9. Modify applications to use the new database.

3. Reconnect your applications to the new Azure target database.

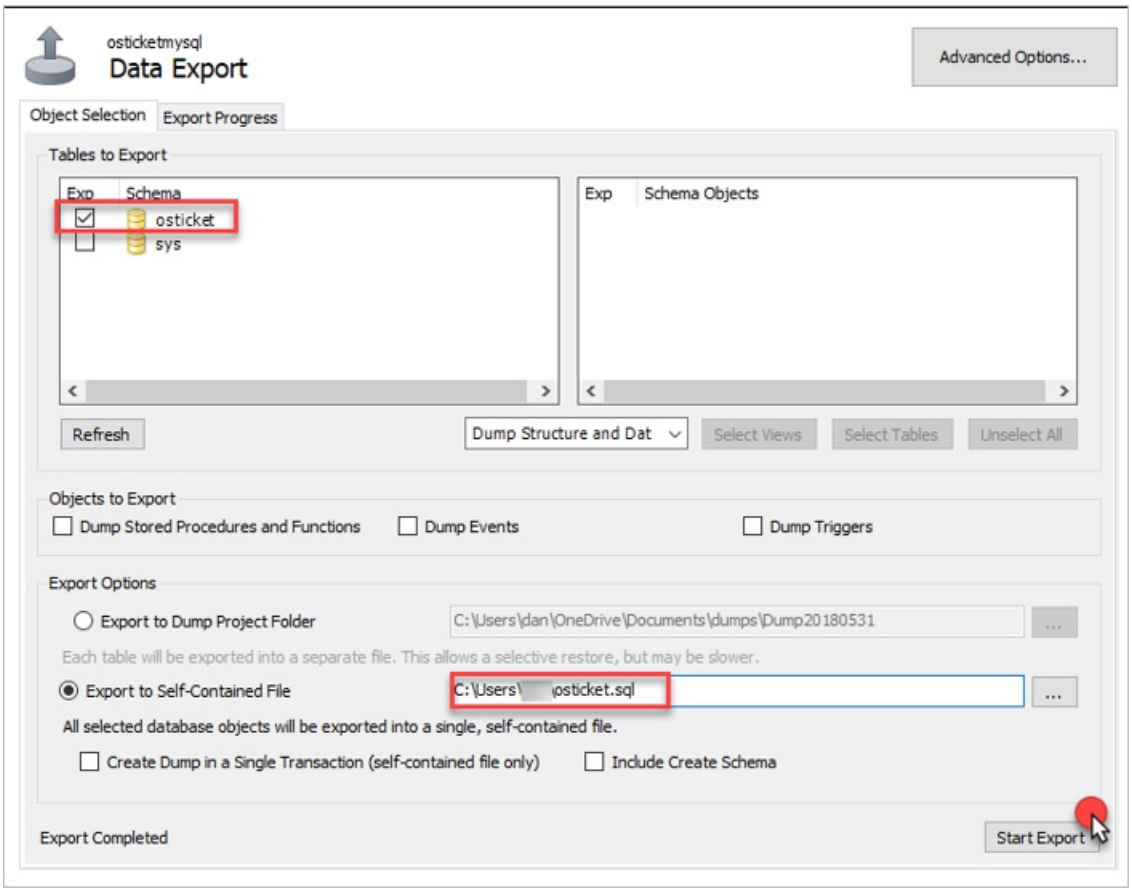
Completed

Step 4b: Migrate the database (MySQL Workbench)

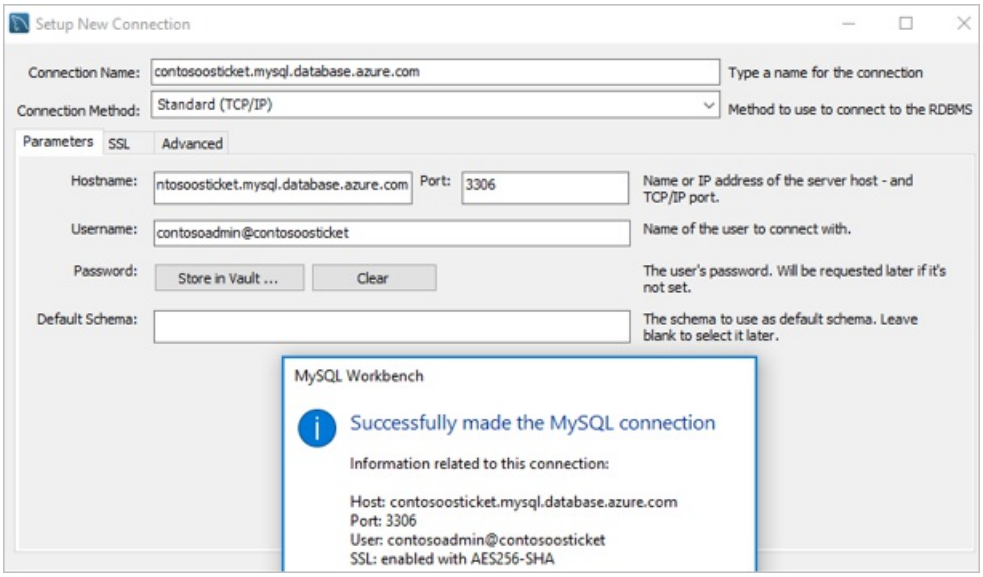
1. The Contoso admins check the prerequisites and downloads MySQL Workbench.
2. They install MySQL Workbench for Windows in accordance with the installation instructions. The machine that they install MySQL Workbench on must be accessible to the `osticketmysql` VM and to Azure via the internet.
3. In MySQL Workbench, they create a MySQL connection to `osticketmysql`.



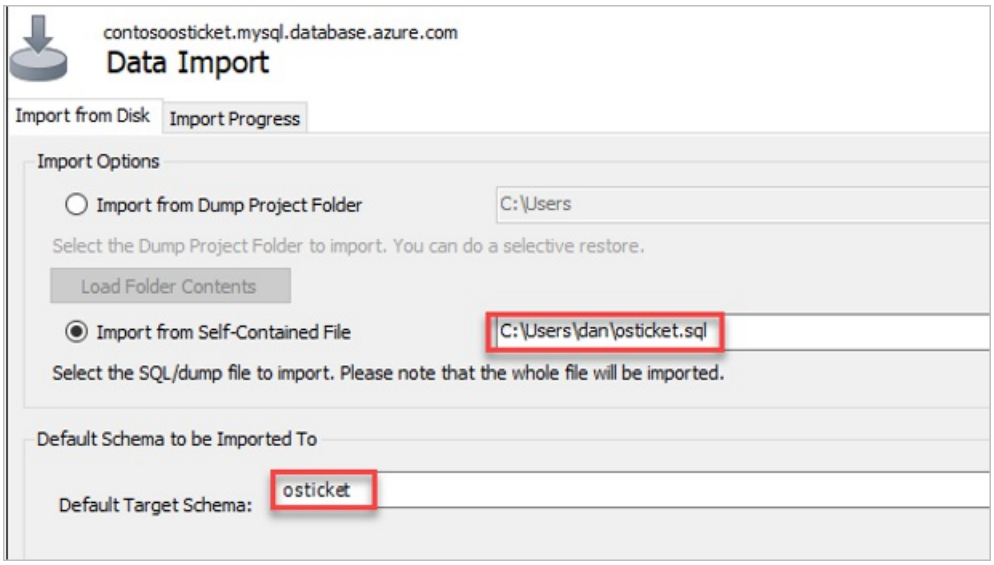
4. They export the database as `osticket` to a local self-contained file.



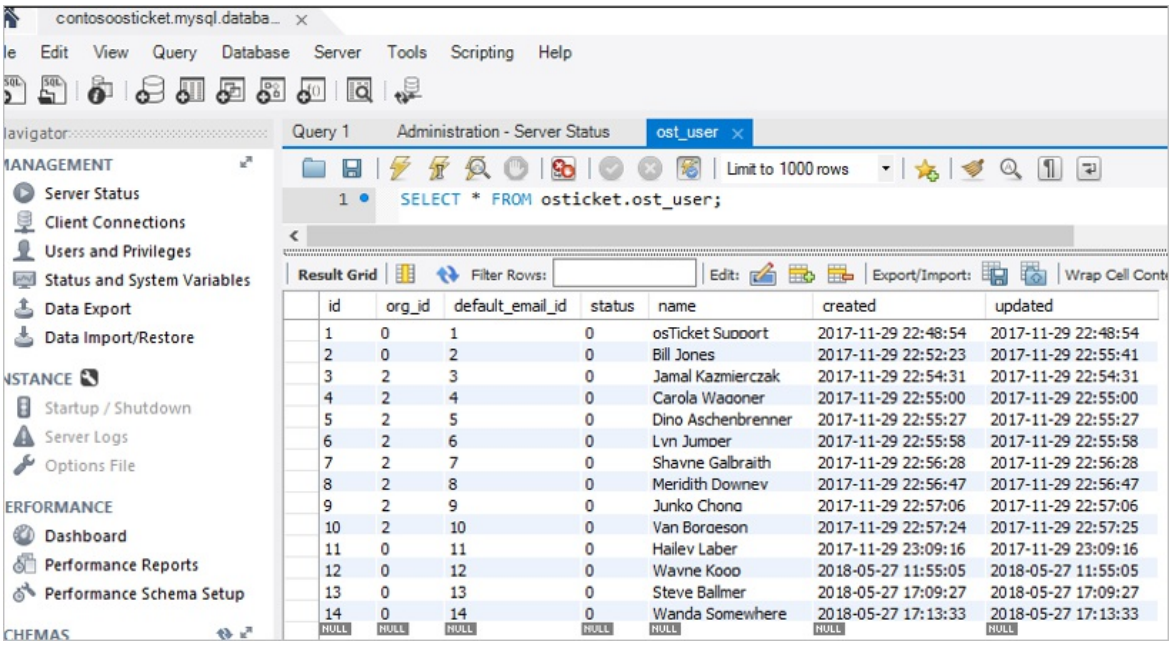
5. After they've backed up the database locally, the admins create a connection to the Azure Database for MySQL instance.

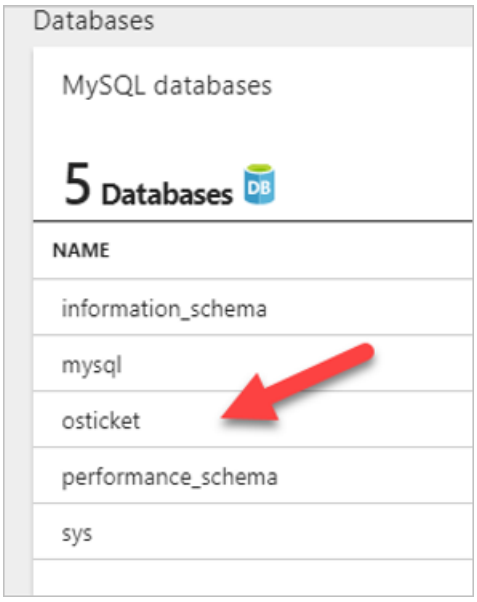


6. Now, they can import (restore) the database in the Azure Database for MySQL instance from the self-contained file. A new schema, `osticket`, is created for the instance.

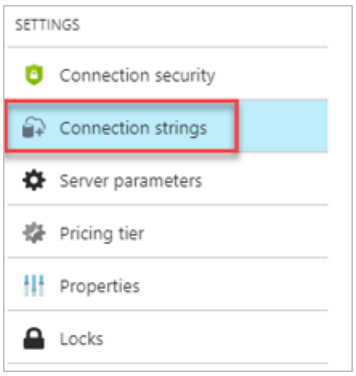


7. After they've restored the data, the admins can query it by using MySQL Workbench. The data is displayed in the Azure portal.

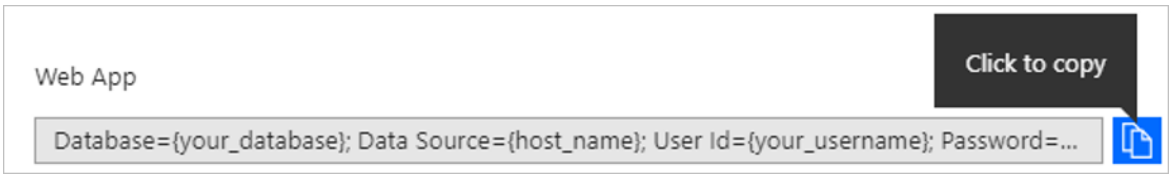




8. The admins update the database information on the web apps. On the MySQL instance, they open **Connection Strings**.



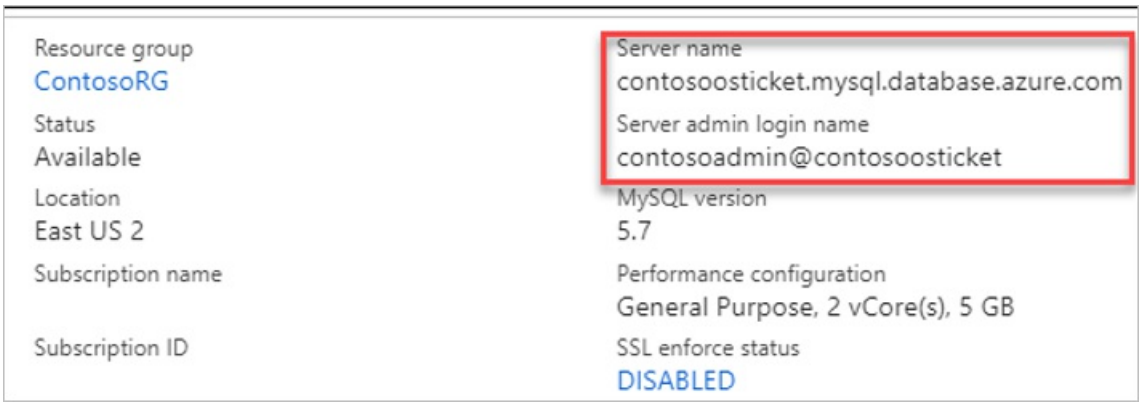
9. In the connection strings list, they select the web app settings and then copy them by selecting **Click to copy**.



10. They open a new file in Notepad, paste the string into it, and update the string to match the osTicket database, MySQL instance, and credentials settings.



11. They can verify the server name and sign-in via the **Overview** pane for the MySQL instance in the Azure portal.



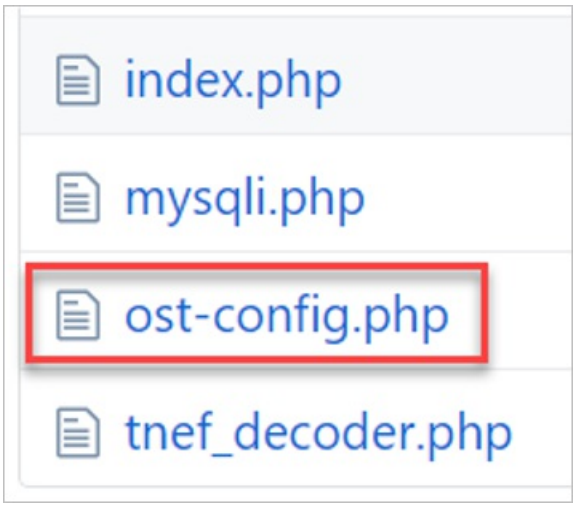
Step 5: Set up GitHub

Contoso admins create a new private GitHub repo and set up a connection to the osTicket database in Azure Database for MySQL. Then, they load the web app into Azure App Service.

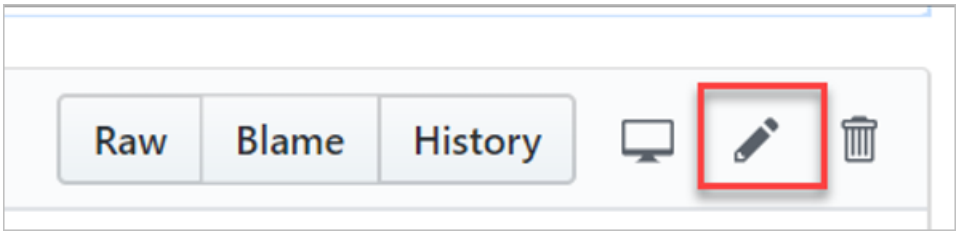
1. They browse to the osTicket software public GitHub repo and fork it to the Contoso GitHub account.



2. After they fork the repo, they browse to the `include` folder and select the `ost-config.php` file.



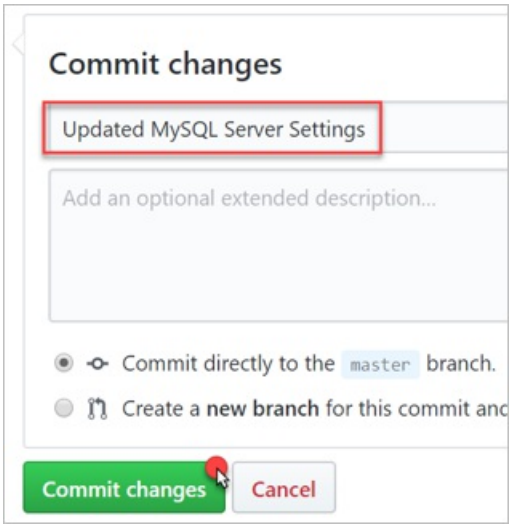
3. The file opens in the browser, and they edit it.



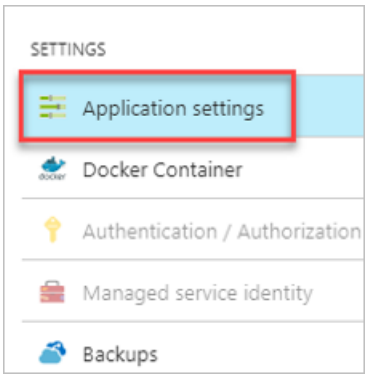
4. In the editor, the admins update the database details, specifically for `DBHOST` and `DBUSER`.

```
38  # Database Options
39  # ------------------------------------------------
40  # Mysql Login info
41  define('DBTYPE','mysql');
42  define('DBHOST','contosoosticket.mysql.database.azure.com');
43  define('DBNAME','osticket');
44  define('DBUSER','contosoadmin@contosoosticket');
45  define('DBPASS','[redacted]');
```

5. They commit the changes.



6. For each web app (`osticket-eus2` and `osticket-cus`), in the Azure portal, they select **Application settings** on the left pane and then modify the settings.



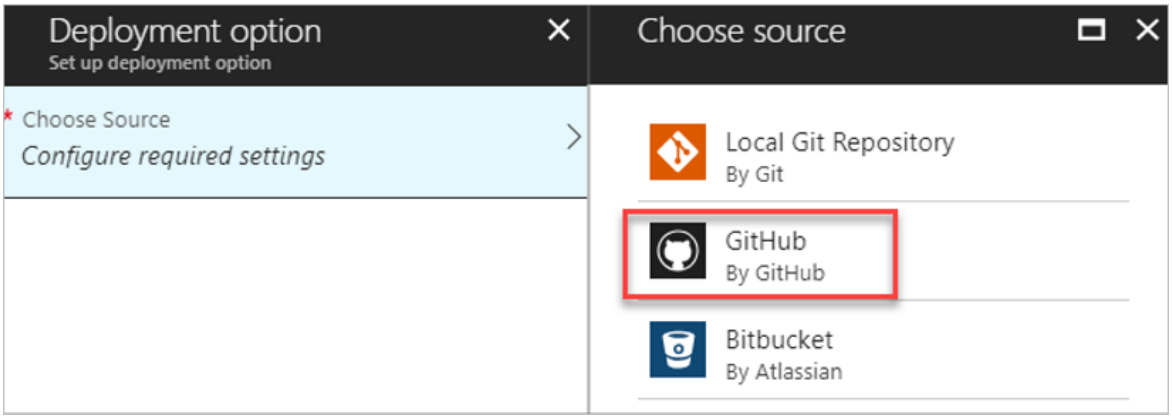
7. They enter the connection string with the name `osticket`, and copy the string from Notepad into the **value area**. They select **MySQL** in the dropdown list next to the string, and save the settings.



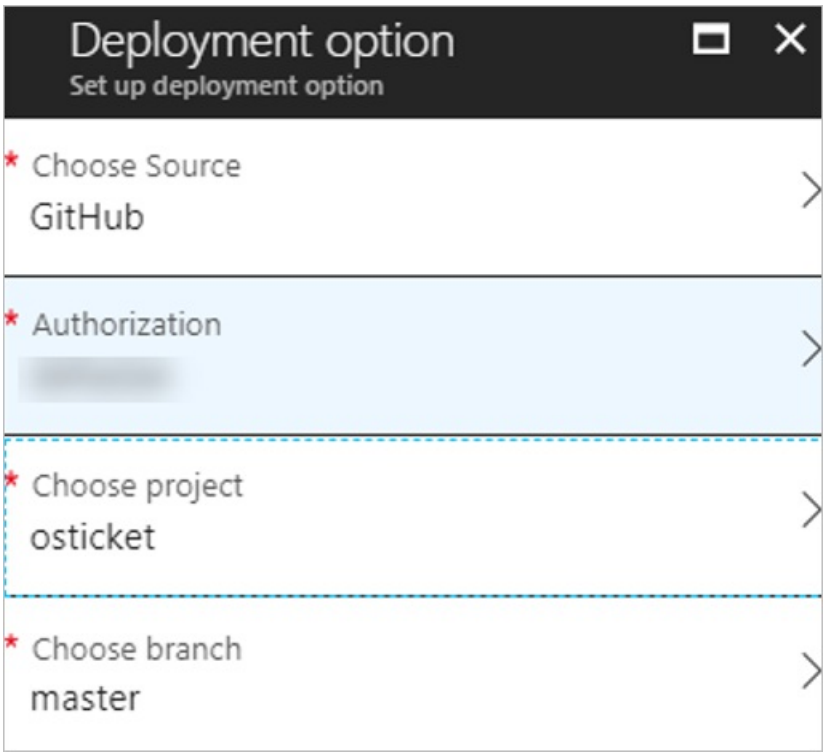
Step 6: Configure the web apps

As the final step in the migration process, Contoso admins configure the web apps with the osTicket websites.

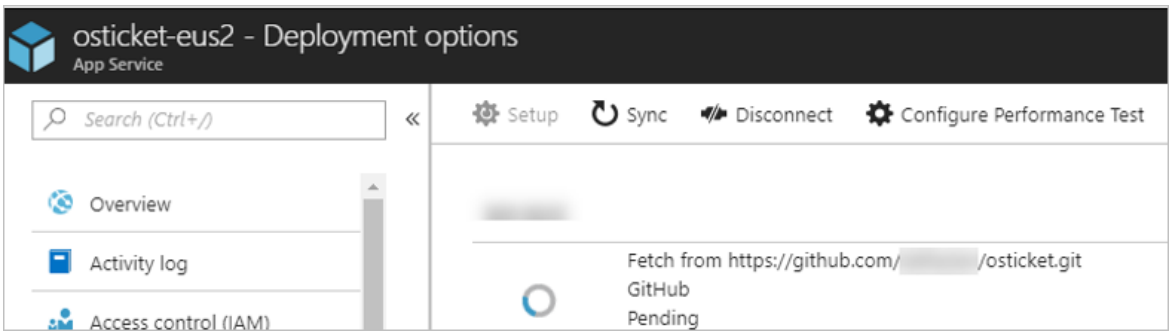
1. In the primary web app, `osticket-eus2`, they open **Deployment option** and then set the source to **GitHub**.



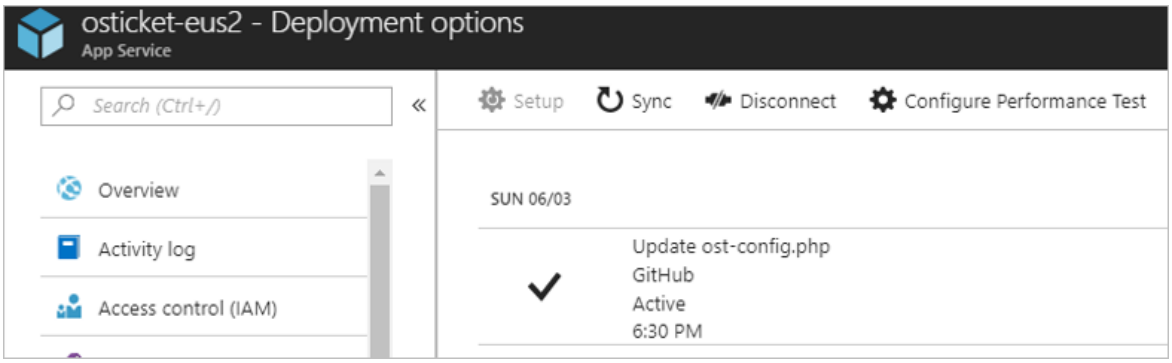
2. They select the deployment options.



3. After they set the options, the configuration shows as **Pending** in the Azure portal.

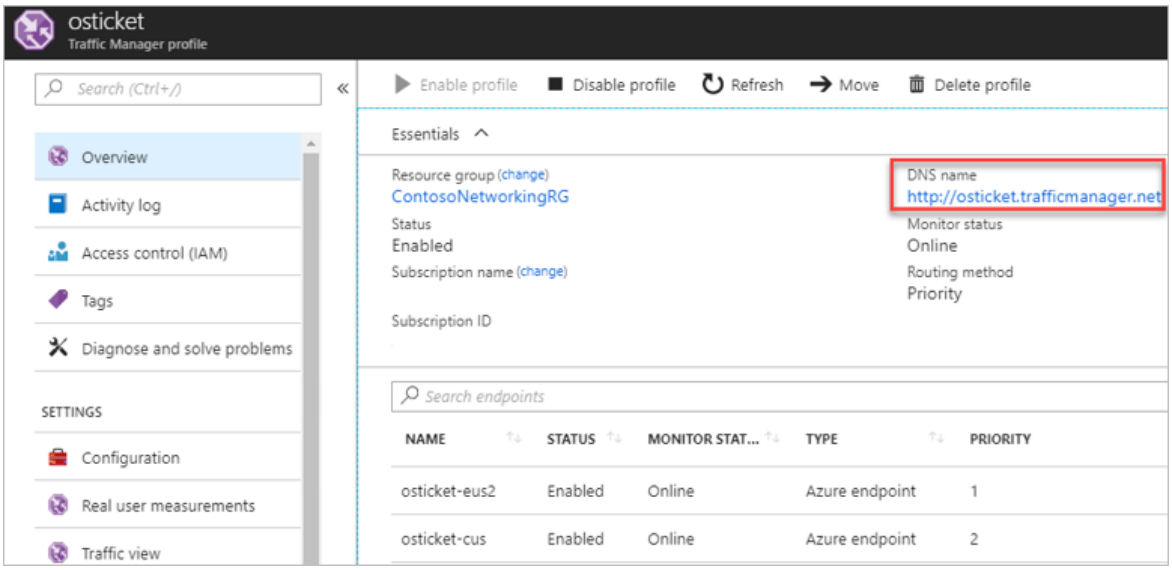


4. After the configuration is updated and the osTicket web app is loaded from GitHub to the Docker container that runs the Azure App Service, the site shows as **Active**.

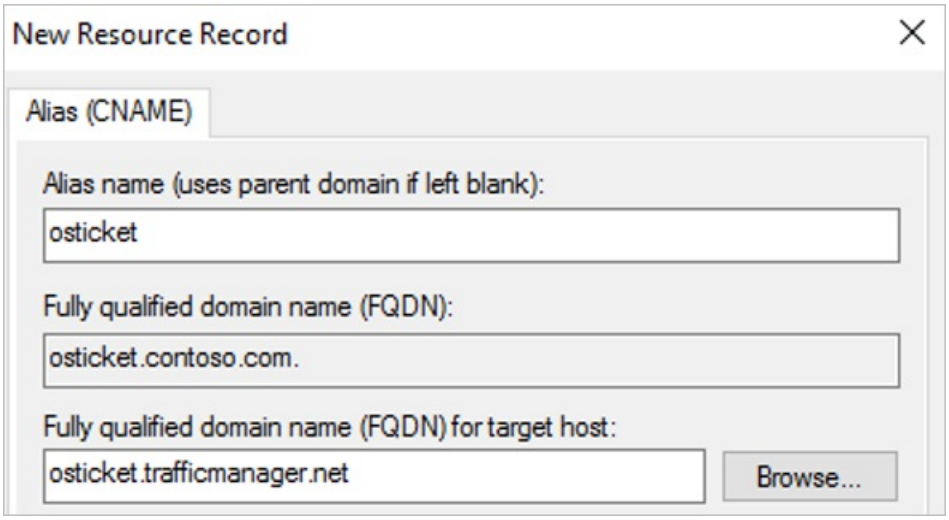


5. They repeat the preceding steps for the secondary web app, `osticket-cus`.

6. After the site is configured, it's accessible via the Traffic Manager profile. The DNS name is the new location of the osTicket application. Learn more.



7. Contoso wants to use a DNS name that's easy to remember. On the **New Resource Record** pane, they create an alias, a CNAME, and a fully qualified domain name, `osticket.contoso.com`, which points to the Traffic Manager name in the DNS on their domain controllers.



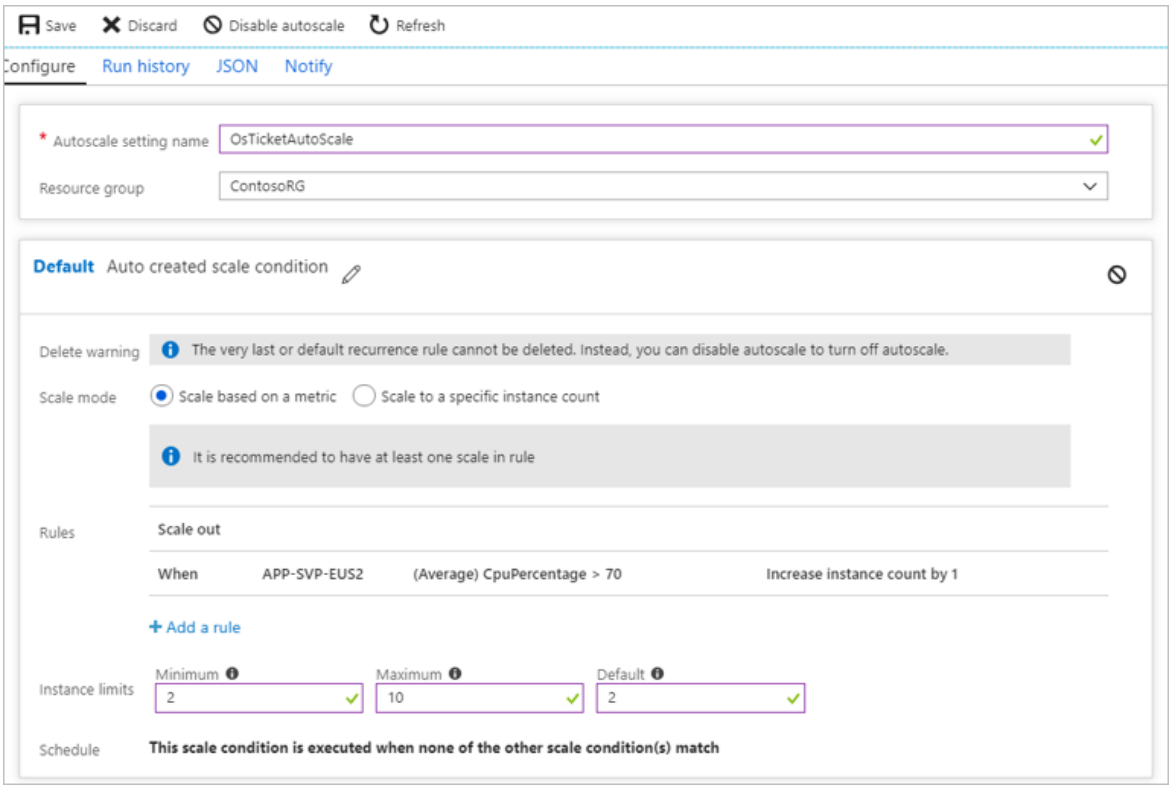
8. They configure both the `osticket-eus2` and `osticket-cus` web apps to allow the custom host names.



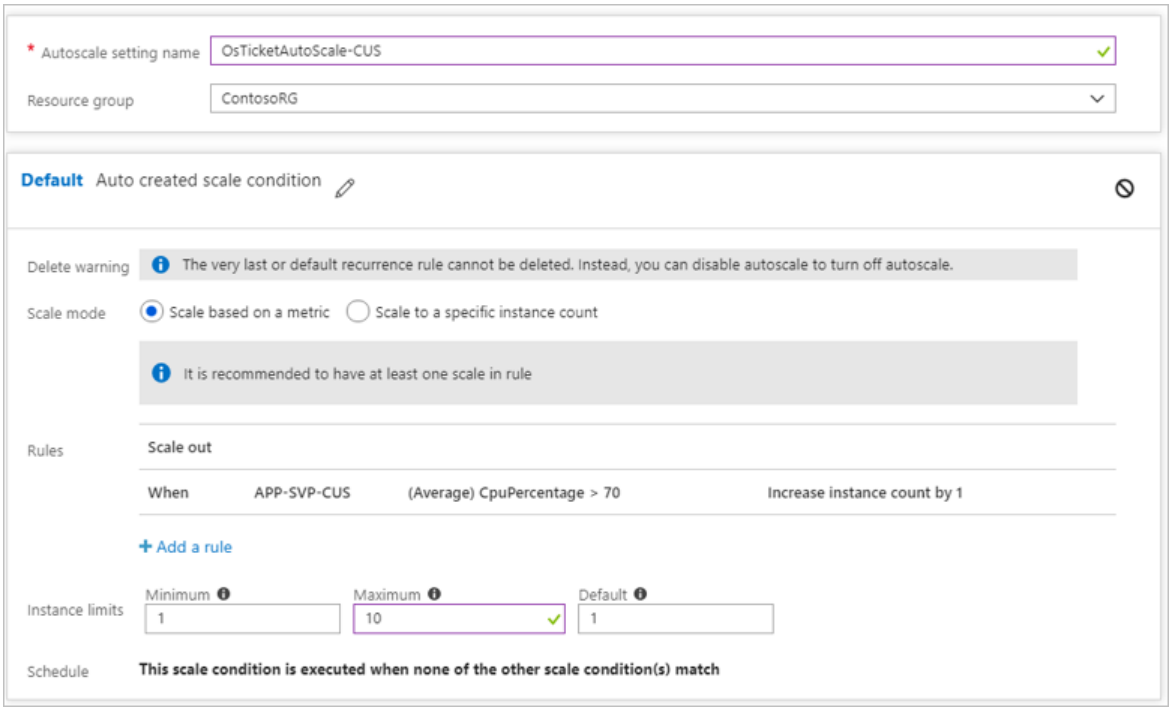
Set up autoscaling

Finally, the Contoso admins set up automatic scaling for the application. Automatic scaling ensures that, as agents use the application, the application instances increase and decrease according to business needs.

1. In App Service `APP-SVP-EUS2`, they open **Scale Unit**.
2. They configure a new autoscale setting with a single rule that increases the instance count by one when the CPU usage for the current instance is above 70 percent for 10 minutes.



3. They configure the same setting on `APP-SVP-CUS` to ensure that the same behavior applies if the application fails over to the secondary region. The only difference is that they set the default instance to 1, because this is for failovers only.



Clean up after migration

With the migration complete, the osTicket application is refactored to run in an Azure App Service web app with

continuous delivery by using a private GitHub repo. The application runs in two regions for increased resilience. The osTicket database runs in Azure Database for MySQL after migration to the PaaS platform.

To clean up after the migration, Contoso does the following:

- They remove the VMware VMs from the vCenter inventory.
- They remove the on-premises VMs from local backup jobs.
- They update internal documentation to show new locations and IP addresses.
- They review any resources that interact with the on-premises VMs, and update any relevant settings or documentation to reflect the new configuration.
- They reconfigure monitoring to point to the `osticket-trafficmanager.net` URL, to track that the application is up and running.

Review the deployment

With the application now running, Contoso needs to fully operationalize and secure their new infrastructure.

Security

The Contoso security team reviews the application to determine any security issues. They identify that the communication between the osTicket application and the MySQL database instance isn't configured for SSL. They do all this to ensure that the database traffic can't be hacked. Learn more.

Backups

- The osTicket web apps don't contain state data and thus don't require backup.
- The Contoso team doesn't need to configure backup for the database. Azure Database for MySQL automatically creates server backups and stores. The team elected to use geo-redundancy for the database, so it's resilient and production-ready. Backups can be used to restore their server to a point-in-time. Learn more.

Licensing and cost optimization

- There are no licensing issues for the PaaS deployment.
- Contoso will use Azure Cost Management + Billing to ensure that they stay within the budgets established by their IT leadership.

Rebuild an on-premises application in Azure

5/18/2021 • 23 minutes to read • Edit Online

This article demonstrates how the fictional company Contoso rebuilds a two-tier Windows .NET application that's running on VMware virtual machines (VMs) as part of a migration to Azure. Contoso migrates the front-end VM to an Azure App Service web app. Contoso builds the application back end by using microservices that are deployed to containers managed by Azure Kubernetes Service (AKS). The site interacts with Azure Functions to provide pet photo functionality.

The SmartHotel360 application used in this example is provided under an open-source license. If you want to use it for your own testing purposes, you can download it from GitHub.

Business drivers

The Contoso IT leadership team has worked closely with business partners to understand what they want to achieve with this migration:

- **Address business growth.** Contoso is growing and wants to provide differentiated experiences for customers on Contoso websites.
- **Be agile.** Contoso must be able to react faster than the changes in the marketplace to enable their success in a global economy.
- **Scale.** As the business grows successfully, the Contoso IT team must provide systems that can grow at the same pace.
- **Reduce costs.** Contoso wants to minimize licensing costs.

Migration goals

The Contoso cloud team has pinned down application requirements for this migration. These requirements were used to determine the best migration method:

- The application in Azure must remain as critical as it is today on-premises. It should perform well and scale easily.
- The application shouldn't use infrastructure as a service (IaaS) components. Everything should be built to use platform as a service (PaaS) or serverless services.
- Application builds should run in cloud services, and containers should reside in a private, enterprise-wide registry in the cloud.
- The API service that's used for pet photos should be accurate and reliable in the real world, because decisions made by the application must be honored in their hotels. Any pet granted access is allowed to stay at the hotels.
- To meet requirements for a DevOps pipeline, Contoso will use a Git repository in Azure Repos for source code management. Automated builds and releases will be used to build code and deploy to Azure App Service, Azure Functions, and AKS.
- Separate continuous integration/continuous development (CI/CD) pipelines are needed for microservices on the back end and for the website on the front end.
- The back-end services and the front-end web app have different release cycles. To meet this requirement, Contoso will deploy two different pipelines.
- Contoso needs management approval for all front-end website deployment, and the CI/CD pipeline must provide this.

Solution design

After pinning down their goals and requirements, Contoso designs and reviews a deployment solution, and identifies the migration process, including the Azure services that will be used for the migration.

Current application

- The SmartHotel360 on-premises application is tiered across two VMs (`WEBVM` and `SQLVM`).
- The VMs are located on VMware ESXi host `contosohost1.contoso.com` (version 6.5).
- The VMware environment is managed by vCenter Server 6.5 (`vcenter.contoso.com`), running on a VM.
- Contoso has an on-premises datacenter (`contoso-datacenter`), with an on-premises domain controller (`contosodc1`).
- The on-premises VMs in the Contoso datacenter will be decommissioned after the migration is done.

Proposed architecture

- The front end of the application is deployed as an Azure App Service web app in the primary Azure region.
- An Azure function provides uploads of pet photos, and the site interacts with this functionality.
- The pet photo function uses the Computer Vision API of Azure Cognitive Services along with Azure Cosmos DB.
- The back end of the site is built by using microservices. These microservices will be deployed to containers that are managed in AKS.
- Containers will be built using Azure DevOps and then pushed to Azure Container Registry.
- For now, Contoso will manually deploy the web app and function code by using Visual Studio.
- Contoso will deploy microservices by using a PowerShell script that calls Kubernetes command-line tools.

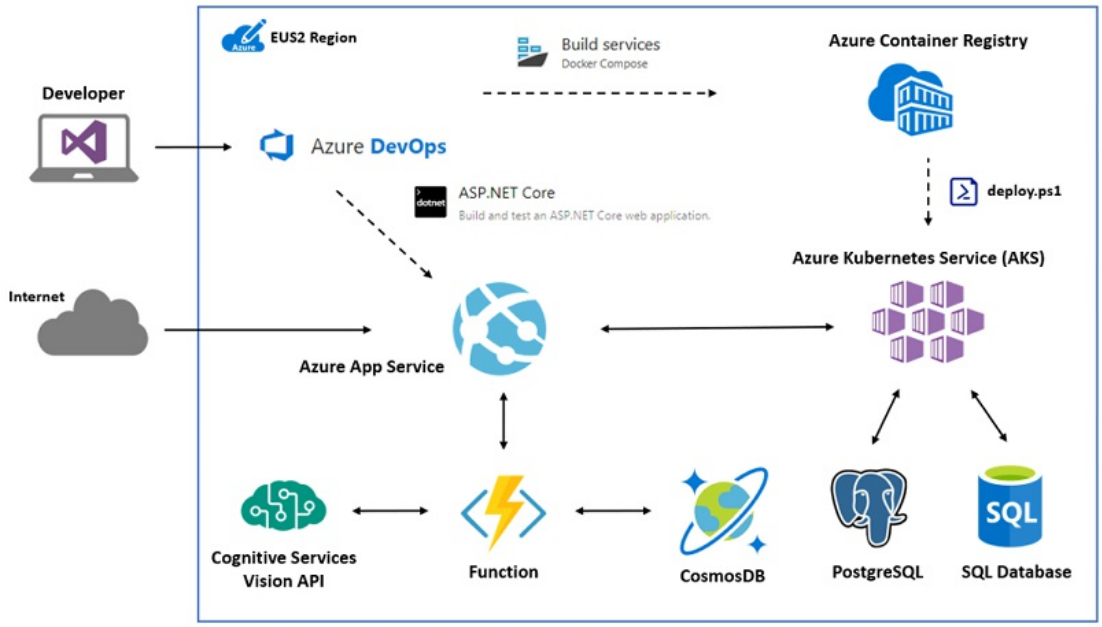


Figure 1: Scenario architecture.

Solution review

Contoso evaluates the proposed design by putting together a pros and cons list.

CONSIDERATION	DETAILS
Pros	Using PaaS and serverless solutions for the end-to-end deployment significantly reduces the management time that Contoso must provide. Moving to a microservices-based architecture allows Contoso to easily extend the solution over time. New functionality can be brought online without disrupting any of the existing solutions' code bases. The web app will be configured with multiple instances, with no single point of failure. Autoscaling will be enabled so that the application can handle differing traffic volumes. With the move to PaaS services, Contoso can retire out-of-date solutions that run on the Windows Server 2008 R2 operating system. Azure Cosmos DB has built-in fault tolerance, which requires no configuration by Contoso. This means that the data tier is no longer a single point of failover.
Cons	Containers are more complex than other migration options. The learning curve could be an issue for Contoso. They introduce a new level of complexity that provides value in spite of the curve. The operations team at Contoso needs to ramp up to understand and support Azure, containers, and microservices for the application. Contoso hasn't fully implemented DevOps for the entire solution. Contoso needs to consider that for the deployment of services to AKS, Azure Functions, and Azure App Service.

Migration process

1. Contoso provisions Azure Container Registry, AKS, and Azure Cosmos DB.
2. Contoso provisions the infrastructure for the deployment, including the Azure App Service web app, storage account, function, and API.
3. After the infrastructure is in place, Contoso builds their microservices container images by using Azure DevOps, which pushes the images to the container registry.
4. Contoso deploys these microservices to AKS by using a PowerShell script.
5. Finally, Contoso deploys the function and web app.

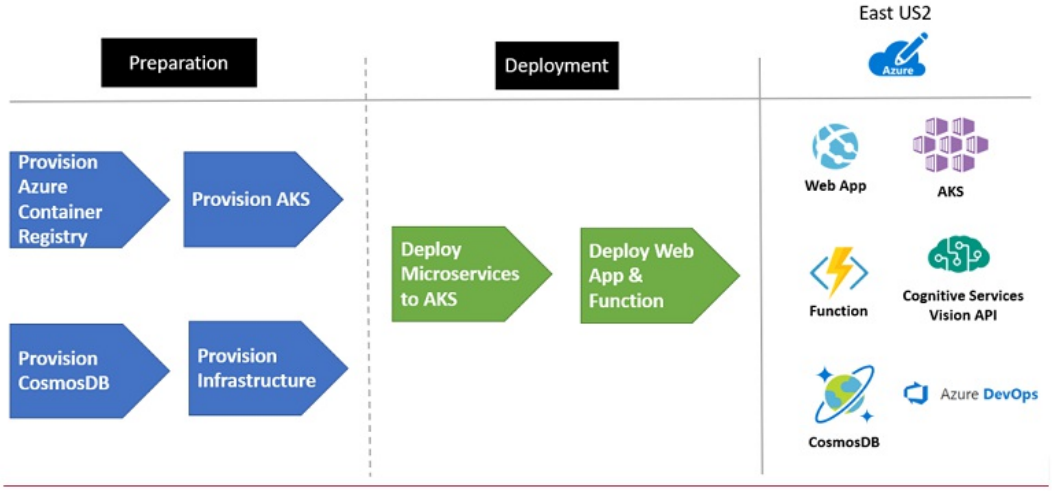


Figure 2: The migration process.

Azure services

SERVICE	DESCRIPTION	COST
Azure Kubernetes Service	Simplifies Kubernetes management, deployment, and operations. Provides a fully managed Kubernetes container orchestration service.	AKS is a free service. Pay for only the VMs and the associated storage and networking resources that are consumed. Learn more.
Azure Functions	Accelerates development with an event-driven, serverless compute experience. Scale on demand.	Pay only for consumed resources. Plan is billed based on per-second resource consumption and executions. Learn more.
Azure Container Registry	Stores images for all types of container deployments.	Cost is based on features, storage, and usage duration. Learn more.
Azure App Service	Quickly build, deploy, and scale enterprise-grade web, mobile, and API apps that run on any platform.	App Service plans are billed on a per-second basis. Learn more.

Prerequisites

Here's what Contoso needs for this scenario:

REQUIREMENTS	DETAILS
Azure subscription	• Contoso created subscriptions in an earlier article. If you don't have an Azure subscription, create a free account. • If you create a free account, you're the admin of your subscription and can perform all actions. • If you use an existing subscription and you're not the admin, you need to work with the admin to assign Owner or Contributor permissions to you.
Azure infrastructure	• Learn how Contoso set up an Azure infrastructure.

REQUIREMENTS	DETAILS
Developer prerequisites	Contoso needs the following tools on a developer workstation: • Visual Studio Community 2017 version 15.5 • .NET workload, enabled • Git • Azure PowerShell • The Azure CLI • Docker Community Edition (Windows 10) or Docker Enterprise Edition (Windows Server), set to use Windows containers

Scenario steps

Here's how Contoso will run the migration:

- **Step 1: Provision AKS and Azure Container Registry.** Contoso provisions the managed AKS cluster and the container registry by using PowerShell.
- **Step 2: Build Docker containers.** Contoso sets up continuous integration (CI) for Docker containers by using Azure DevOps and pushes the containers to the container registry.
- **Step 3: Deploy back-end microservices.** Contoso deploys the rest of the infrastructure that will be used by back-end microservices.
- **Step 4: Deploy front-end infrastructure.** Contoso deploys the front-end infrastructure, including Blob Storage for the pet phones, Azure Cosmos DB, and the Computer Vision API.
- **Step 5: Migrate the back end.** Contoso deploys microservices and runs them on AKS to migrate the back end.
- **Step 6: Publish the front end.** Contoso publishes the SmartHotel360 application to Azure App Service along with the function app to be called by the pet service.

Provision back-end resources

Contoso admins run a deployment script to create the managed Kubernetes cluster by using AKS and Azure Container Registry. The instructions for this section use the `SmartHotel360-Backend` GitHub repository. The repository contains all the software for this part of the deployment.

Ensure that prerequisites are met

Before they start, Contoso admins ensure that all prerequisite software is installed on the dev machine they're using for the deployment. They clone the repository locally to the dev machine by using Git:

```
git clone https://github.com/Microsoft/SmartHotel360-Backend.git
```

Provision AKS and Azure Container Registry

The Contoso admins provision AKS and Azure Container Registry as follows:

1. In Visual Studio Code, they open the folder and go to the `/deploy/k8s` directory, which contains the script `gen-aks-env.ps1`.
2. They run the script to create the managed Kubernetes cluster, using AKS and Azure Container Registry.

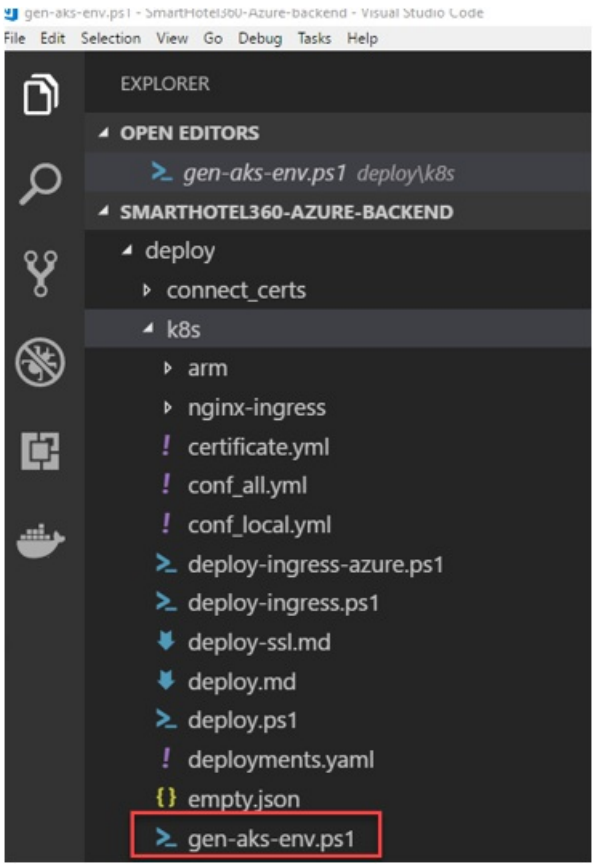


Figure 3: Creating the managed Kubernetes cluster.

3. With the file open, they update the $location parameter to `eastus2`, and save the file.

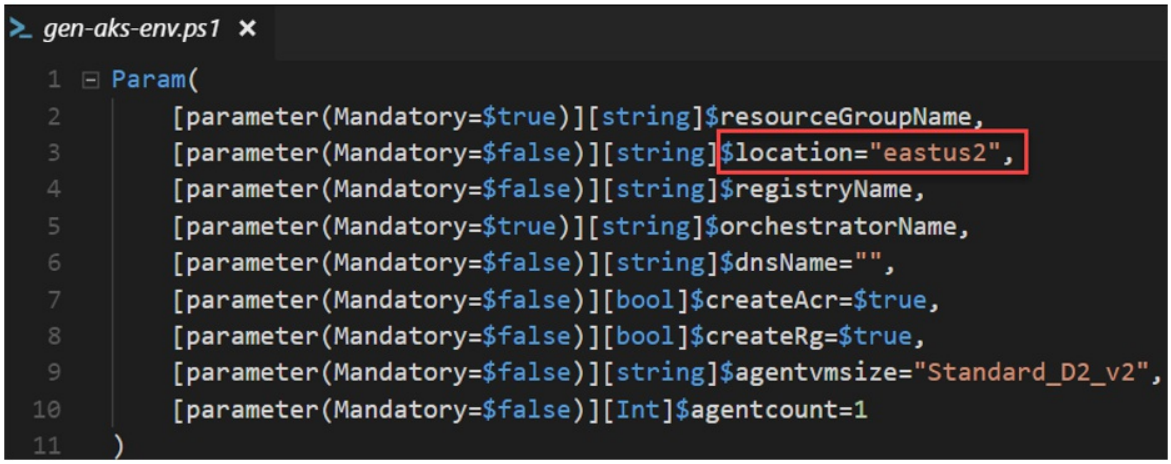


Figure 4: Saving the file.

4. They select **View** > **Integrated terminal** to open the integrated terminal in Visual Studio Code.

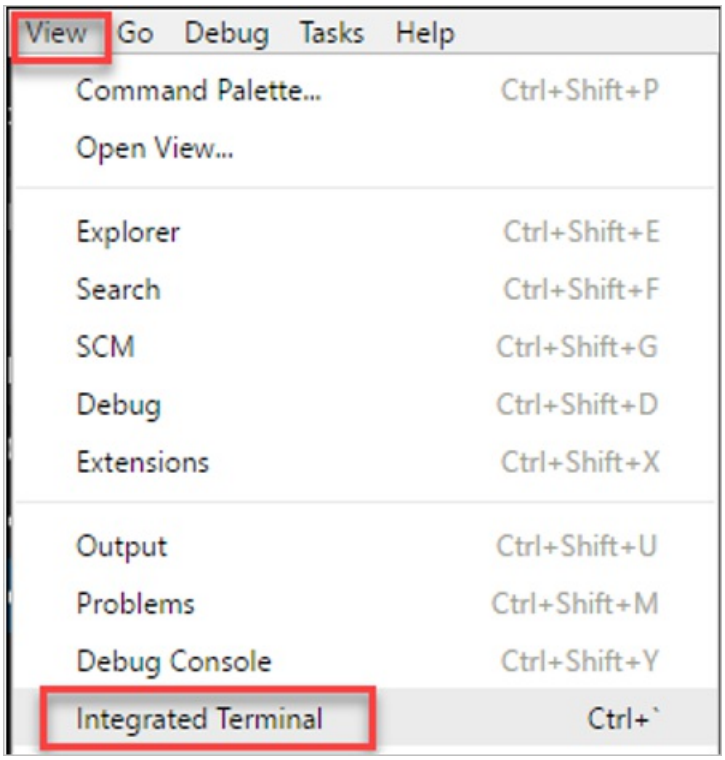


Figure 5: The terminal in Visual Studio Code.

5. In the PowerShell integrated terminal, they sign in to Azure using the `Connect-AzureRmAccount` command. For more information, see Get started with PowerShell.

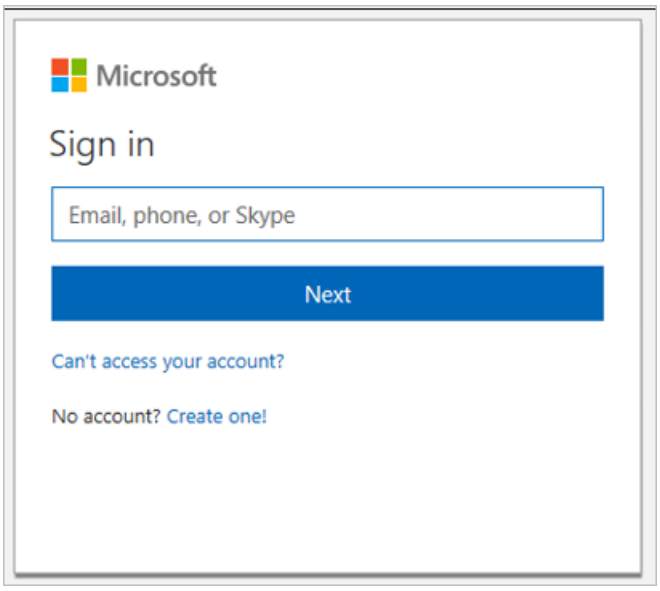


Figure 6: The PowerShell integrated terminal.

6. They authenticate the Azure CLI by running the `az login` command and following the instructions to authenticate using their web browser. Learn more about signing in with the Azure CLI.

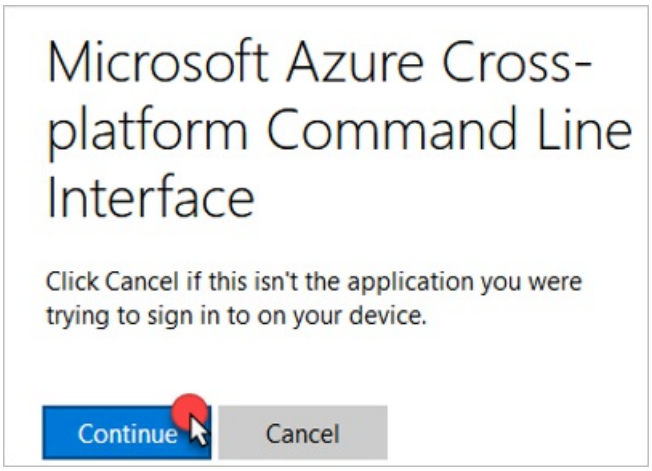


Figure 7: Authenticating the Azure CLI.

7. They run the following command while passing the resource group name of `ContosoRG`, the name of the AKS cluster `smarthotel-aks-eus2`, and the new registry name.

```
.\gen-aks-env.ps1 -resourceGroupName ContosoRg -orchestratorName smarthotelakseus2 -registryName
smarthotelacreus2
```

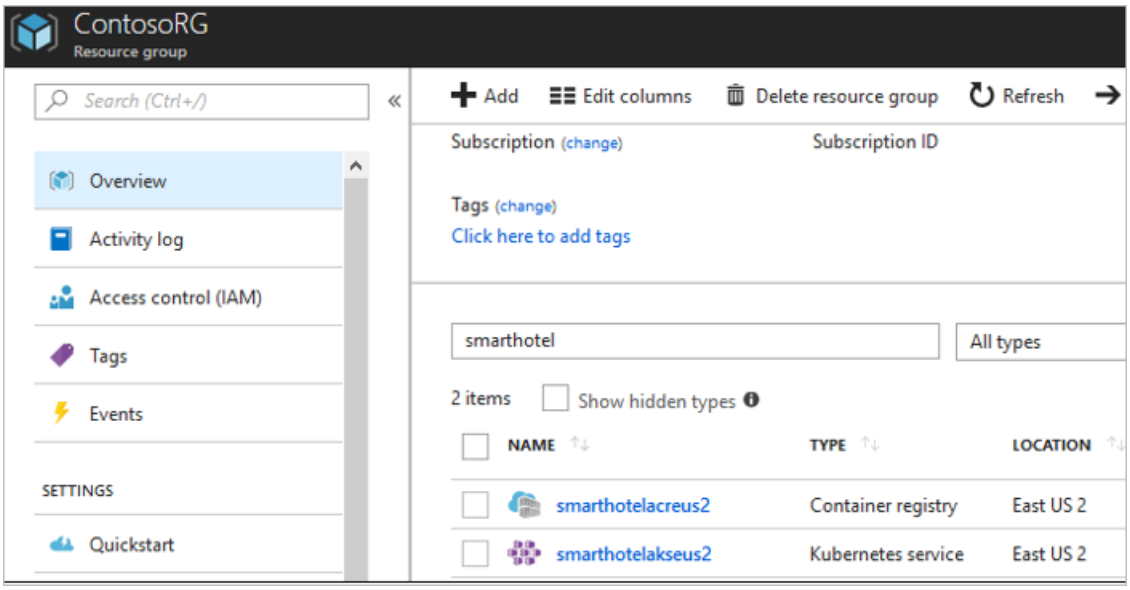


Figure 8: Running the command.

8. Azure creates another resource group that contains the resources for the AKS cluster.

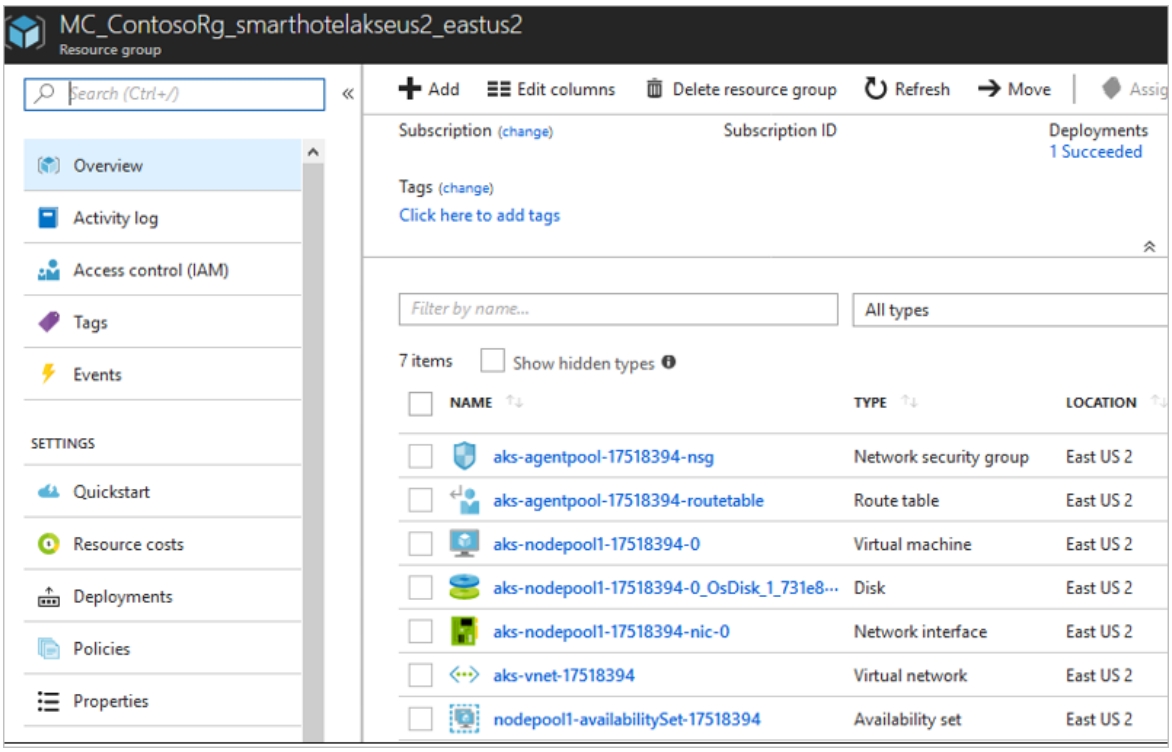


Figure 9: Azure creating a resource group.

9. After the deployment is finished, they install the `kubectl` command-line tool. The tool is already installed on the Azure Cloud Shell.

```
az aks install-cli
```

10. They verify the connection to the cluster by running the `kubectl get nodes` command. The node has the same name as the VM in the automatically created resource group.

```
NAME                       STATUS    ROLES     AGE       VERSION
aks-nodepool1-17518394-0   Ready     agent     42m       v1.9.6
```

Figure 10: Verifying the connection to the cluster.

11. They run the following command to start the Kubernetes dashboard:

```
az aks browse --resource-group ContosoRG --name smarthotelakseus2
```

12. A browser tab opens to the dashboard. This is a tunneled connection that uses the Azure CLI.

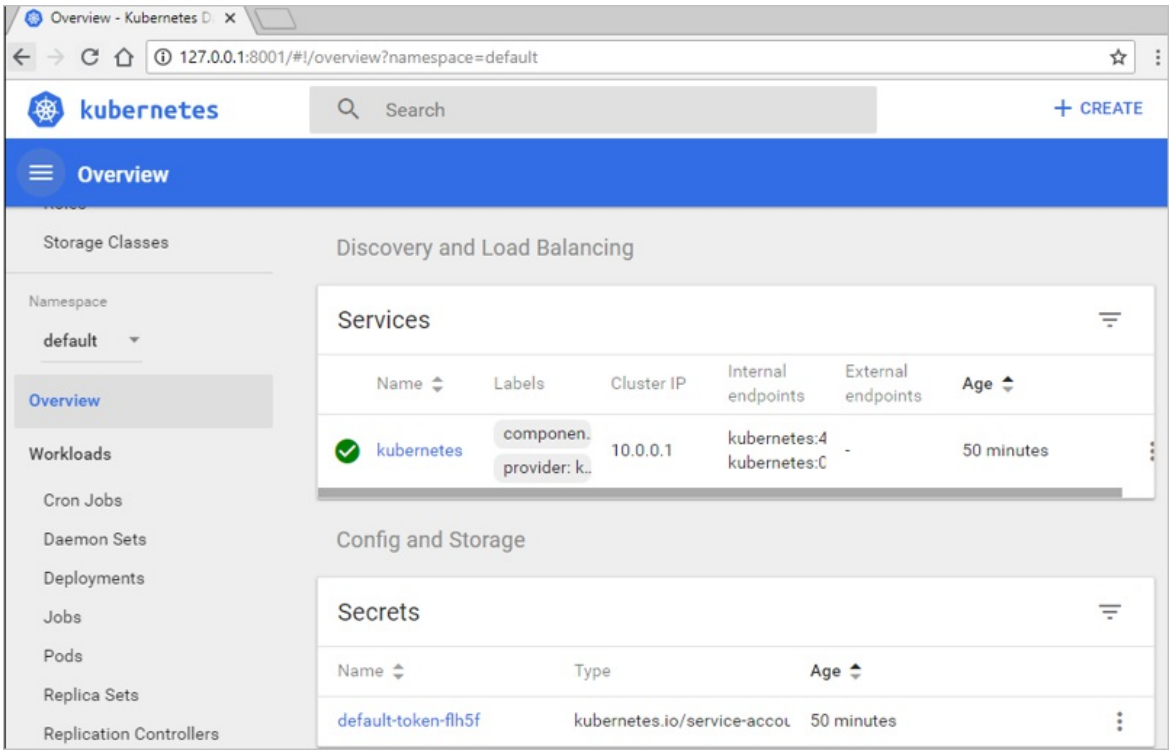


Figure 11: A tunneled connection.

Step 2: Configure the back-end pipeline

Create an Azure DevOps project and build

Contoso creates an Azure DevOps project, configures a CI build to create the container, and then pushes it to the container registry. The instructions in this section use the `SmartHotel360-Backend` repository.

1. From `visualstudio.com`, they create a new organization (`contosodevops360.visualstudio.com`) and configure it to use Git.
2. They create a new project (`SmartHotelBackend`), selecting **Git** for version control and **Agile** for the workflow.

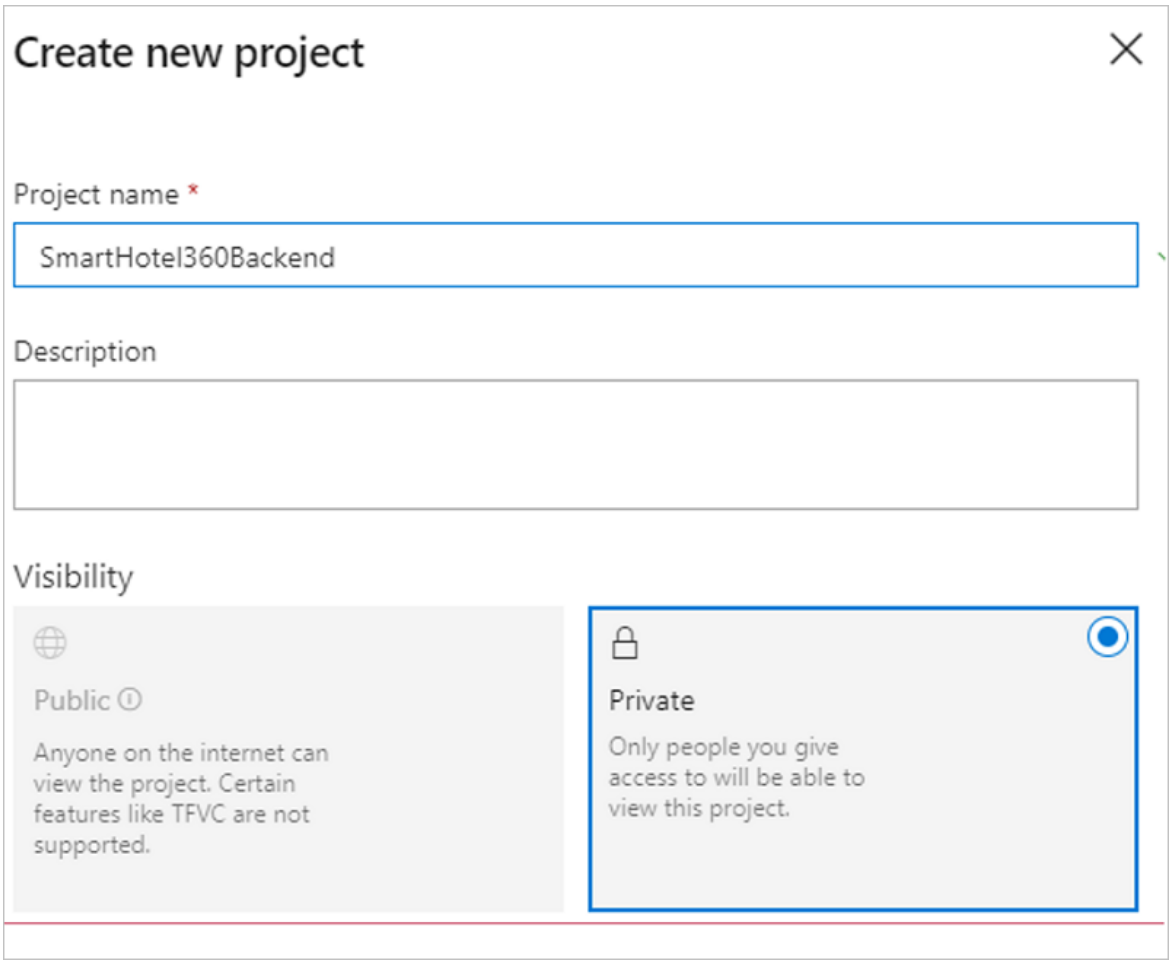


Figure 12: Creating a new project.

3. They import the GitHub repo.

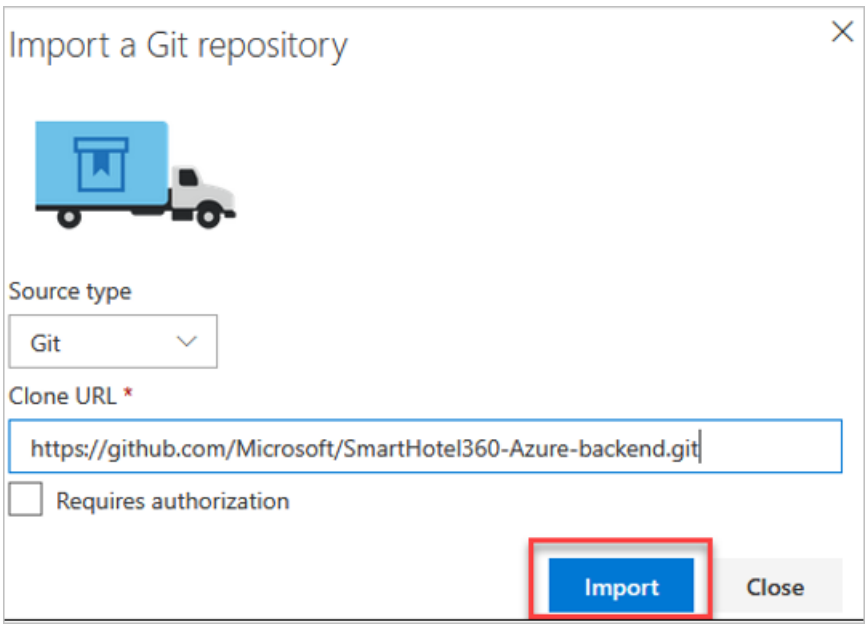


Figure 13: Importing the GitHub repo.

4. In **Pipelines**, they select **Build** and create a new pipeline by using Azure Repos Git as a source from the repository.

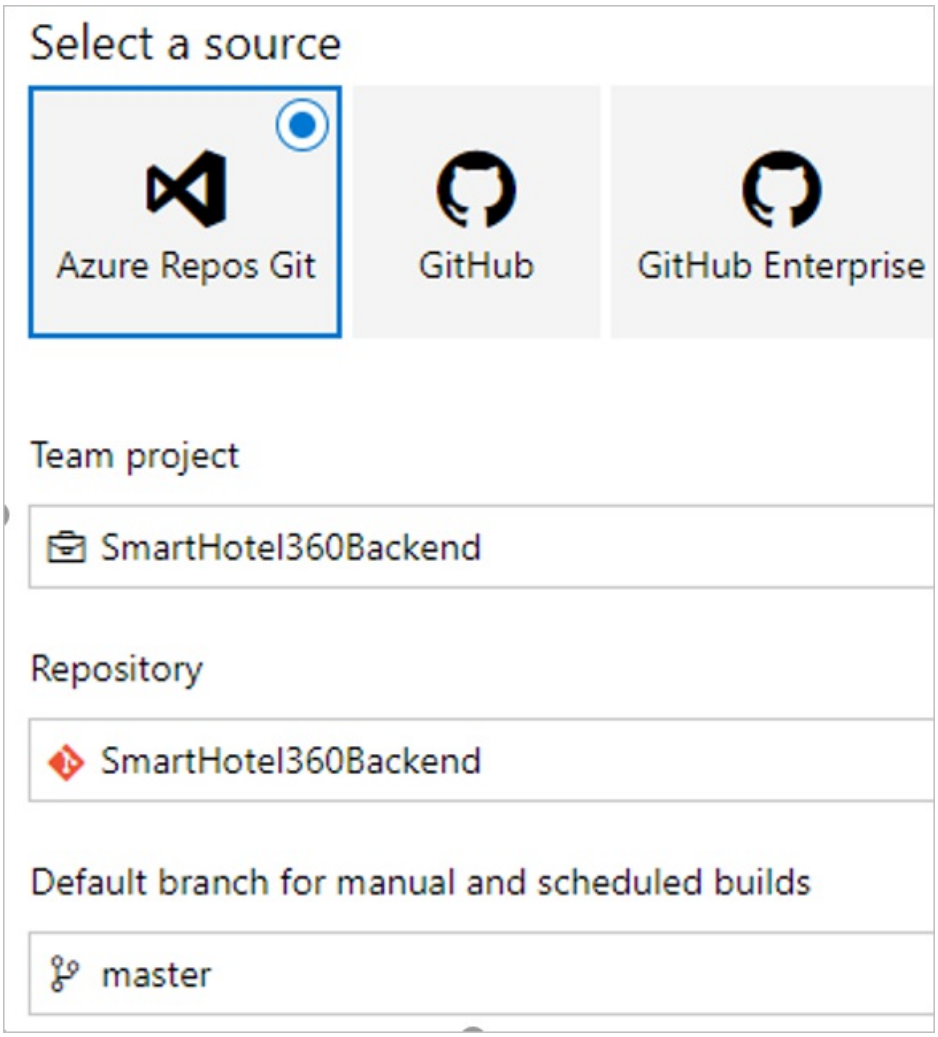


Figure 14: Creating a new pipeline.

5. They select **Empty job**.

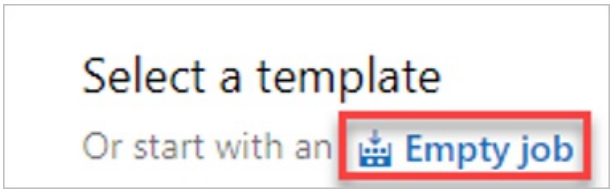


Figure 15: Starting with an empty job.

6. They select **Hosted Linux Preview** for the build pipeline.

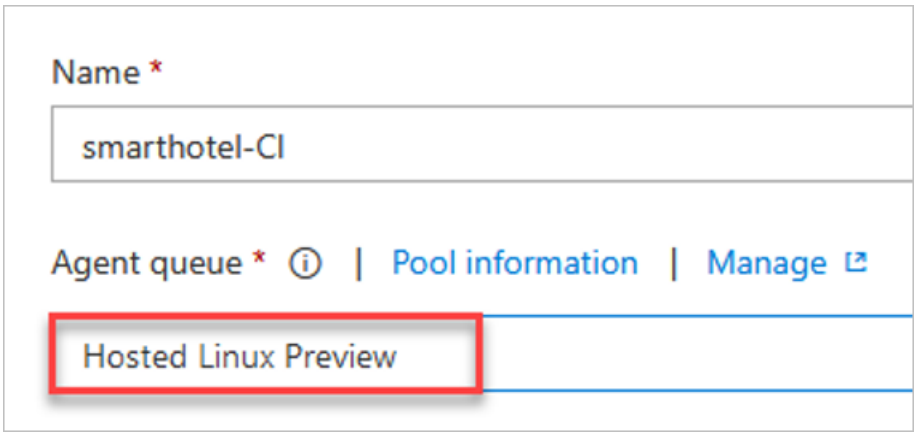


Figure 16: Setting up the build pipeline.

7. In **Step 1**, they add a **Docker Compose** task. This task builds the Docker Compose.

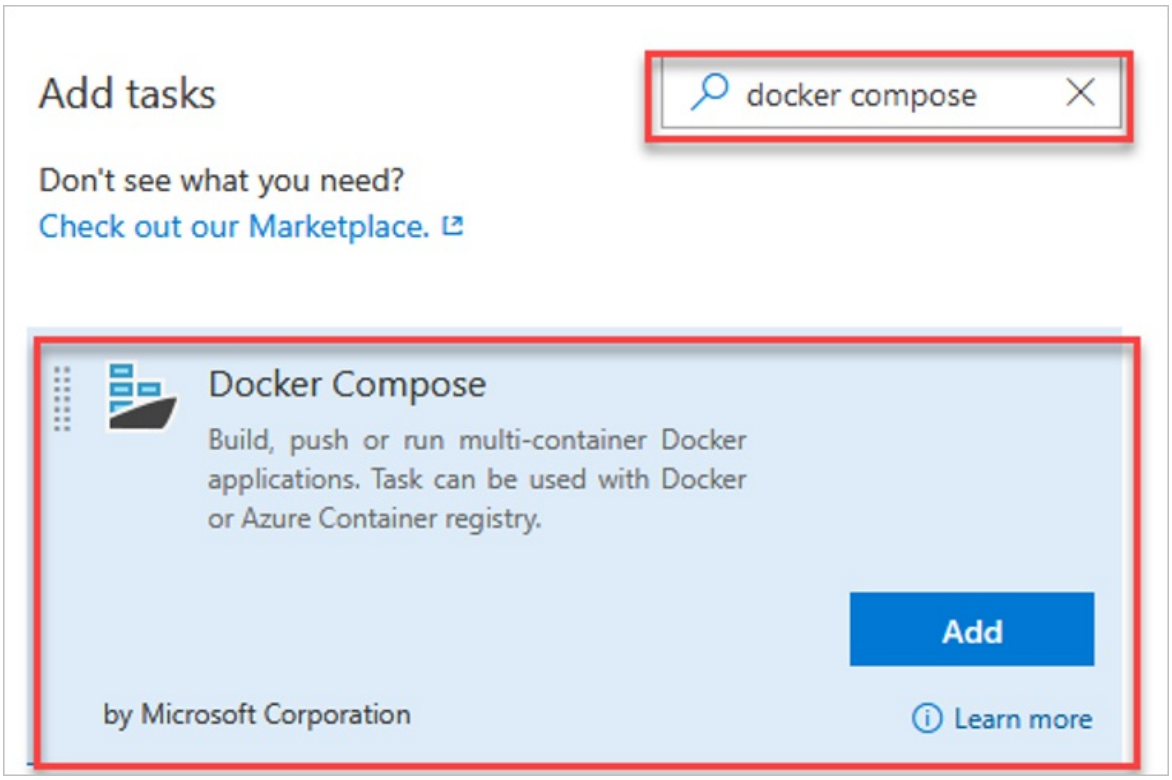


Figure 17: Building the Docker Compose.

8. They repeat and add another **Docker Compose** task. This one pushes the containers to the container registry.

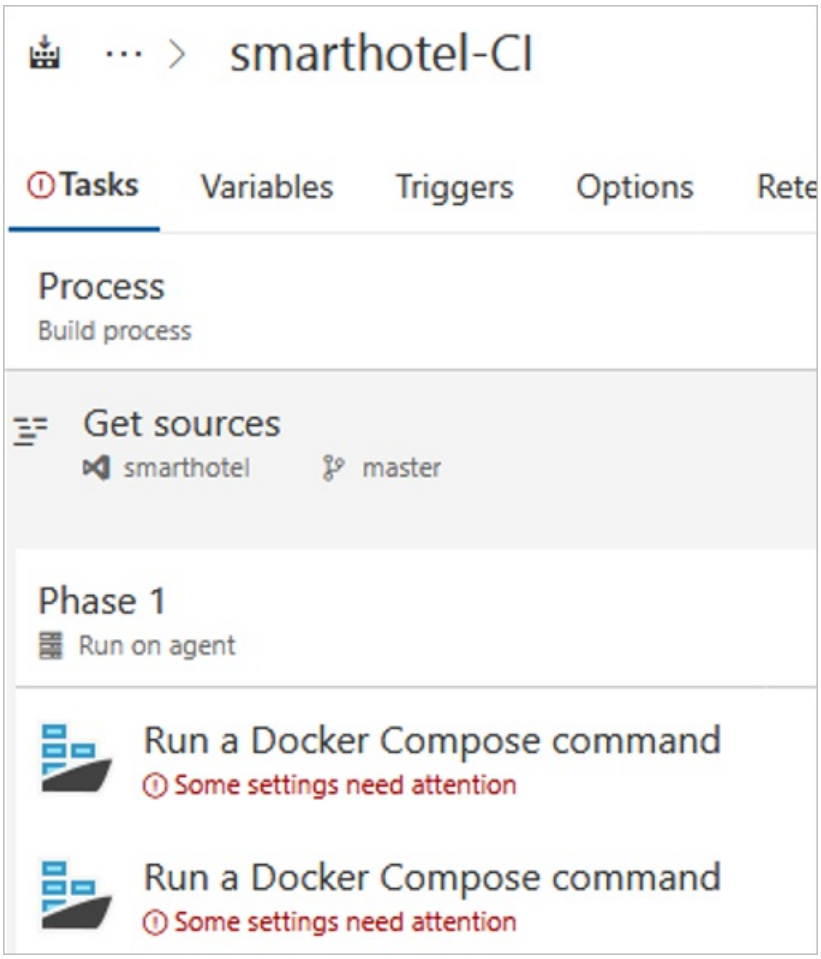


Figure 18: Adding another Docker Compose task.

9. They select the first task to build and configure the build with the Azure subscription, authorization, and Container Registry.

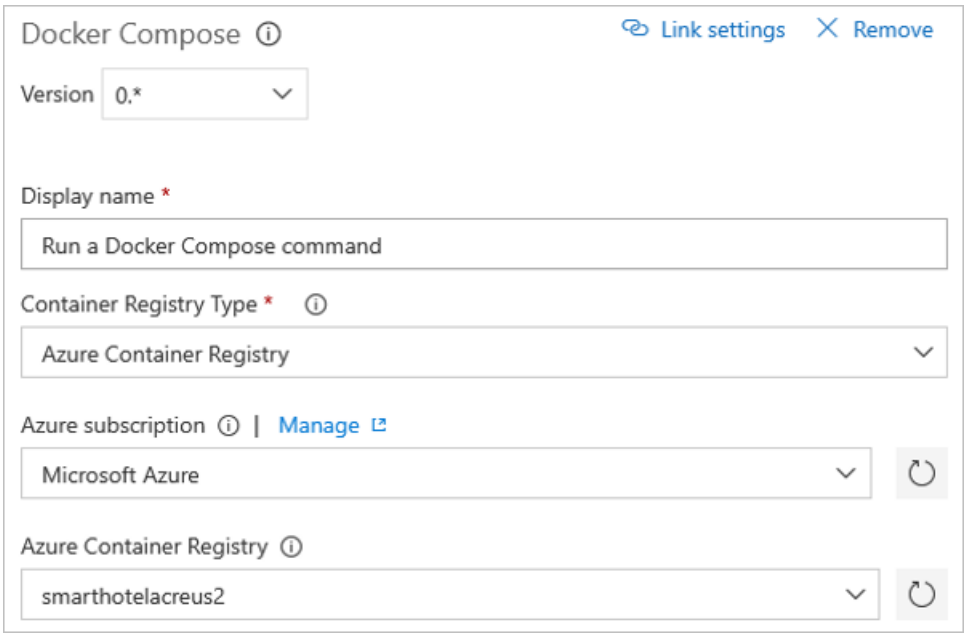


Figure 19: Building and configuring the build.

10. They specify the path of the `docker-compose.yaml` file in the `src` folder of the repo. They choose to build service images and include the latest tag. When the action changes to **Build service images**, the name of the Azure DevOps task changes to **Build services automatically**.

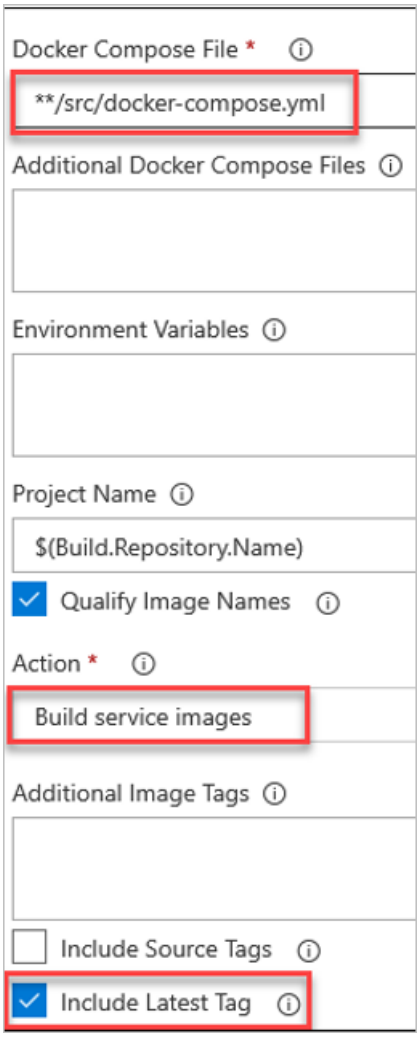


Figure 20: The specifics of the task.

11. Now, they configure the second Docker task (to push). They select the subscription and the container registry (`smarthotelacreus2`).

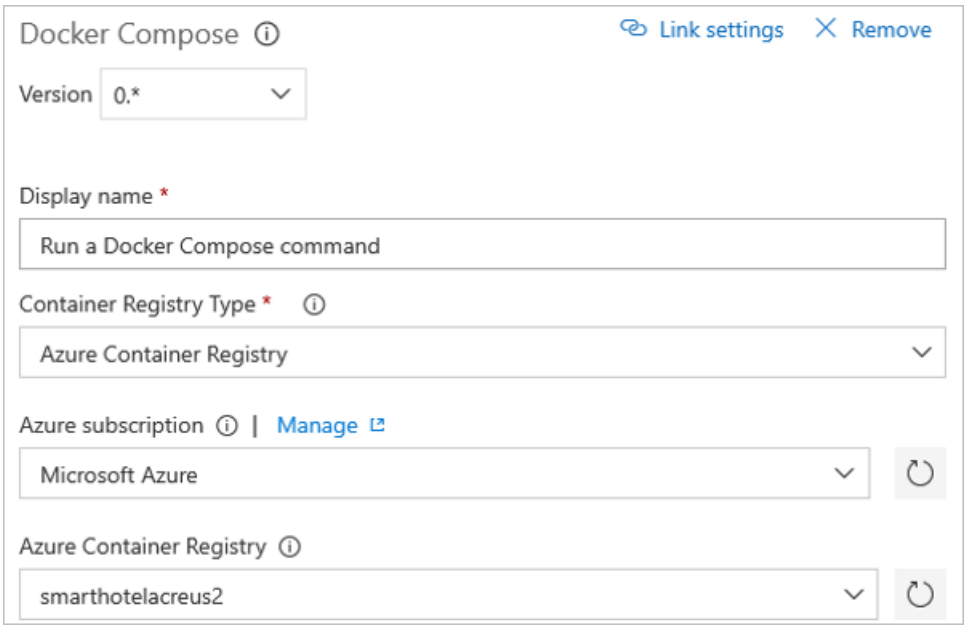


Figure 21: Configuring the second Docker task.

12. They enter the `docker-compose.yaml` file name and select **Push service images**, including the latest tag. When the action changes to **Push service images**, the name of the Azure DevOps task changes to **Push services automatically**.

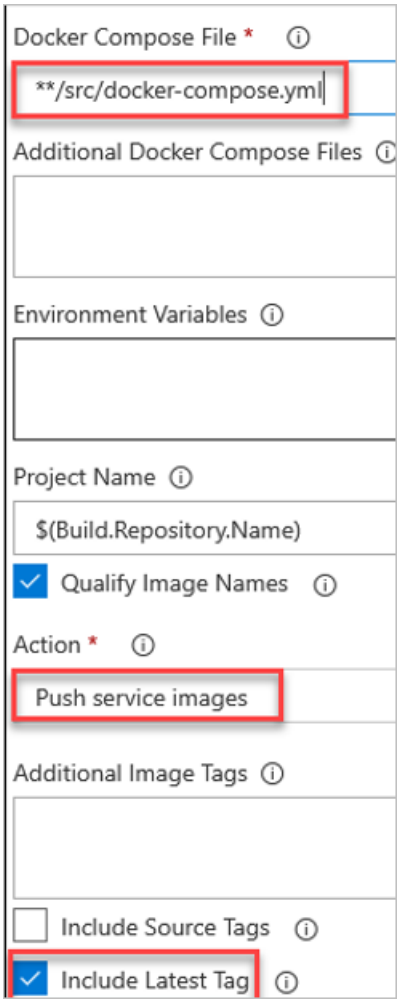


Figure 22: Changing the Azure DevOps task name.

13. With the Azure DevOps tasks configured, Contoso saves the build pipeline and starts the build process.

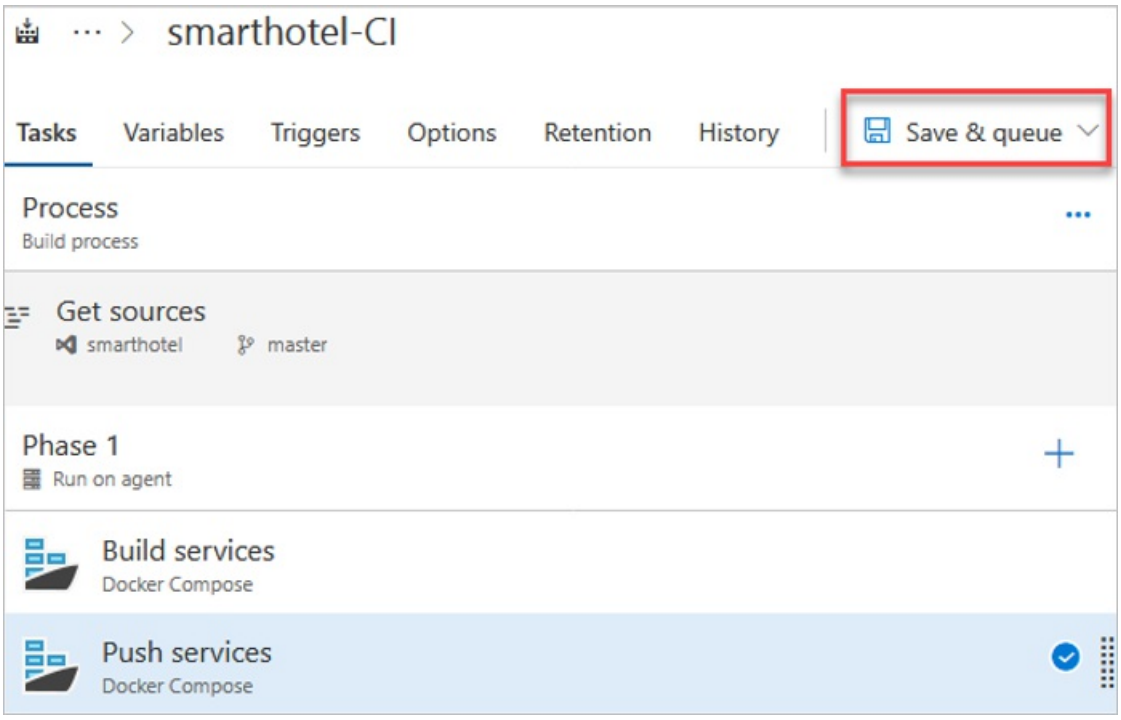
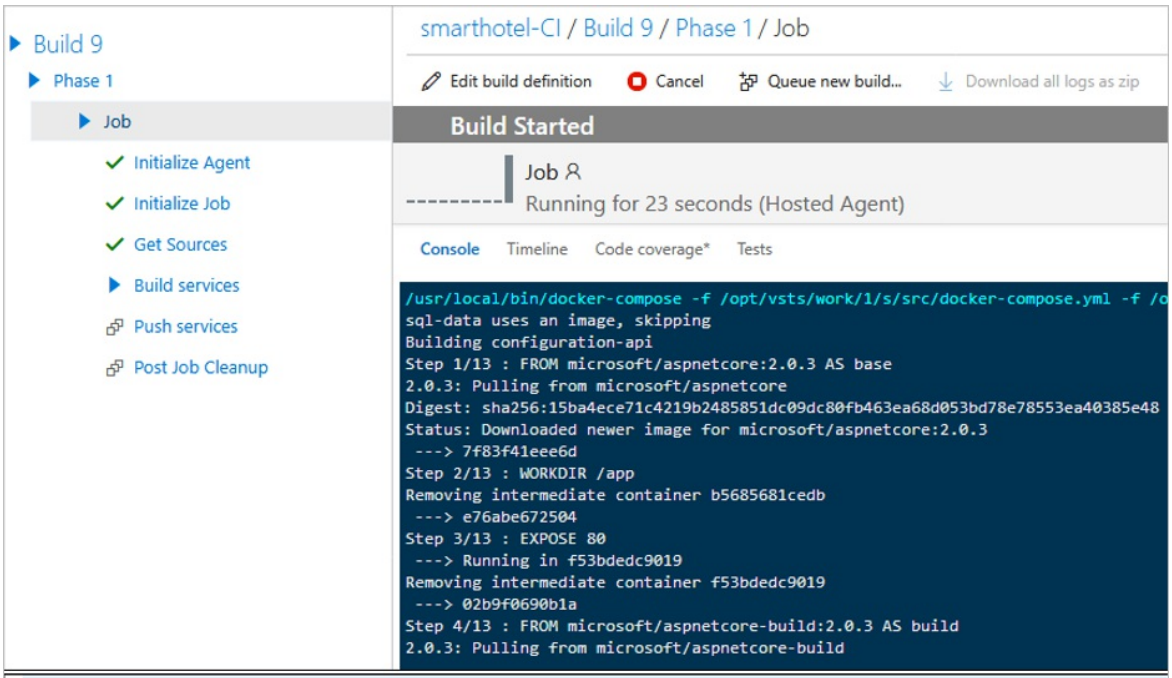


Figure 23: Starting the build process.

14. They select the build job to check progress.



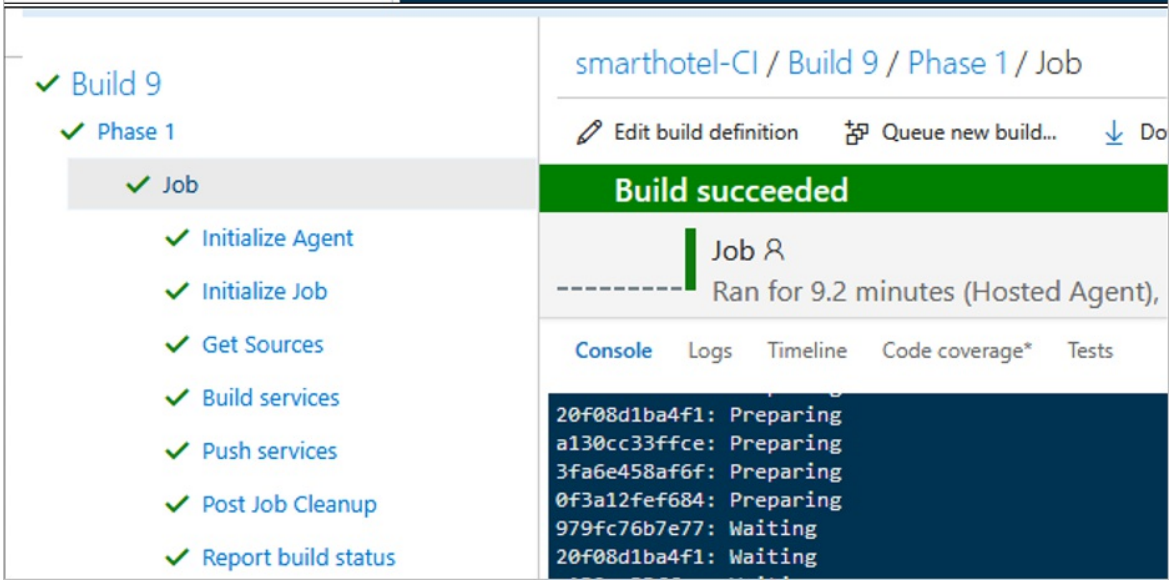


Figure 24: Checking the progress.

15. After the build finishes, the container registry shows the new repos, which are populated with the containers used by the microservices.

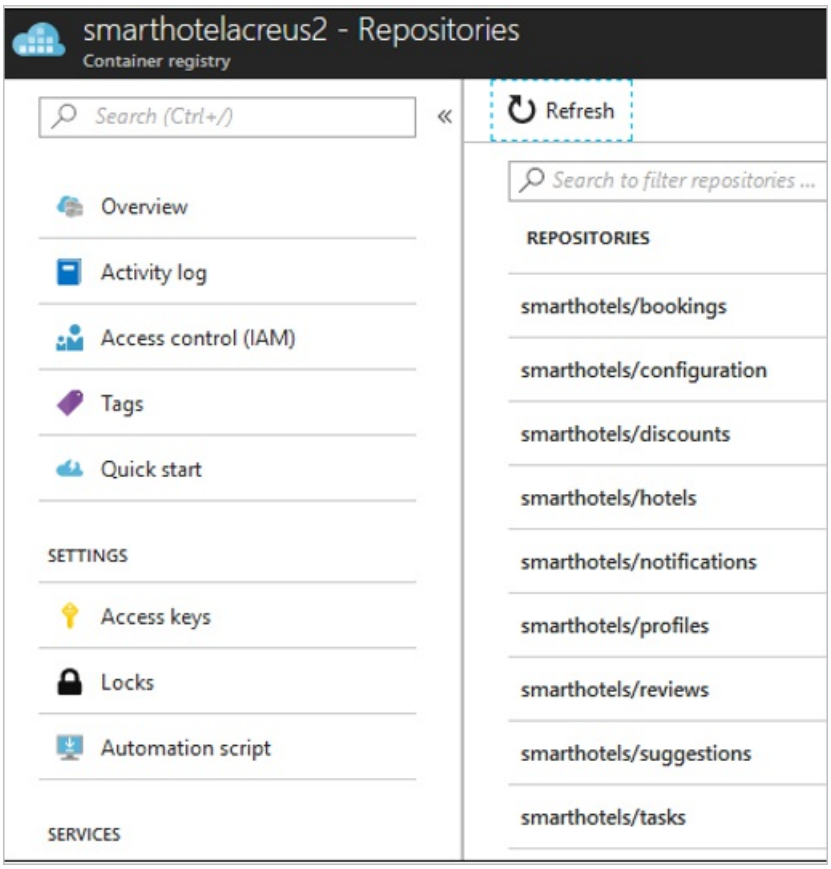


Figure 25: Viewing new repos after the build finishes.

Deploy the back-end infrastructure

With the AKS cluster created and the Docker images built, Contoso admins now deploy the rest of the infrastructure that will be used by back-end microservices.

- Instructions in this section use the `SmartHotel360-Backend` repo.
- In the `/deploy/k8s/arm` folder, there's a single script to create all items.

The admins deploy the infrastructure as follows:

1. They open a developer command prompt and then use the command `az login` for the Azure subscription.
2. They use the `deploy.cmd` file to deploy the Azure resources in the `ContosoRG` resource group and `East US 2` region by typing the following command:

   ```
   .\deploy.cmd azuredeploy ContosoRG -c eastus2
   ```

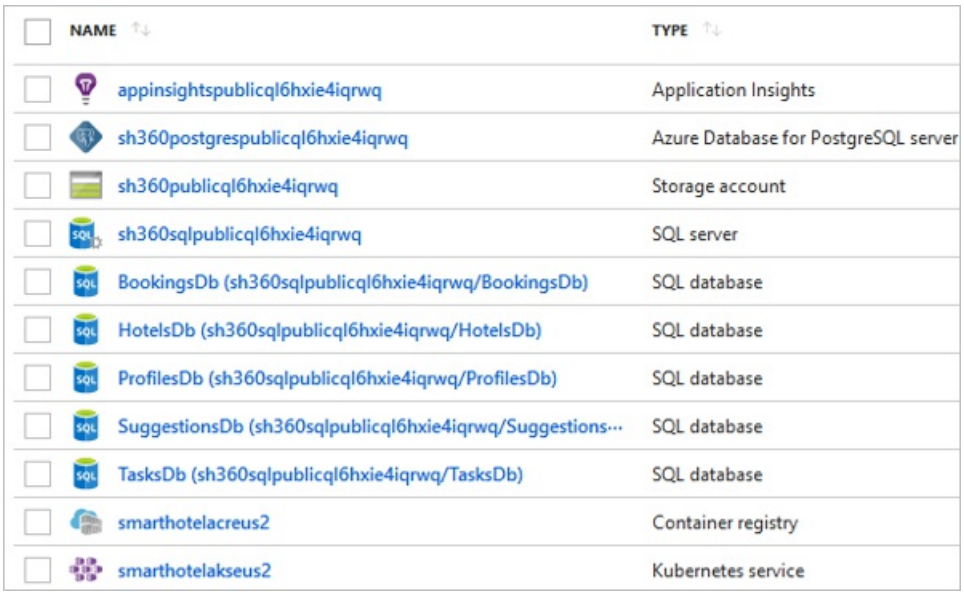


Figure 26: Deploying the back end.

3. In the Azure portal, they capture the connection string for each database for later use.

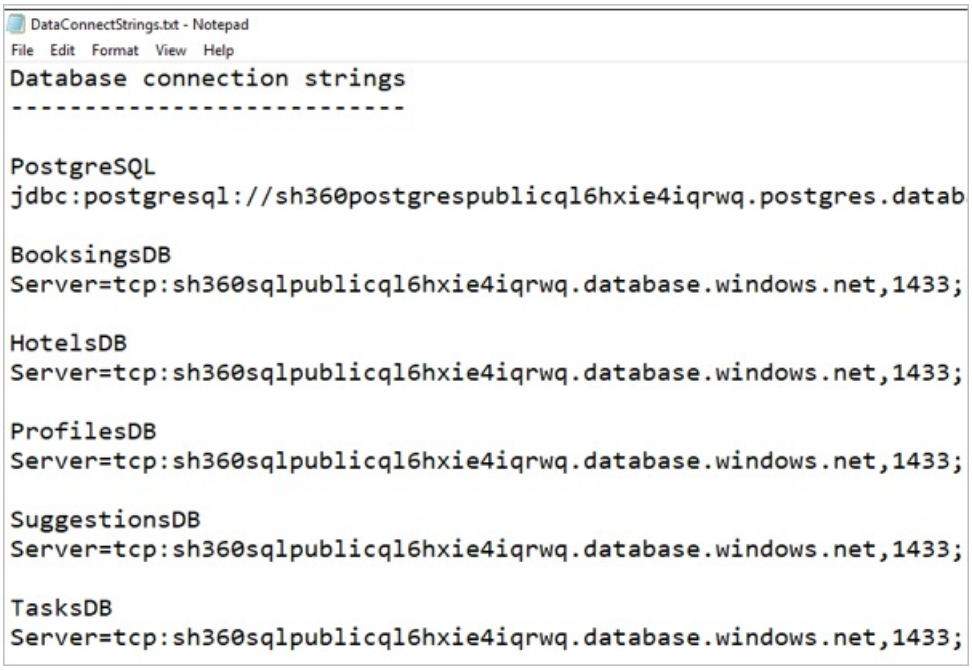


Figure 27: Capturing the connection string for each database.

Create the back-end release pipeline

Now, Contoso admins do the following:

- Deploy the NGINX ingress controller to allow inbound traffic to the services.
- Deploy the microservices to the AKS cluster.
- As a first step, admins update the connection strings to the microservices by using Azure DevOps. They then configure a new Azure DevOps release pipeline to deploy the microservices.
- The instructions in this section use the `SmartHotel360-Backend` repo.
- Some of the configuration settings (for example, Active Directory B2C) aren't covered in this article. For more information about these settings, review the previously mentioned `SmartHotel360-Backend` repo.

The admins create the pipeline:

1. In Visual Studio, they update the `/deploy/k8s/config_local.yml` file with the database connection information that they noted earlier.

2. They open Azure DevOps and, in the `SmartHotel360` project, they select **+ New pipeline** on the **Releases** pane.

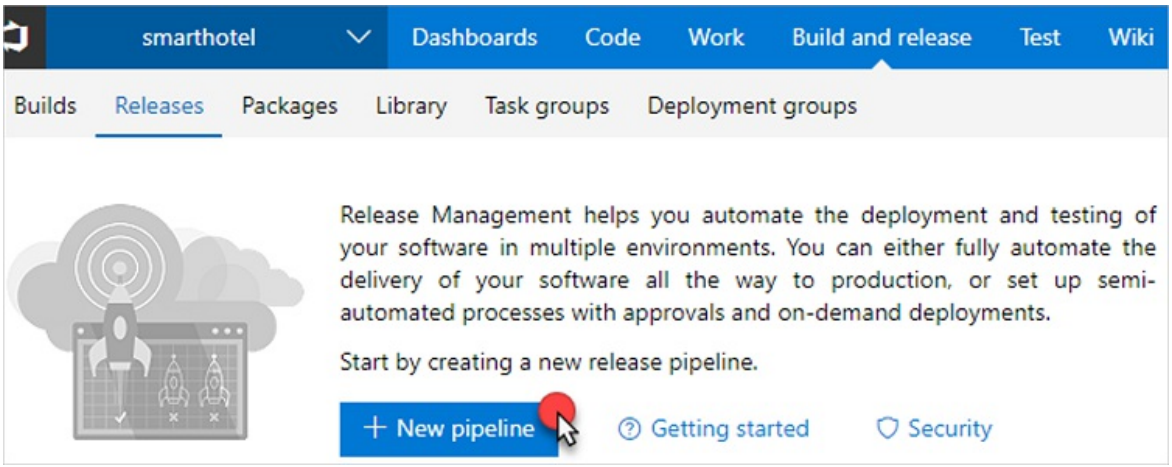


Figure 29: Creating a new pipeline.

3. They select **Empty Job** to start the pipeline without a template.
4. They provide the stage and pipeline names.

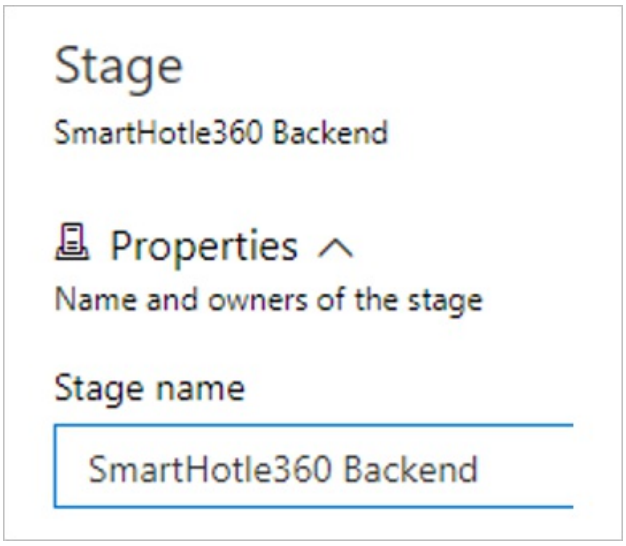


Figure 30: The stage name.

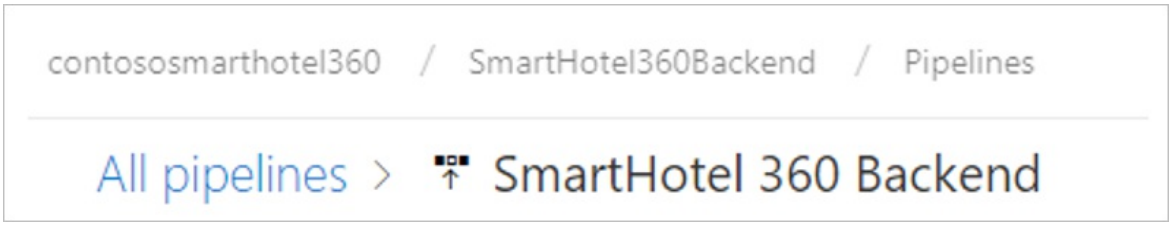


Figure 31: The pipeline name.

5. They add an artifact.

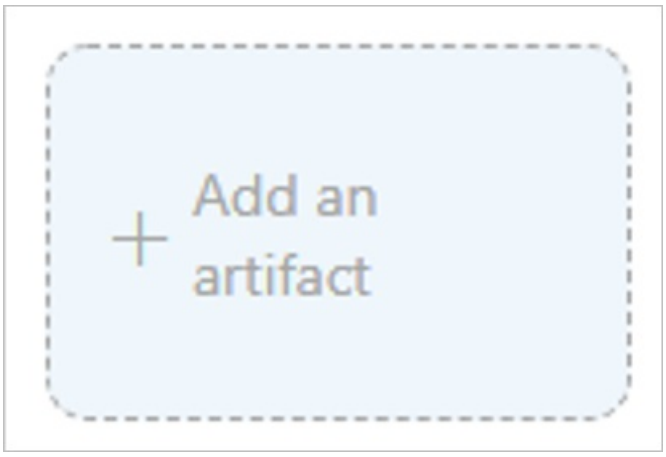


Figure 32: Adding an artifact.

6. They select **Git** as the source type and specify the project, source, and main branch for the SmartHotel360 application.

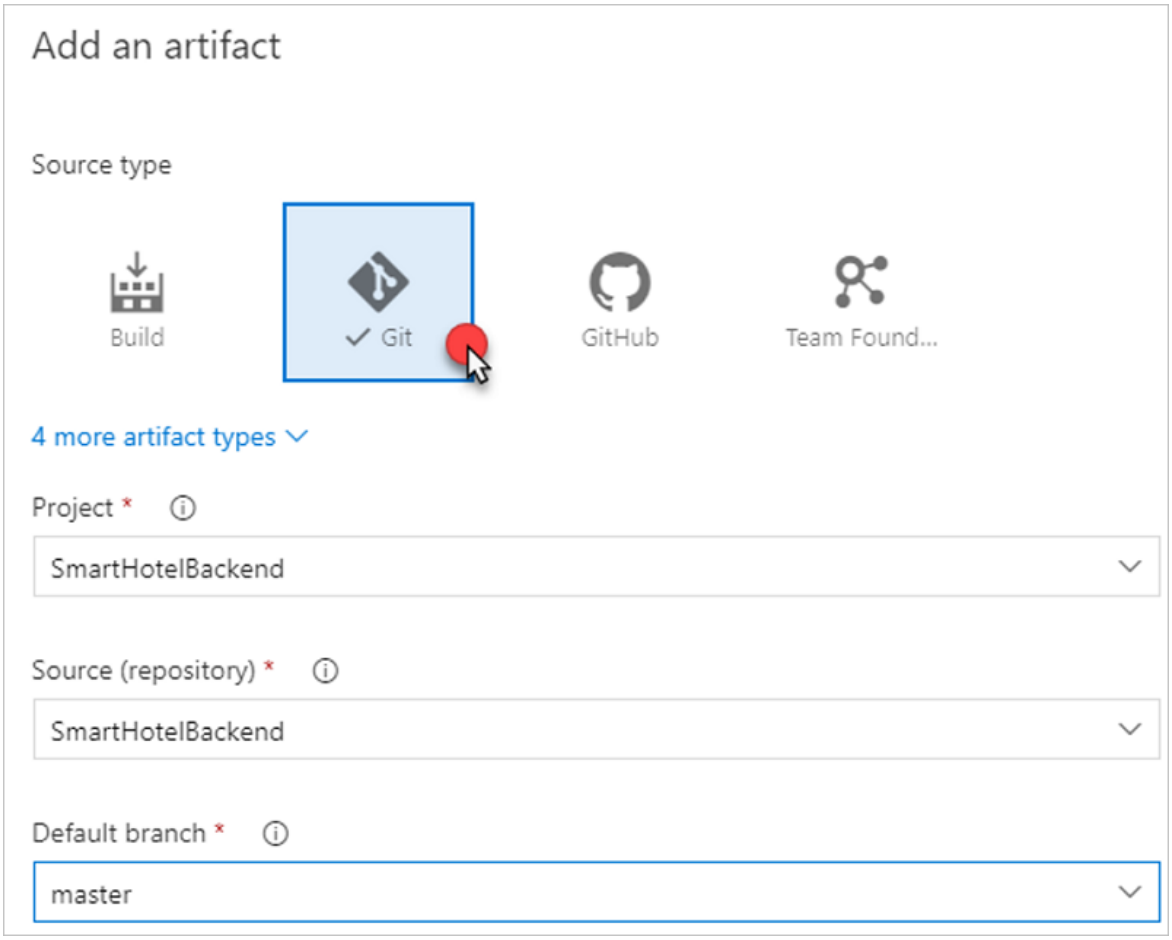


Figure 33: The artifact settings pane.

7. They select the task link.

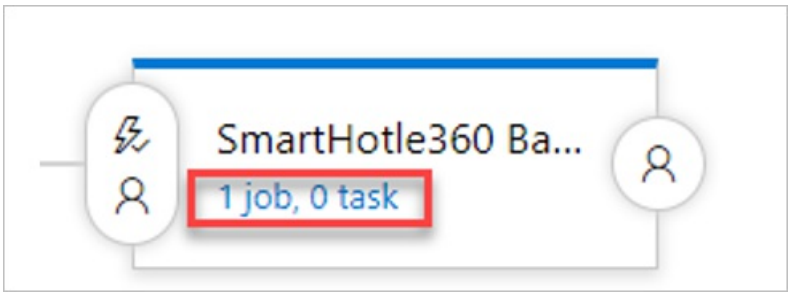


Figure 34: The task link.

8. They add a new Azure PowerShell task so that they can run a PowerShell script in an Azure environment.

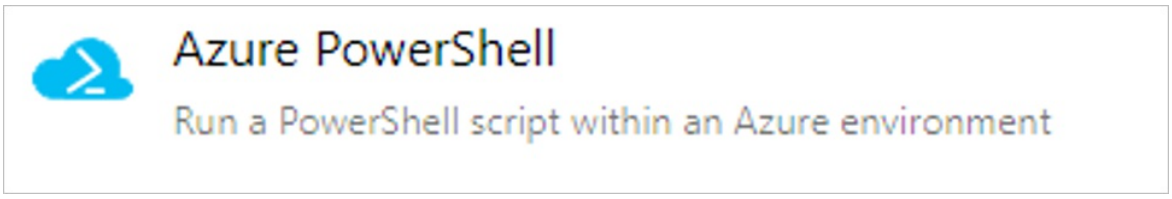


Figure 35: Adding a new PowerShell task.

9. They select the Azure subscription for the task and select the `deploy.ps1` script from the Git repo.

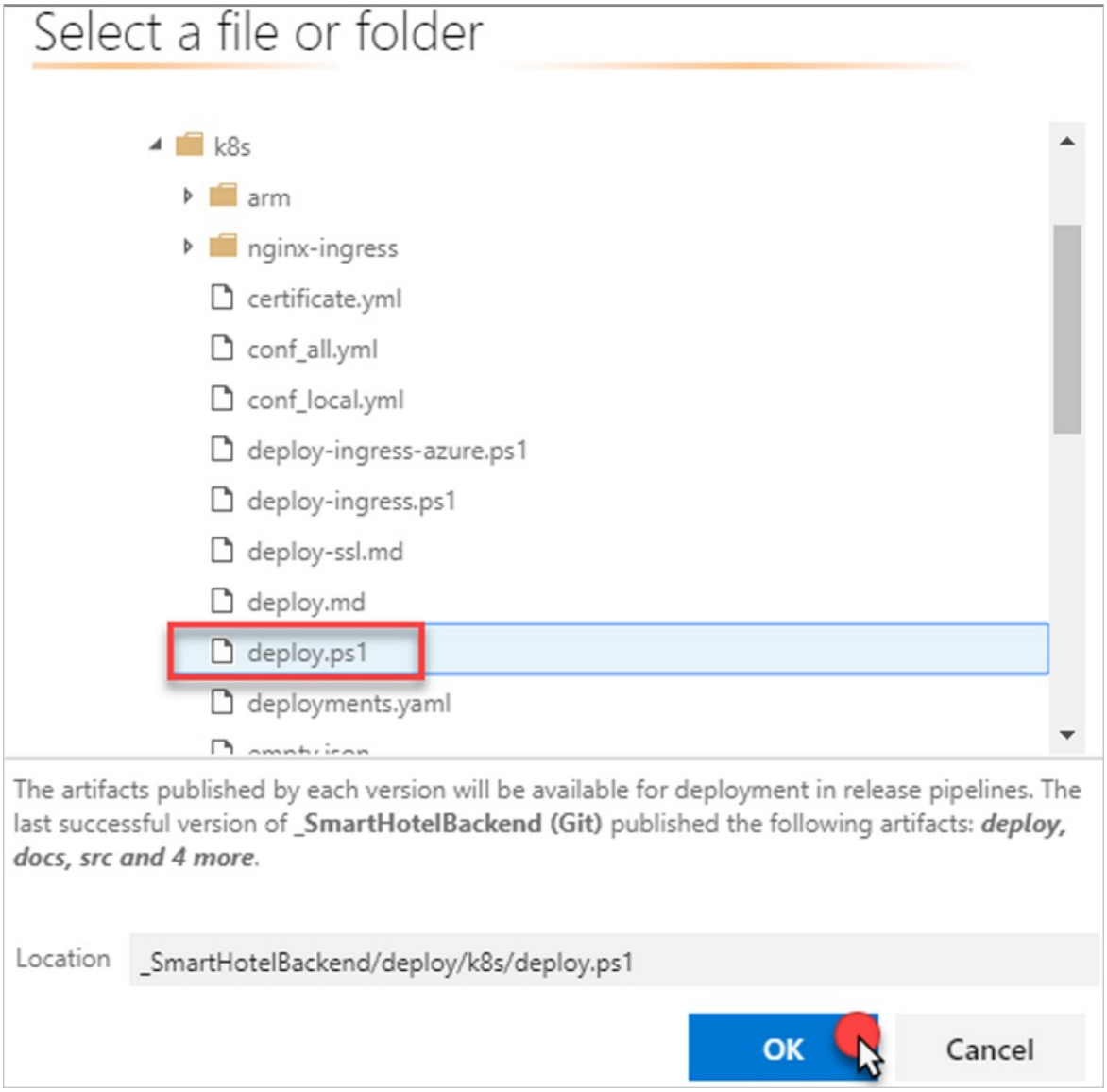


Figure 36: Running the script.

10. They add arguments to the script. The script will delete all cluster content (except **ingress** and **ingress controller**), and deploy the microservices.

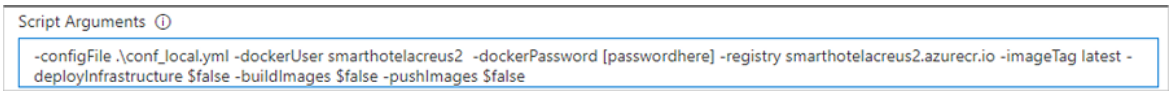


Figure 37: Adding arguments to the script.

11. They set the preferred Azure PowerShell version to the latest version, and save the pipeline.

12. They go back to the **Create a new release** pane and manually create a new release.

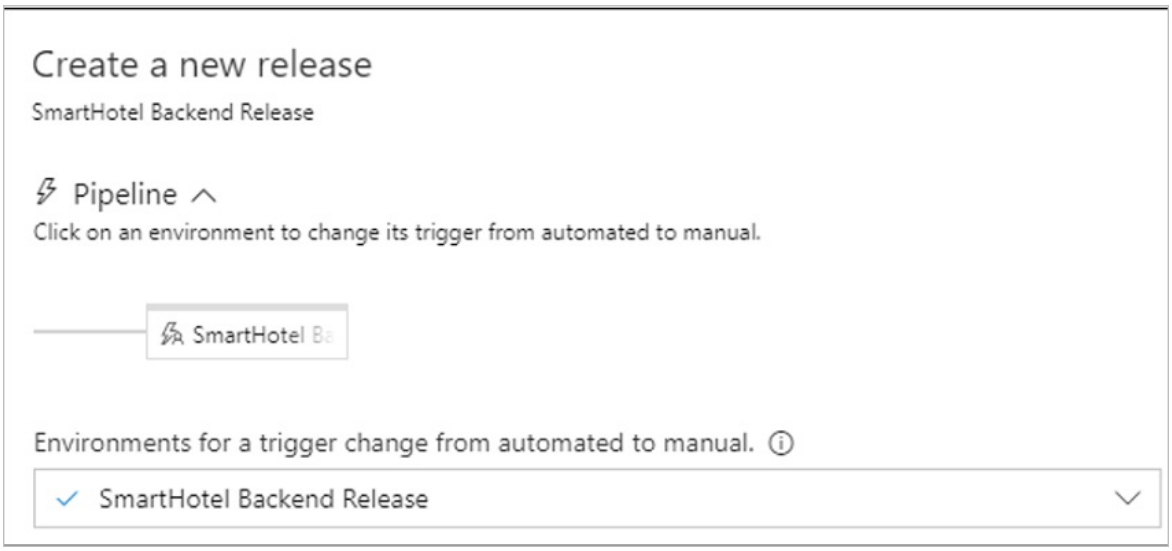


Figure 38: Manually creating a new release.

13. After they create the release, they select it and, under **Actions**, they select **Deploy**.

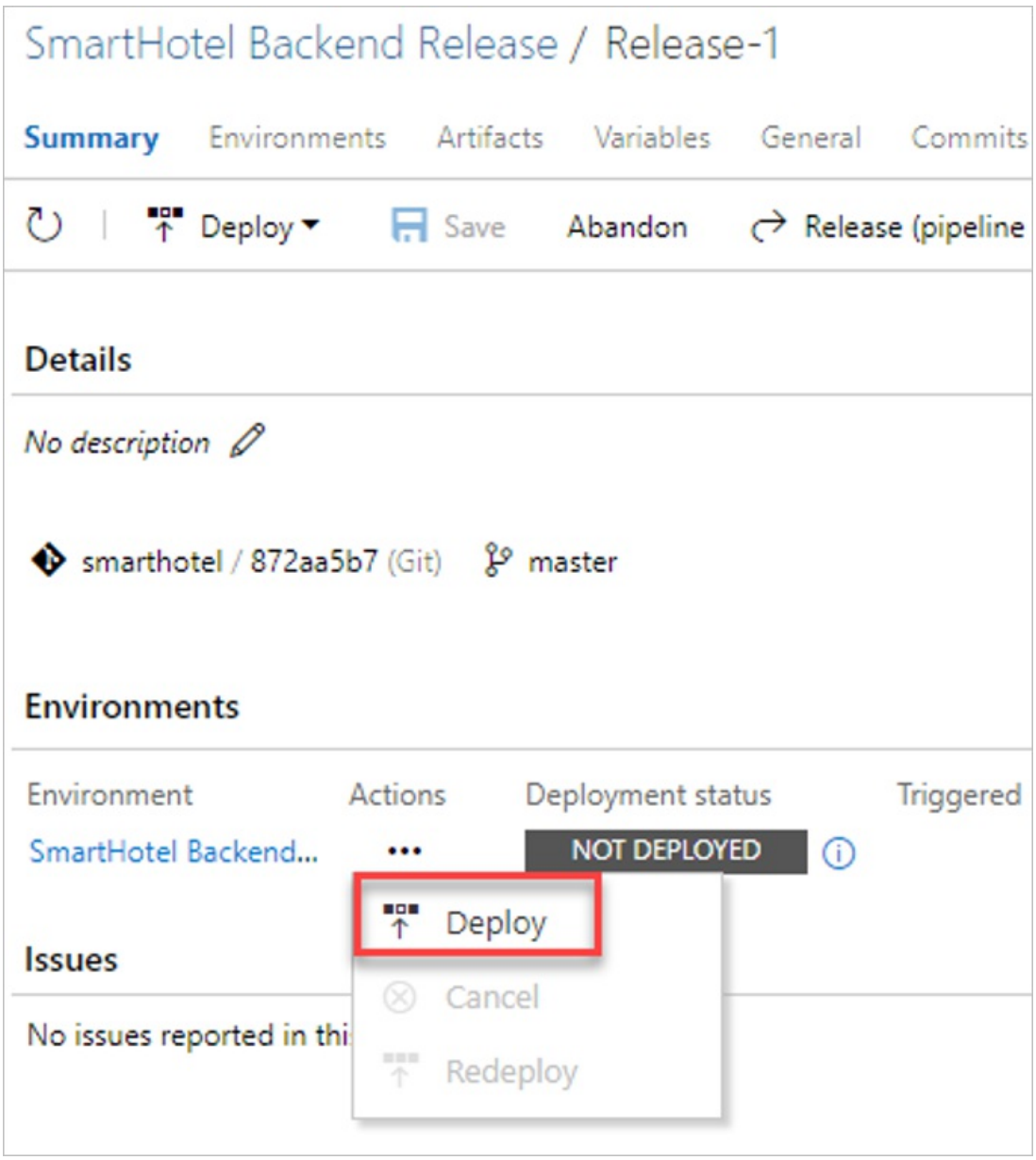


Figure 39: Deploying a release.

14. When the deployment is complete, they run the following command to check the status of services, using the Azure Cloud Shell: `kubectl get services`.

Step 3: Provision front-end services

Contoso admins need to deploy the infrastructure that will be used by the front-end applications. They create:

- A Blob Storage container to store the pet images.
- An Azure Cosmos DB database to store documents containing pet information.
- The Computer Vision API for the website.

Instructions for this section use the `SmartHotel360-Website` repo.

Create Blob Storage containers

1. In the Azure portal, the admins open the storage account that was created, and then select **Blobs**.
2. They create a new container named `Pets` with the public-access level set for the container. Users will upload their pet photos to this container.

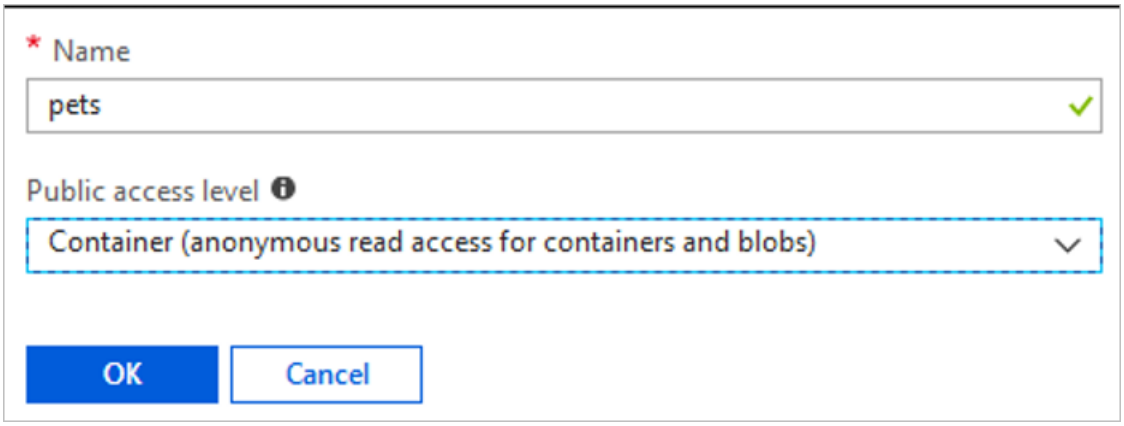


Figure 40: Creating a new container.

3. They create a second new container named `settings`. A file with all the front-end app settings will be placed in this container.

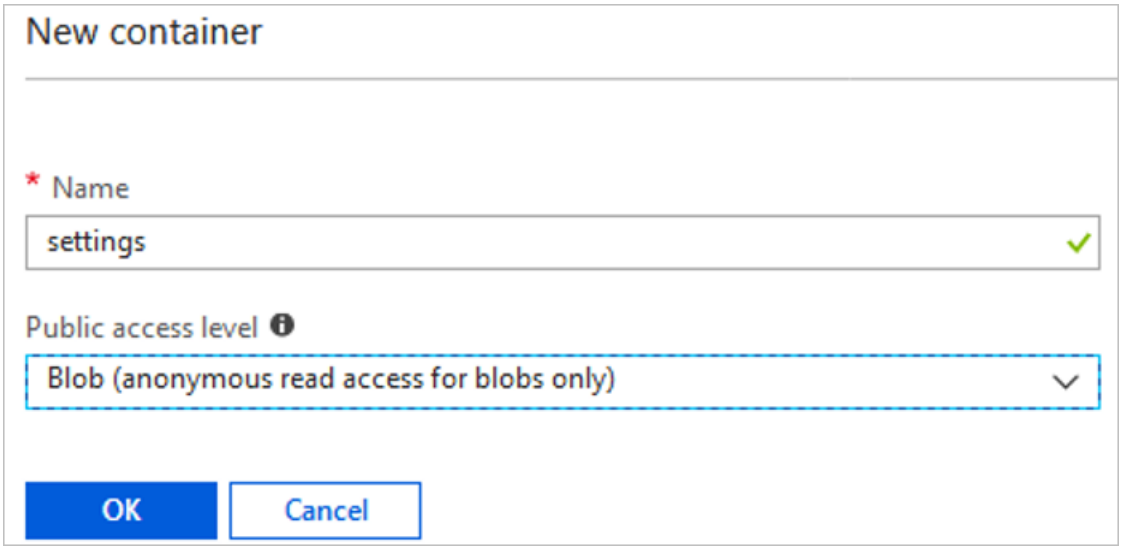


Figure 41: Creating a second container.

4. They capture the access details for the storage account in a text file for future reference.

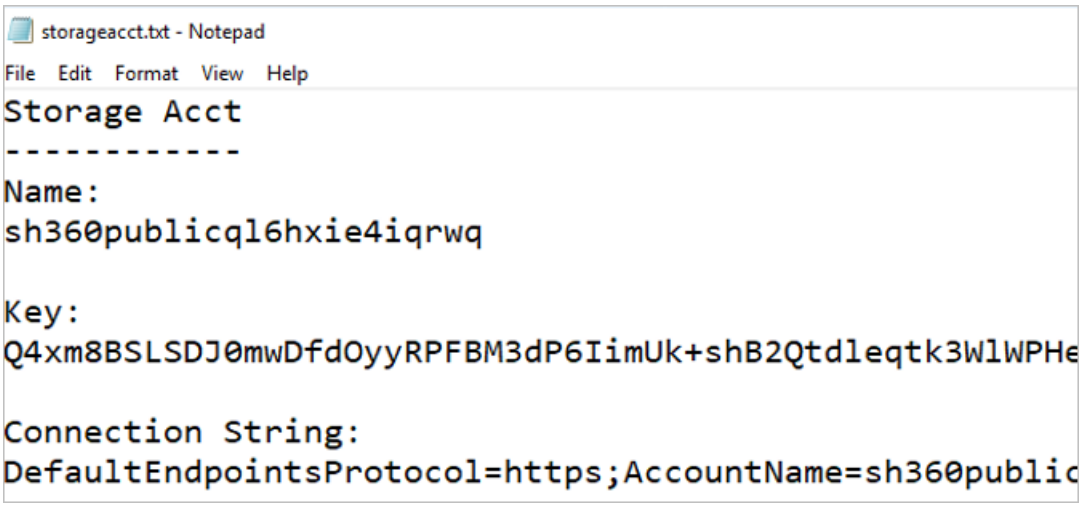


Figure 42: A text file capturing access details.

Provision an Azure Cosmos DB database

Contoso admins provision an Azure Cosmos DB database to be used for pet information.

1. They create an **Azure Cosmos DB** database in Azure Marketplace.

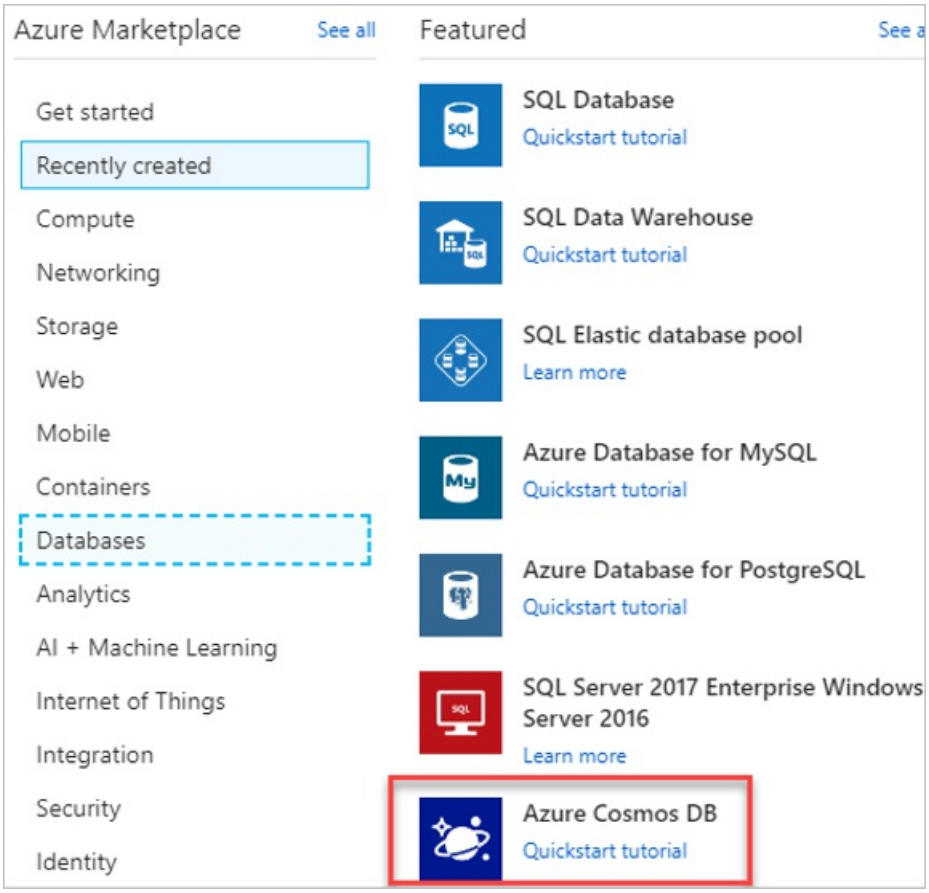


Figure 43: Creating an Azure Cosmos DB database.

2. They specify a name `contososmarthotel`, select the SQL API, and place it in the production resource group `ContosoRG` in the main region `East US 2`.

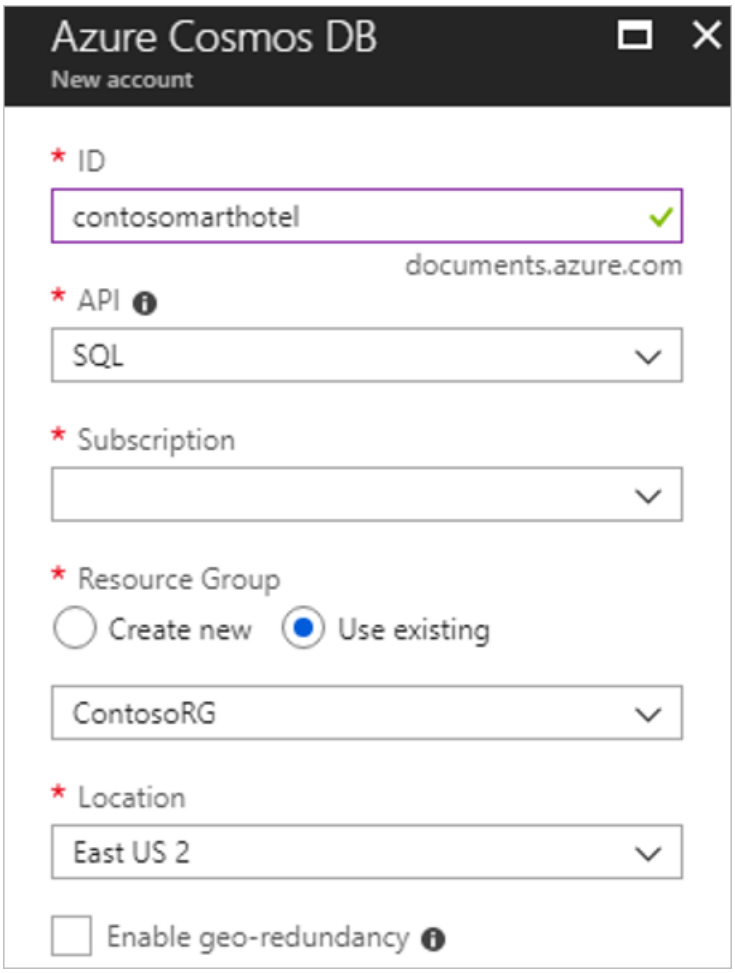


Figure 44: Naming an Azure Cosmos DB database.

3. They add a new collection to the database, with default capacity and throughput.

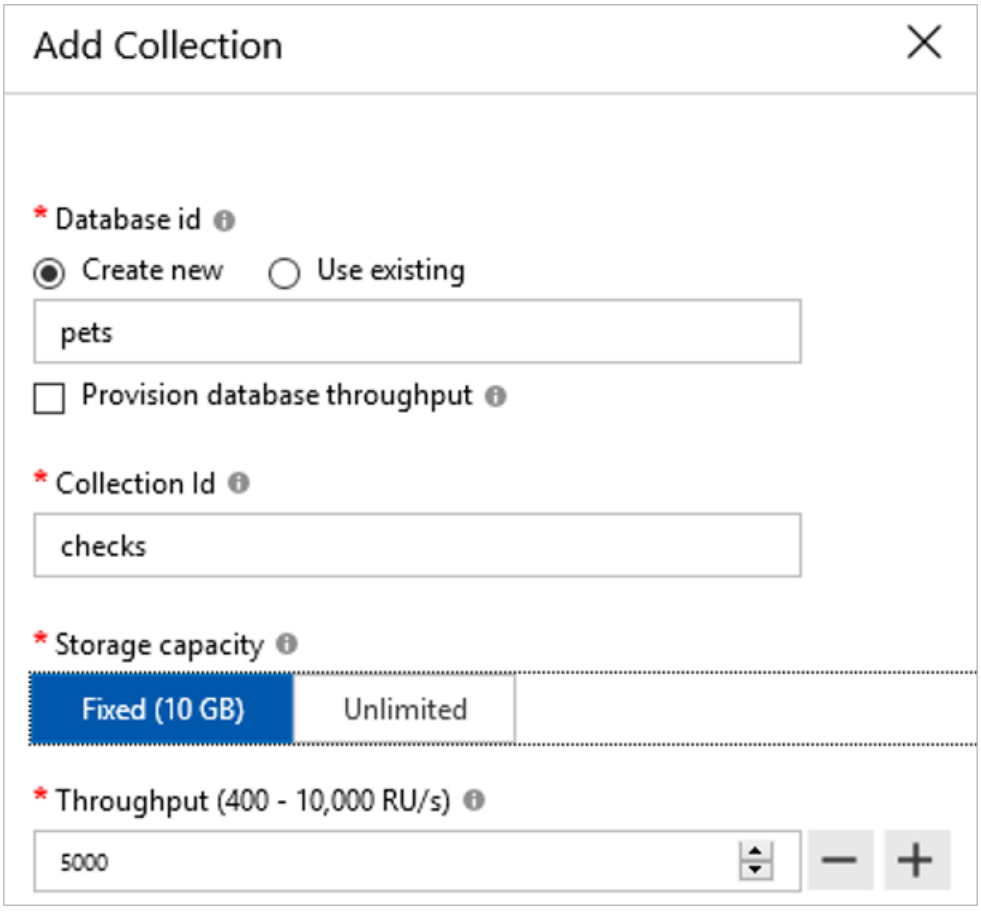


Figure 45: Adding a new collection to the database.

4. They note the connection information for the database for future reference.

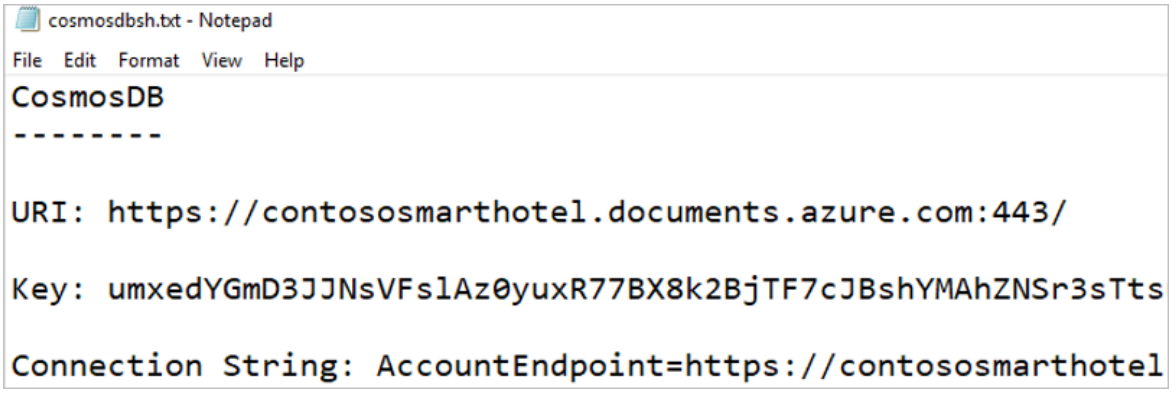


Figure 46: The connection information for the database.

Provision the Computer Vision API

Contoso admins provision the Computer Vision API. The API will be called by the function, to evaluate pictures that are uploaded by users.

1. The admins create a **Computer Vision** instance in Azure Marketplace.

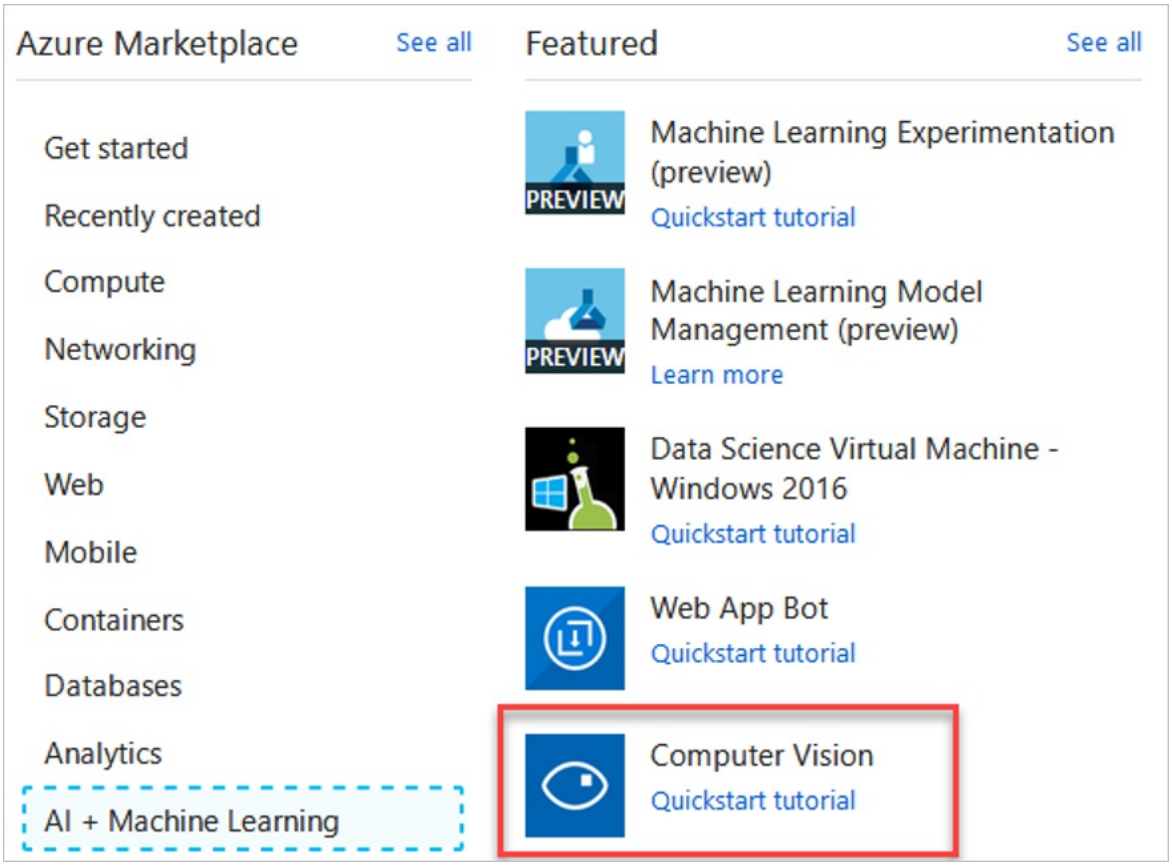


Figure 47: A new instance in Azure Marketplace.

2. They provision the API (`smarthotelpets`) in the production resource group `ContosoRG` , in the main region (`East US 2`).

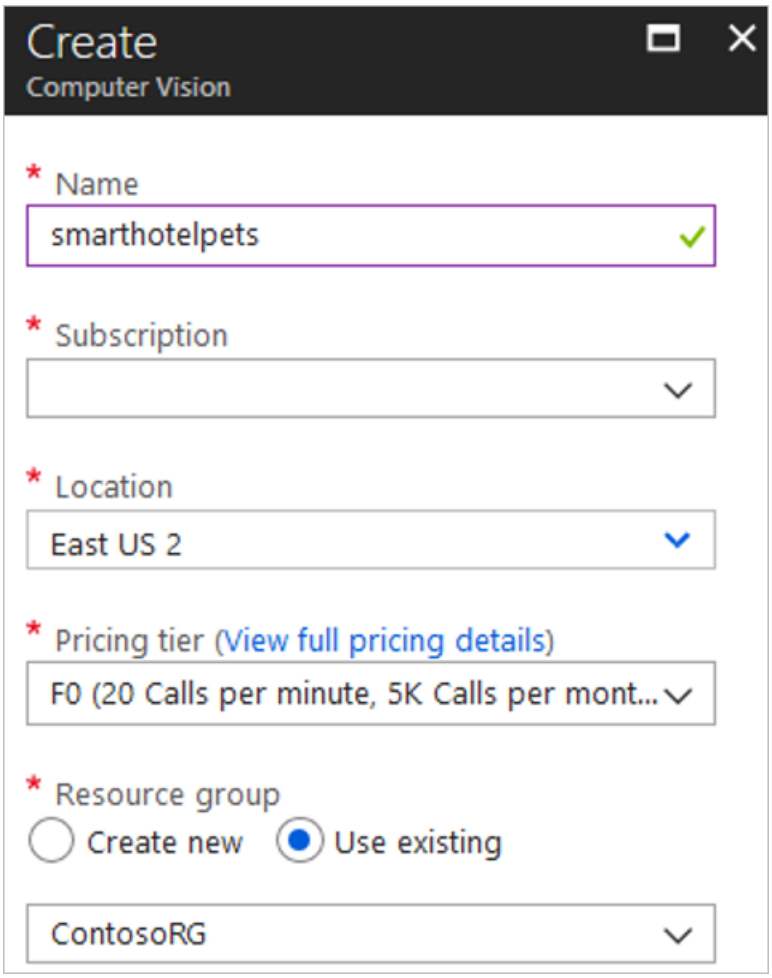


Figure 48: Provisioning an API in a production resource group.

3. They save the connection settings for the API to a text file for later reference.

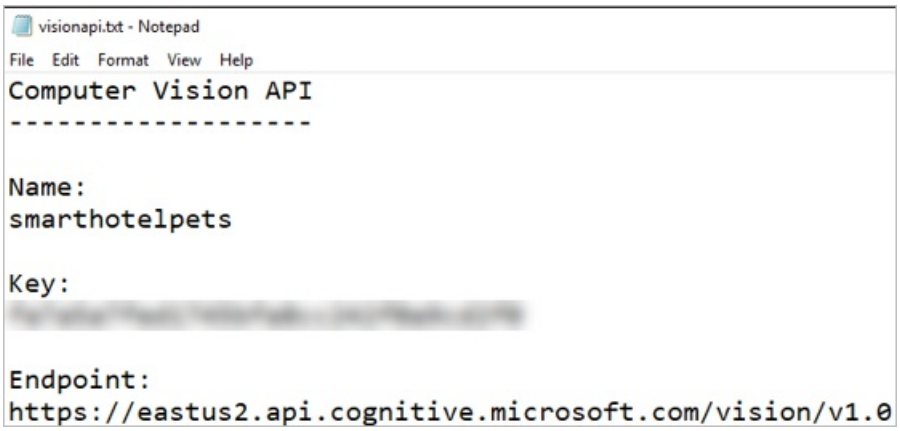


Figure 49: Saving the API connection settings.

Provision the Azure web app

Contoso admins provision the web app by using the Azure portal.

1. They select **Web App** in the portal.

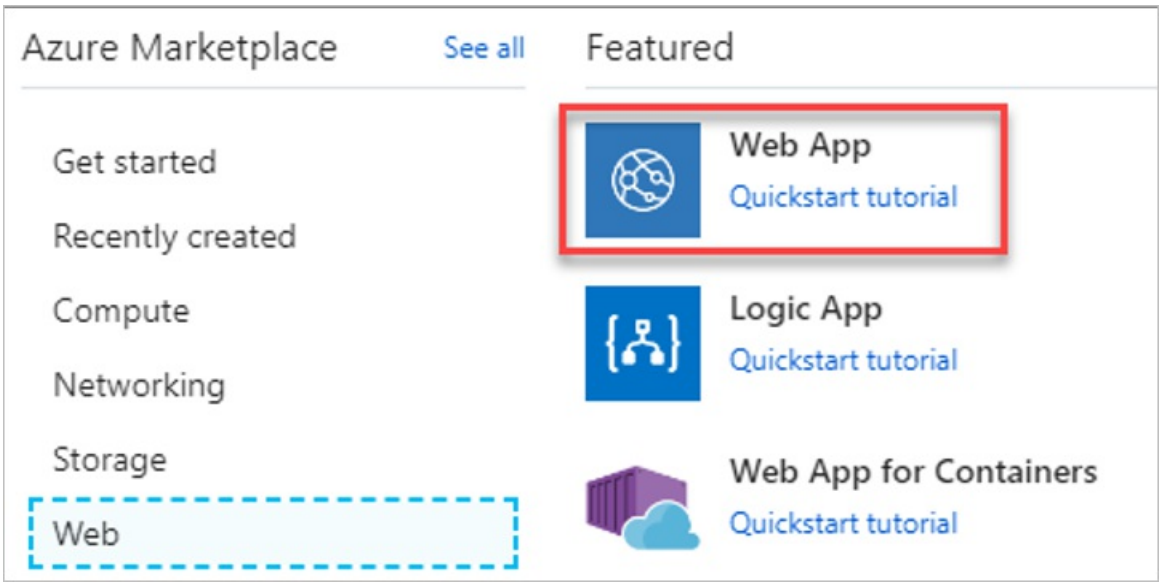


Figure 50: Selecting the web app.

2. They provide a web app name (`smarthotelcontoso`), run it on Windows, and place it in the production resource group `ContosoRG`. They create a new Application Insights instance for application monitoring.

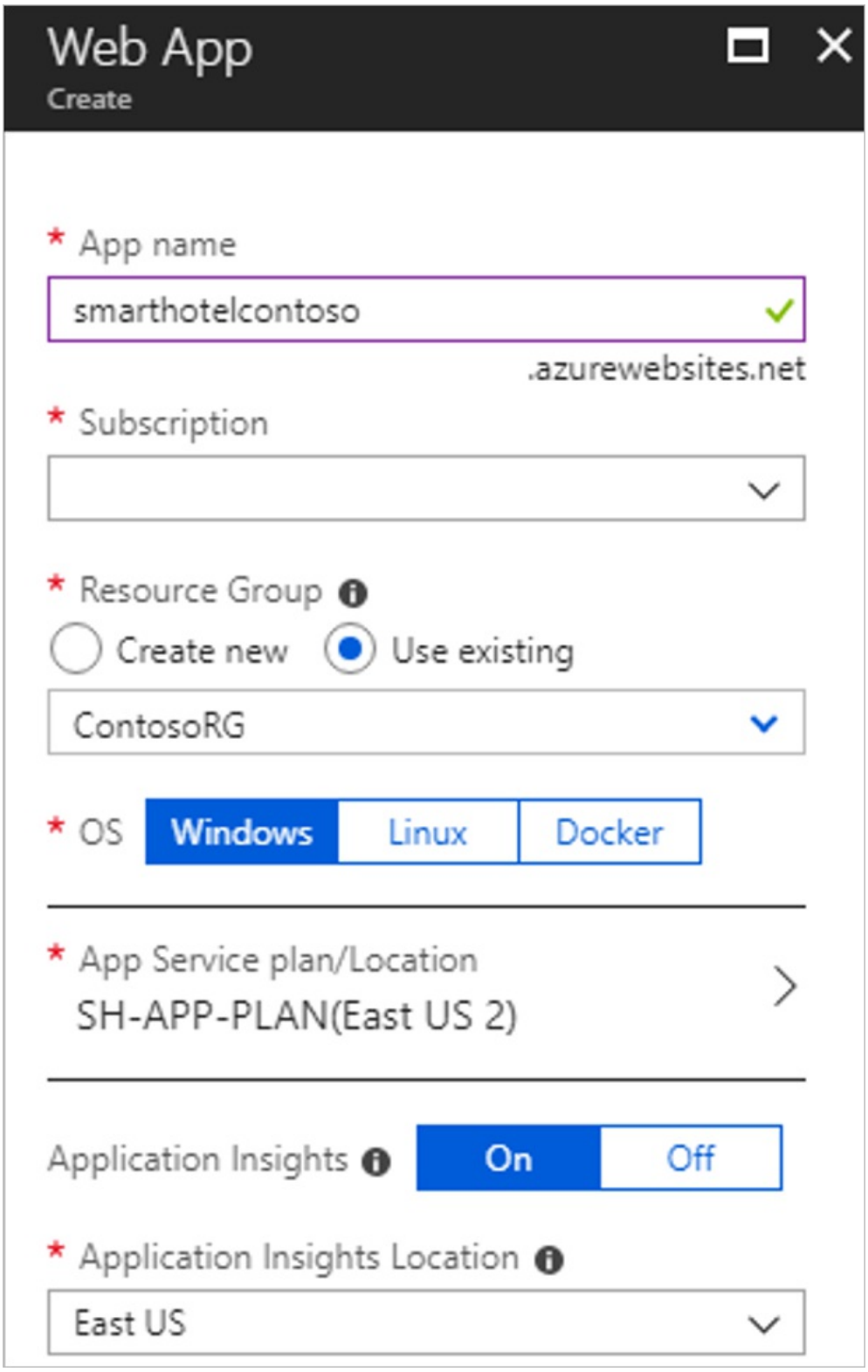


Figure 51: The web app name.

3. After they're done, the admins browse to the address of the application to check on whether it has been created successfully.

4. In the Azure portal, they create a staging slot for the code. The pipeline will be deployed to this slot. This approach ensures that code isn't put into production until the admins perform a release.

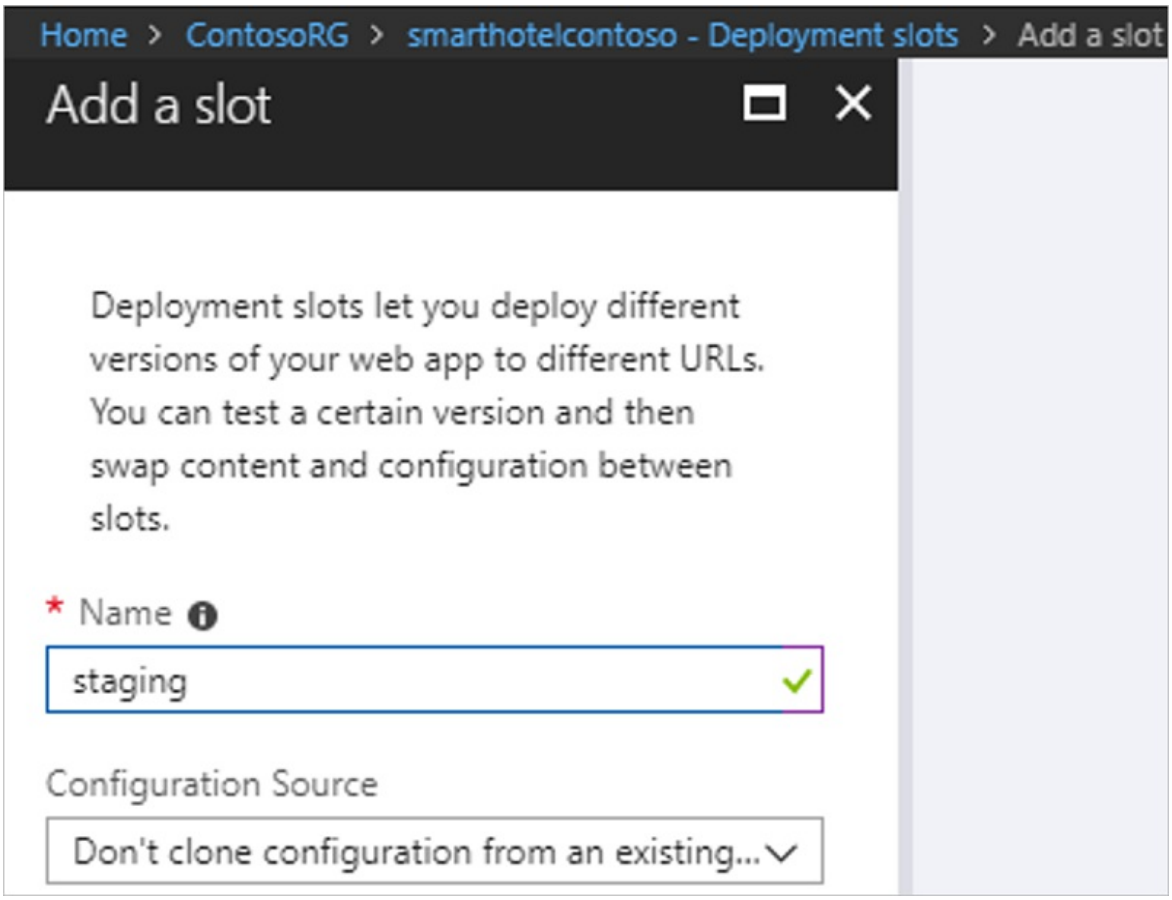


Figure 52: Adding a web app staging slot.

Provision the function app

In the Azure portal, Contoso admins provision the function app.

1. They select **Function App**.

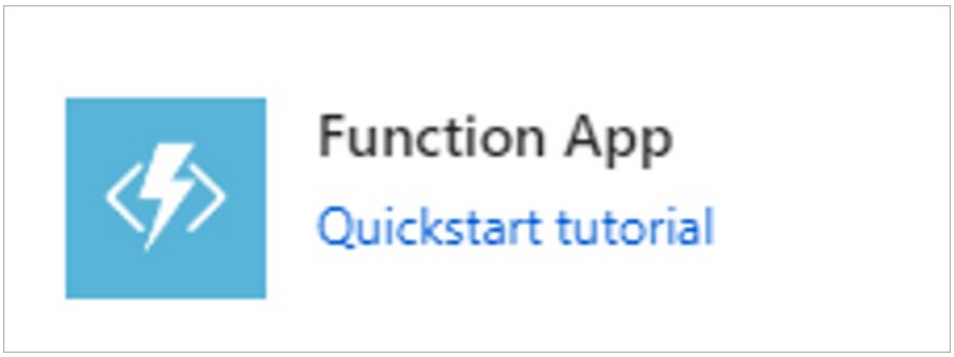


Figure 53: Selecting the function app.

2. They provide a name for the function app (`smarthotelpetchecker`). They place the function app in the production resource group (`ContosoRG`). They set the hosting place to **Consumption Plan**, and place the function app in the `East US 2` region. A new storage account is created along with an Application Insights instance for monitoring.

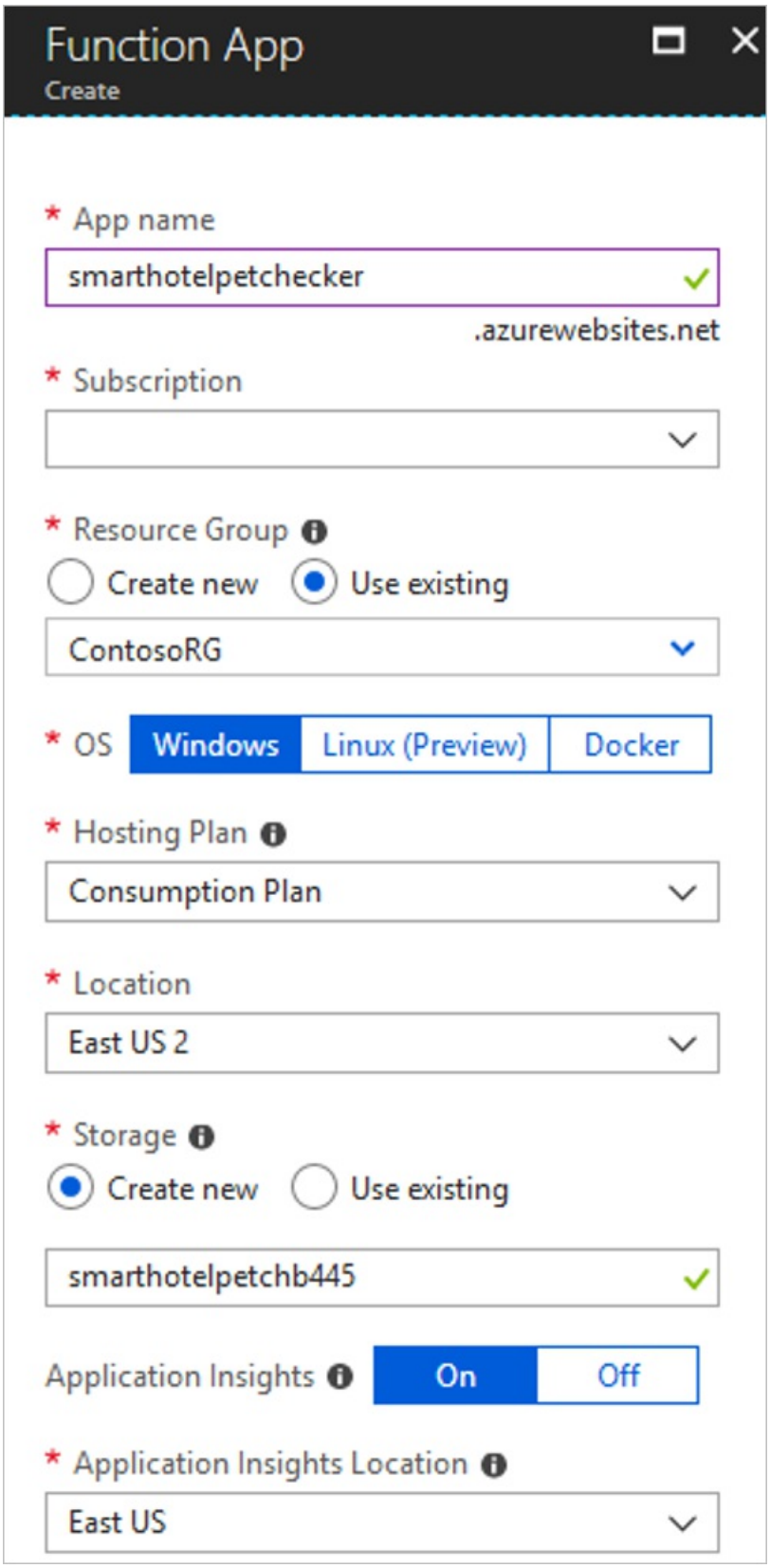


Figure 54: Function app settings.

3. After they've deployed the function app, the admins browse to its address to verify that it has been created successfully.

Step 4: Set up the front-end pipeline

Contoso admins create two different projects for the front-end site.

1. In Azure DevOps, they create a project named `SmartHotelFrontEnd`.

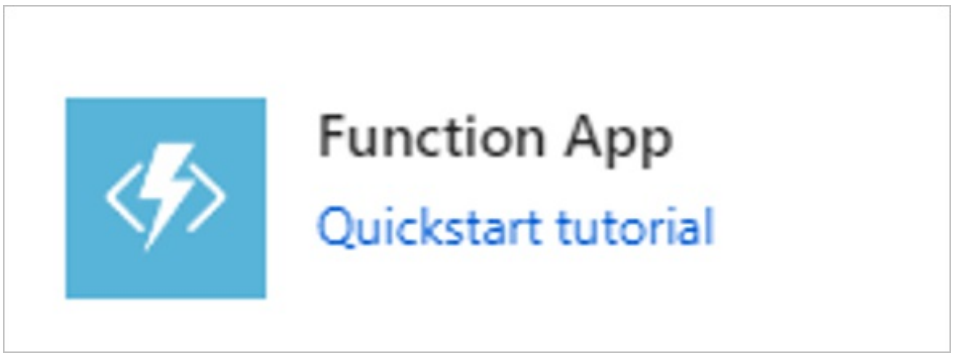


Figure 55: Creating a front-end project.

2. They import the `SmartHotel360-Website` Git repository into the new project.
3. For the function app, they create another Azure DevOps project (`SmartHotelPetChecker`) and import the PetChecker Git repository into this project.

Configure the web app

Now Contoso admins configure the web app to use Contoso resources.

1. The admins connect to the Azure DevOps project and clone the repository locally to the development machine.
2. In Visual Studio, they open the folder to show all the files in the repo.

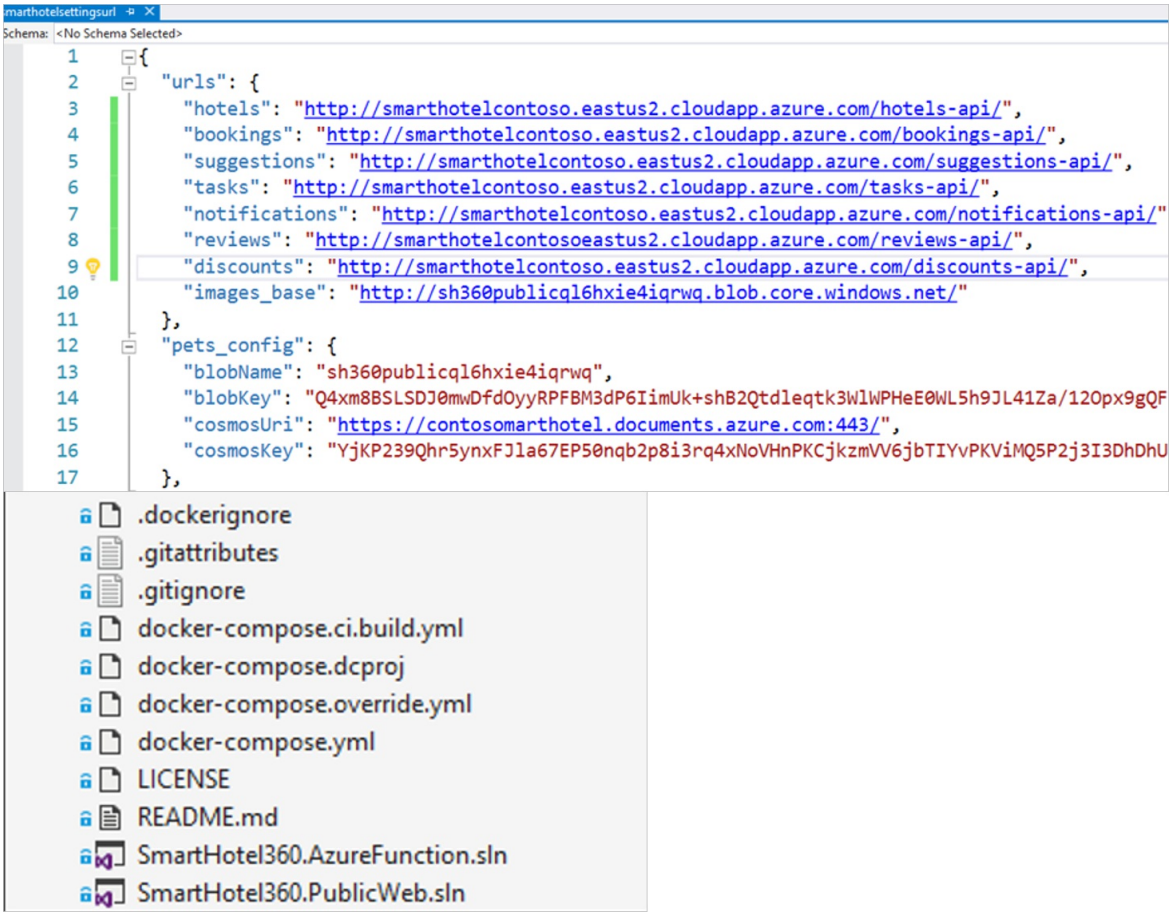


Figure 56: Viewing all files in the repo.

3. They update the configuration changes as required.
   - When the web app starts up, it looks for the `SettingsUrl` app setting.
   - This variable must contain a URL that points to a configuration file.
   - By default, the setting that's used is a public endpoint.
4. They update the `/config-sample.json/sample.json` file.

- This is the configuration file for the web when it uses the public endpoint.
- They edit the `urls` and `pets_config` sections with the values for the AKS API endpoints, storage accounts, and Azure Cosmos DB database.
- The URLs should match the DNS name of the new web app that Contoso will create.
- For Contoso, this is `smarthotelcontoso.eastus2.cloudapp.azure.com`.

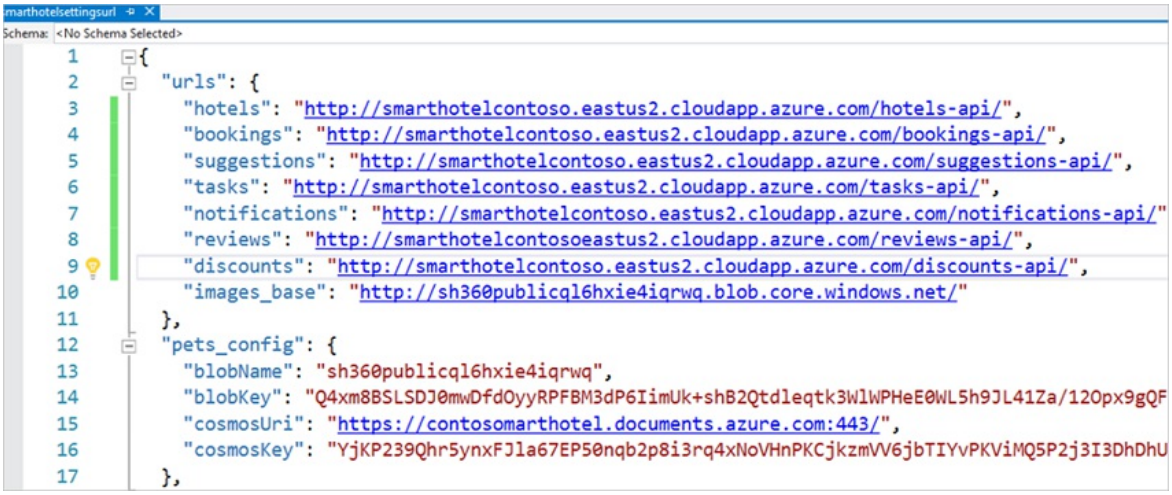


Figure 57: The .json settings.

5. After they update the file, the admins rename it `smarthotelsettingsurl` and upload it to the Blob Storage they created earlier.

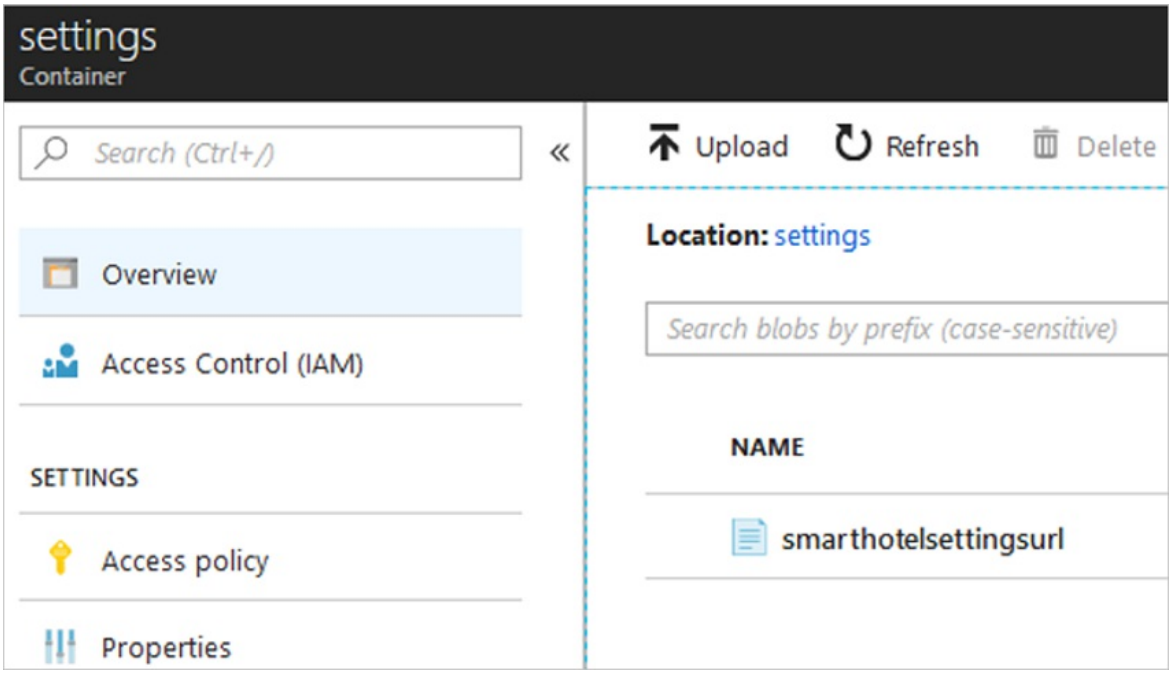


Figure 58: Renaming and uploading the file.

6. They select the file to get the URL. The URL is used by the application when it pulls down the configuration files.

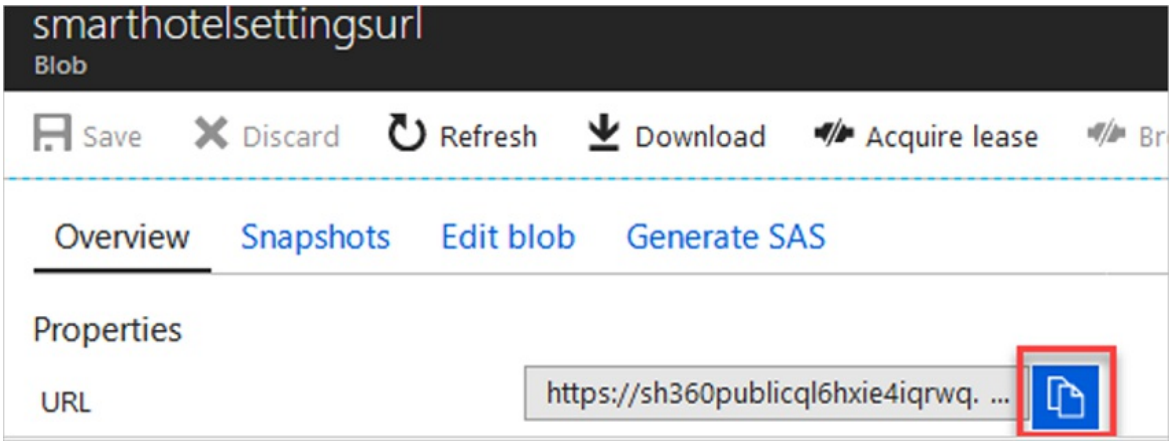


Figure 59: The application URL.

7. In the `appsettings.Production.json` file, they update the `SettingsURL` to the URL of the new file.

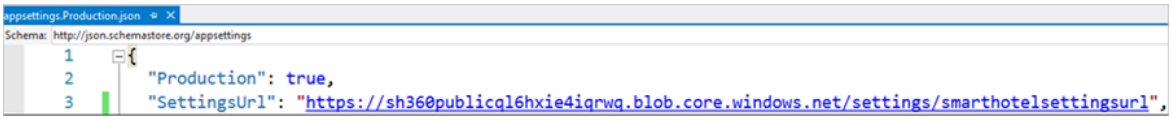


Figure 60: Updating the URL to the new file.

Deploy the website to Azure App Service

Contoso admins can now publish the website.

1. They open Azure DevOps and, in the `SmartHotelFrontEnd` project in **Builds and Releases**, they select **+ New pipeline**.
2. They select **Azure DevOps Git** as a source.
3. They select the **ASP.NET Core** template.
4. They review the pipeline and check to ensure that **Publish Web Projects** and **Zip Published Projects** are selected.

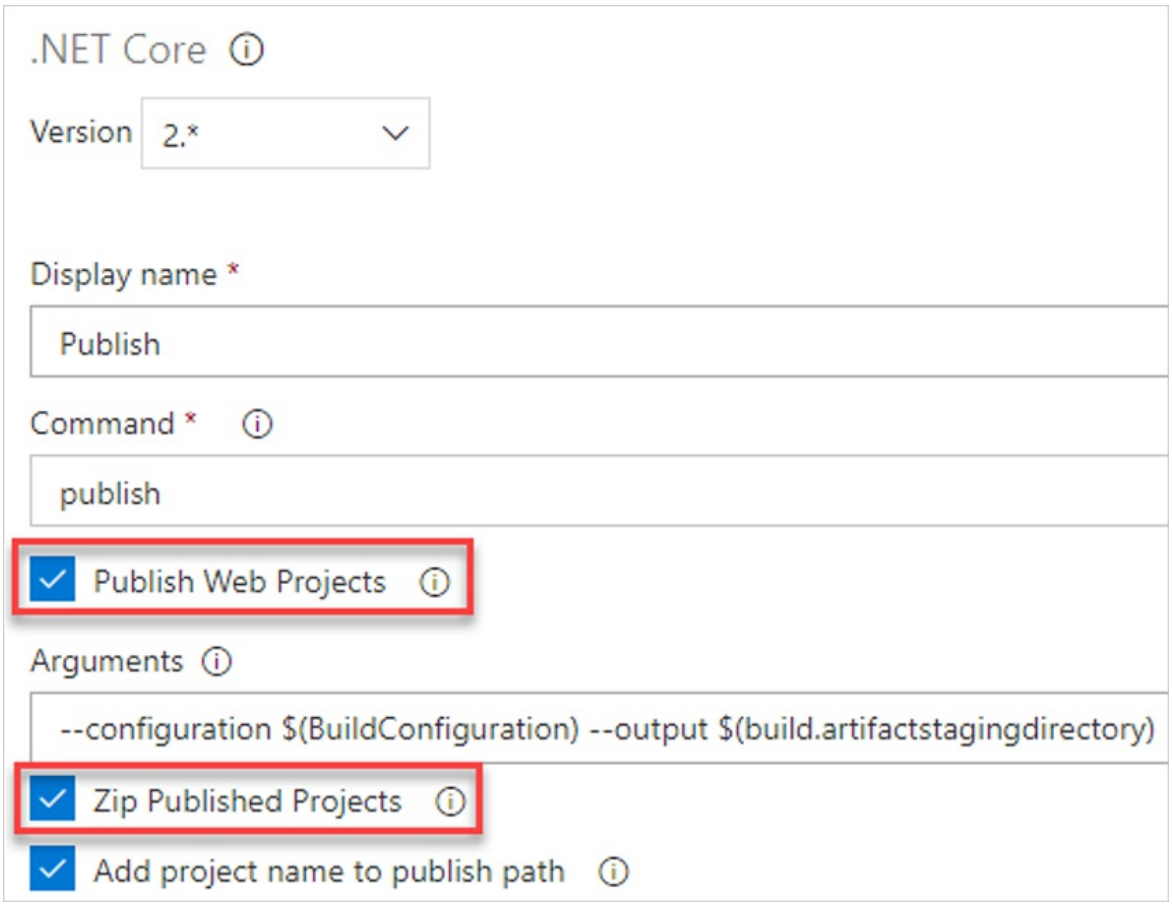


Figure 61: Pipeline settings.

5. In **Triggers**, they enable continuous integration and add the main branch. This ensures that the build pipeline starts each time the solution has new code committed to the main branch.

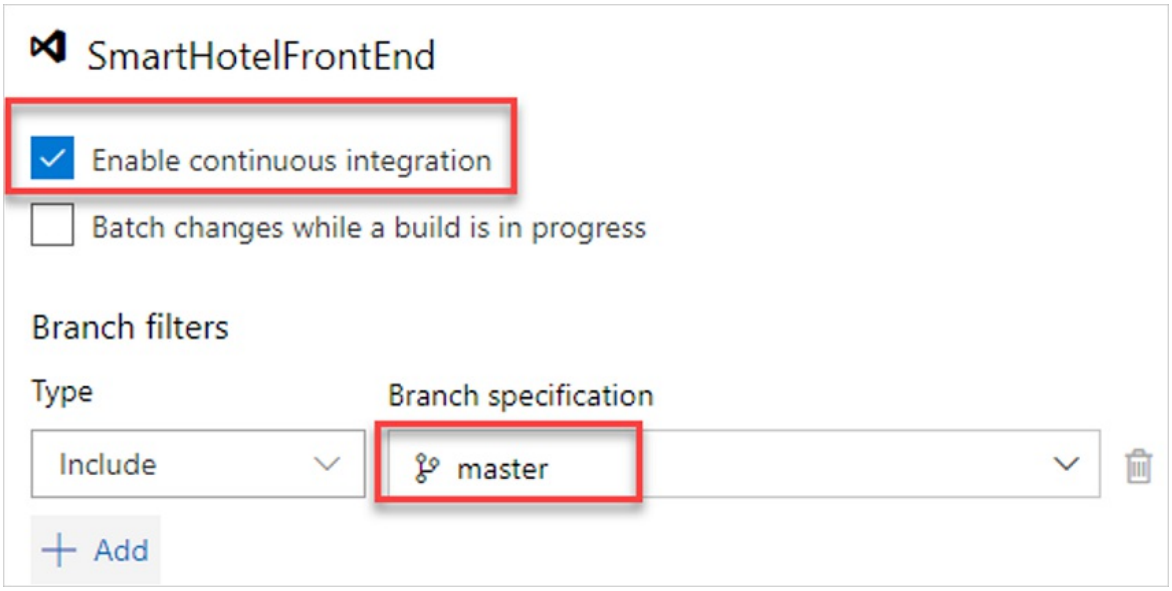


Figure 62: Enabling continuous integration.

6. They select **Save & Queue** to start a build.
7. After the build finishes, they configure a release pipeline by using **Azure App Service Deployment**.
8. They provide a stage name, **Staging**.

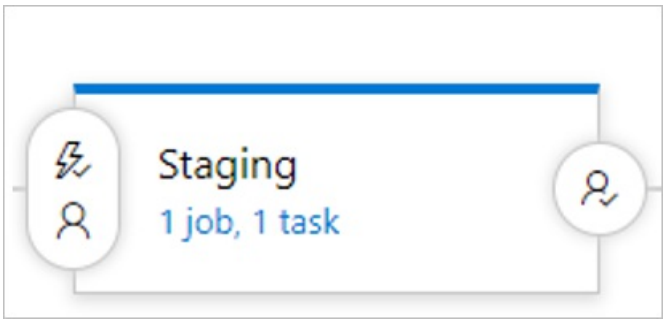


Figure 63: Naming the environment.

9. They add an artifact and select the build that they've configured.

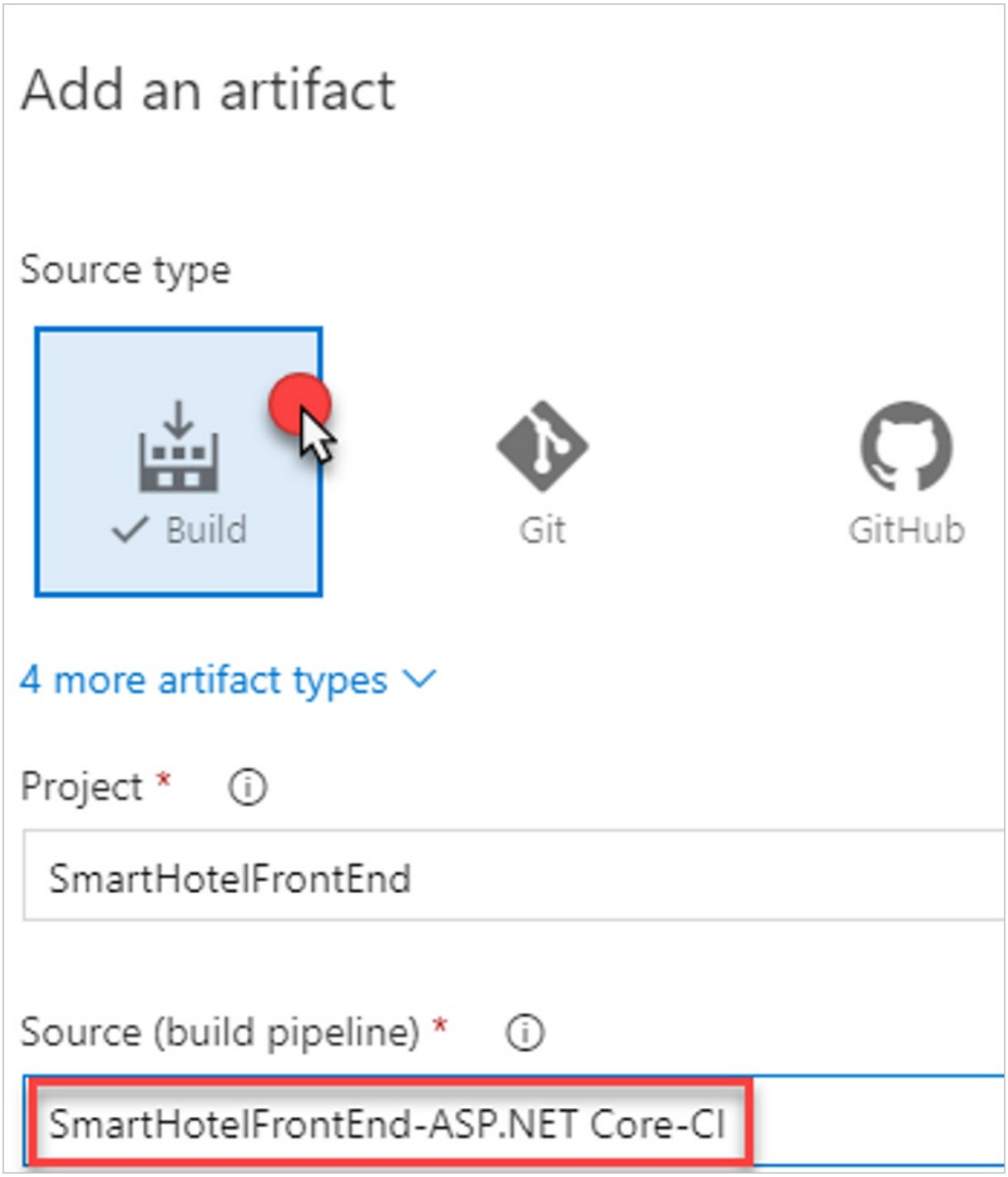


Figure 64: Adding an artifact.

10. They select the lightning bolt icon on the artifact and then set continuous deployment to **Enabled**.

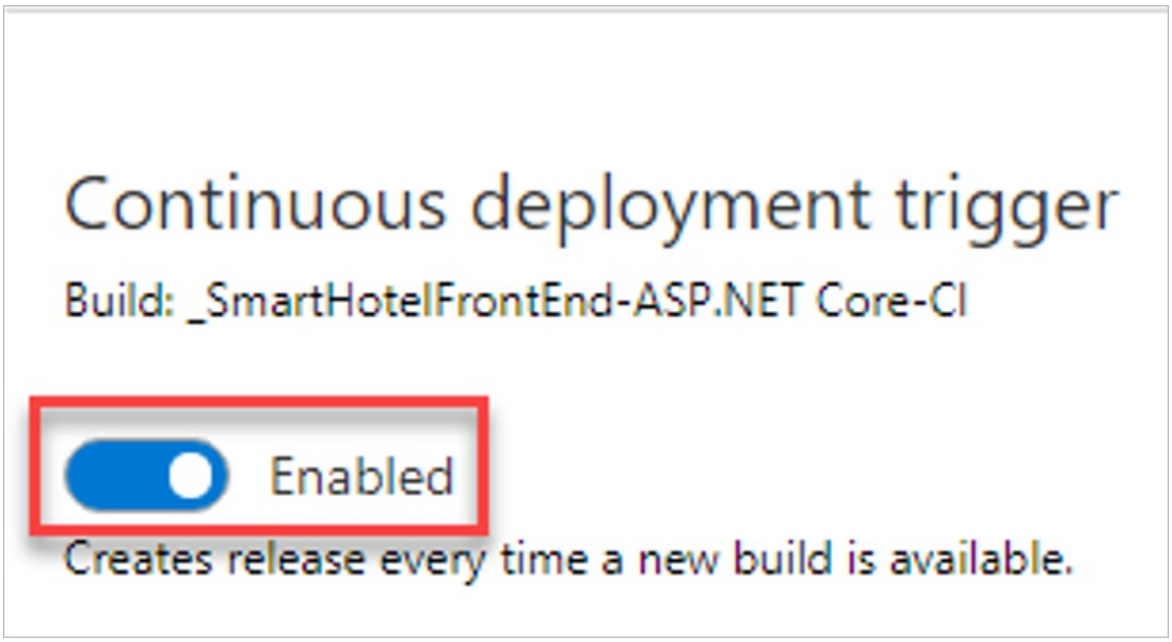


Figure 65: Enabling continuous deployment.

11. In **Environment**, they select **1 job, 1 task** under **Staging**.
12. After selecting the subscription and web app name, the admins open the **Deploy Azure App Service** task. The deployment is configured to use the **staging** deployment slot. This automatically builds code for review and approval in this slot.

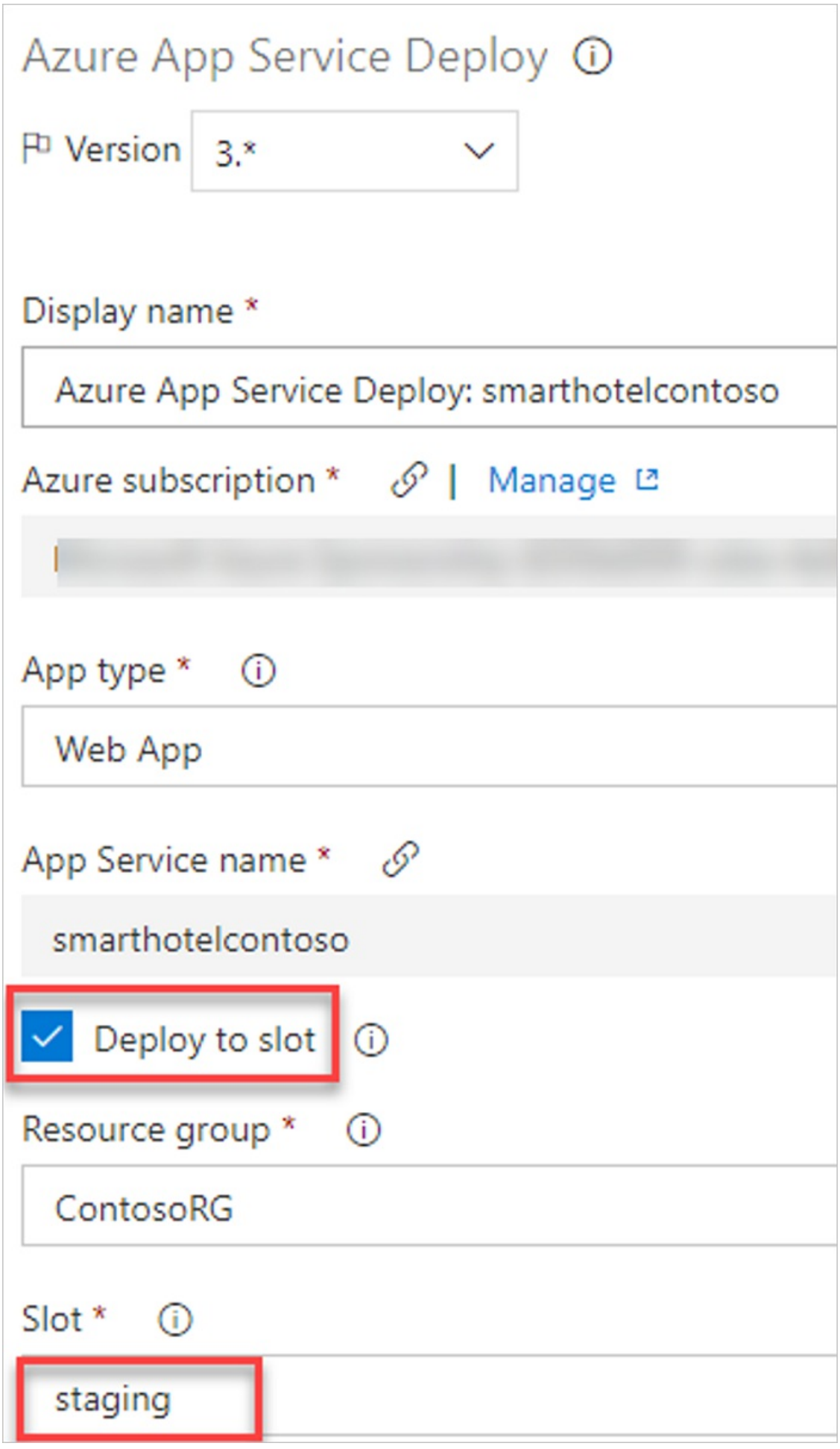


Figure 66: Deploying to a slot.

13. In the **Pipeline**, they add a new stage.

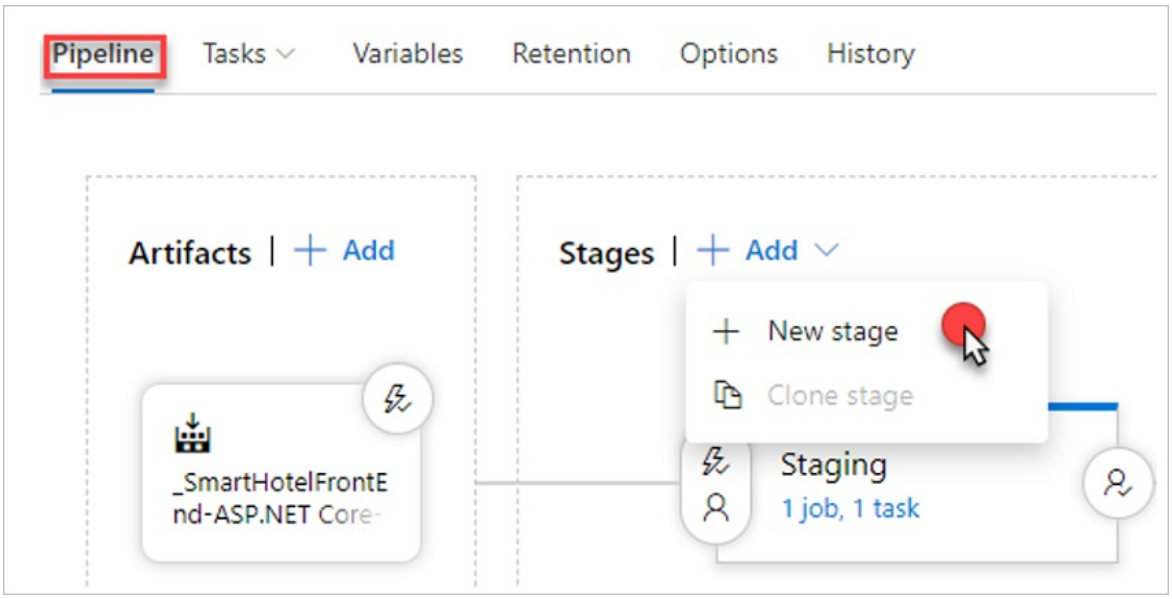


Figure 67: Adding a new stage.

14. They select **Azure App Service deployment with slot** and then name the environment **Prod**.
15. They select **1 job, 2 tasks** and then select the subscription, the app service name, and the **staging** slot.

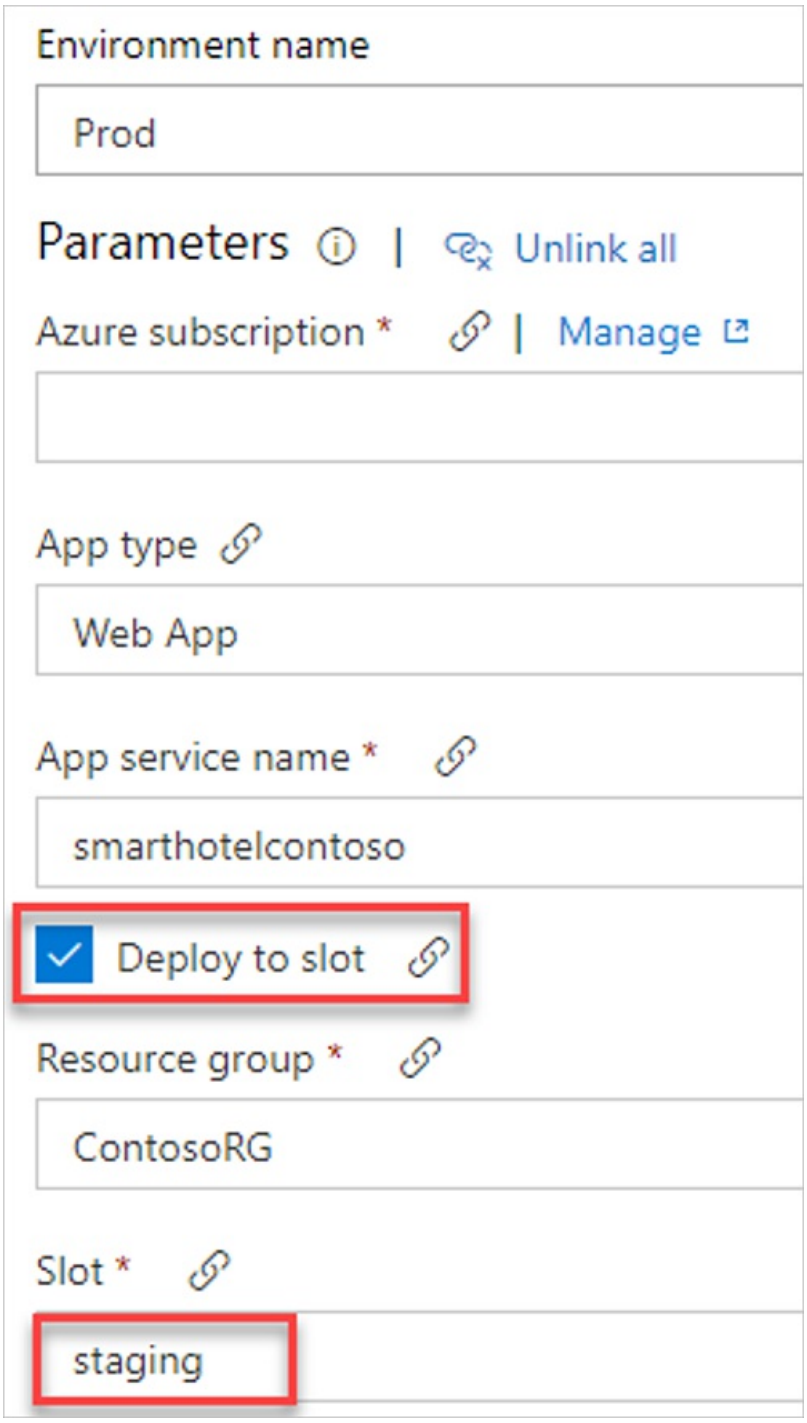


Figure 68: Naming the environment.

16. They remove the **Deploy Azure App Service to Slot** from the pipeline. It was placed there by the previous steps.

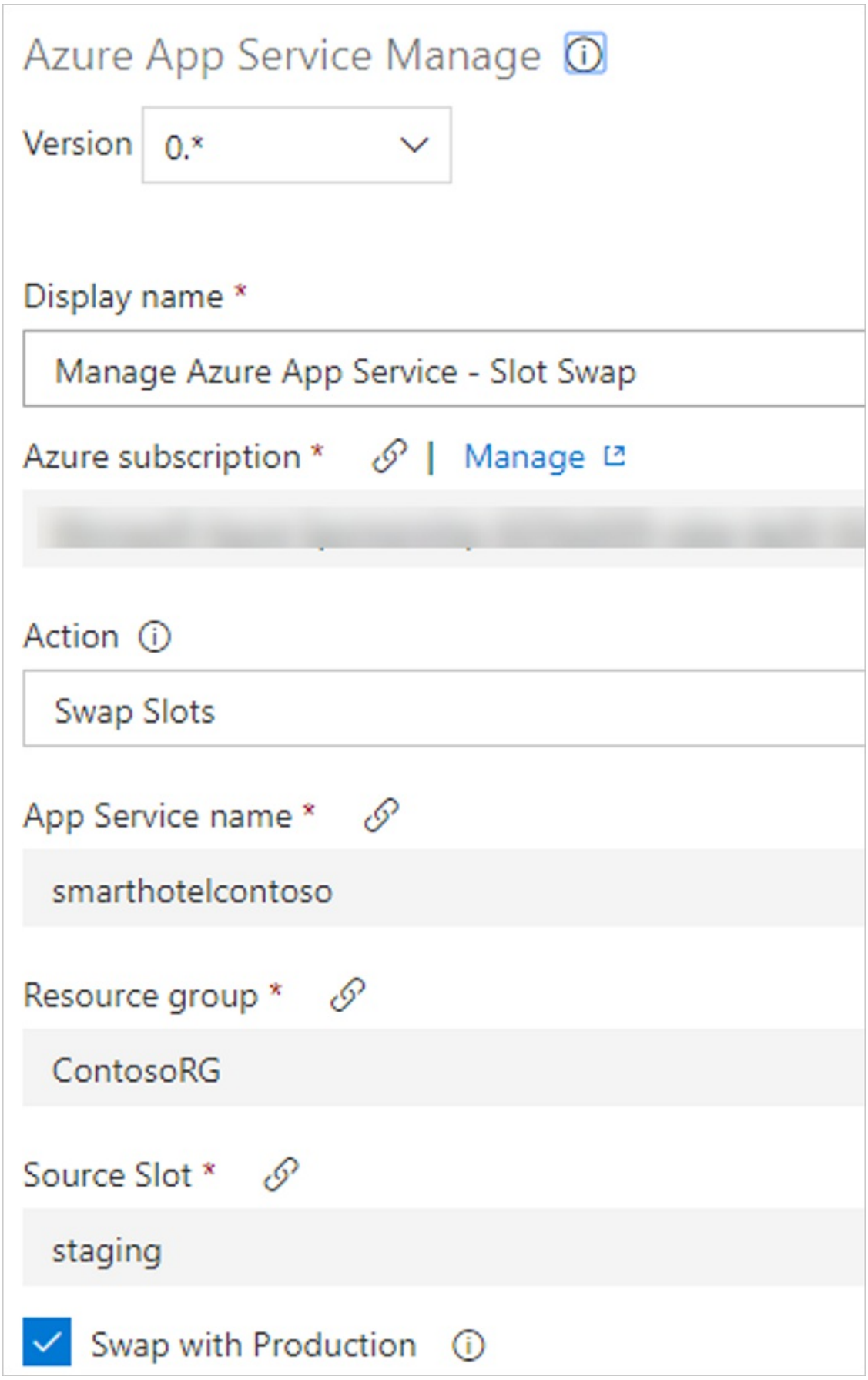


Figure 69: Removing a slot from the pipeline.

17. They save the pipeline. On the pipeline, they select **Post-deployment conditions**.

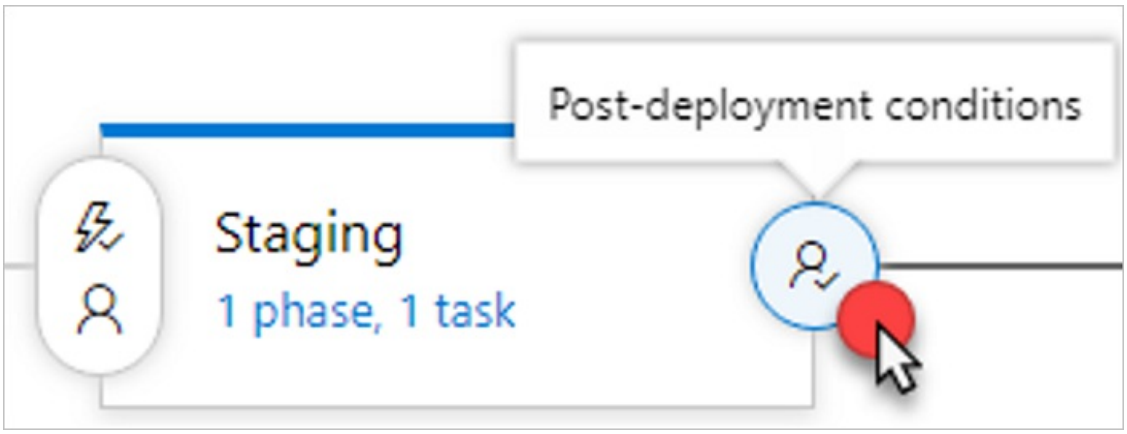


Figure 70: Post-deployment conditions.

18. They enable **Post-deployment approvals** and then add a dev lead as the approver.

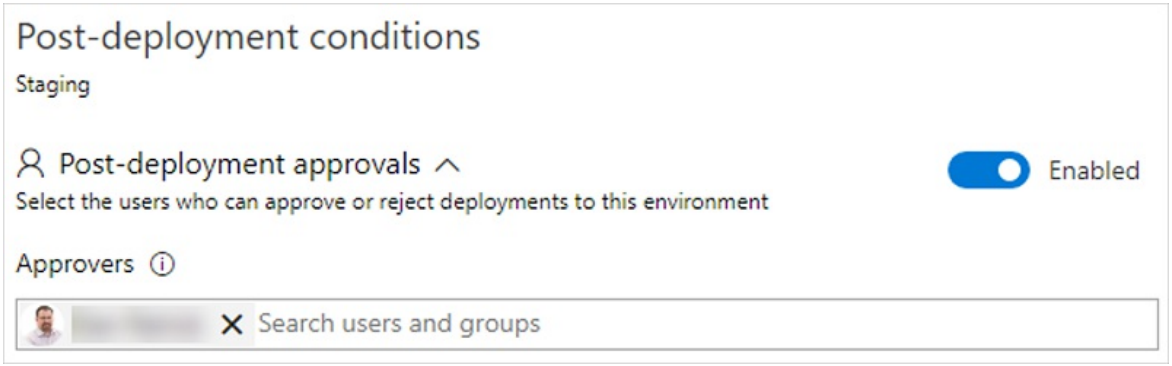


Figure 71: Adding an approver.

19. In the build pipeline, the admins manually kick off a build. This triggers the new release pipeline, which deploys the site to the staging slot. For Contoso, the URL for the slot is `https://smarthotelcontoso-staging.azurewebsites.net/`.

20. After the build finishes and the release is deployed to the slot, Azure DevOps emails the dev lead for approval.

21. The dev lead selects **View approval** and can approve or reject the request in the Azure DevOps portal.

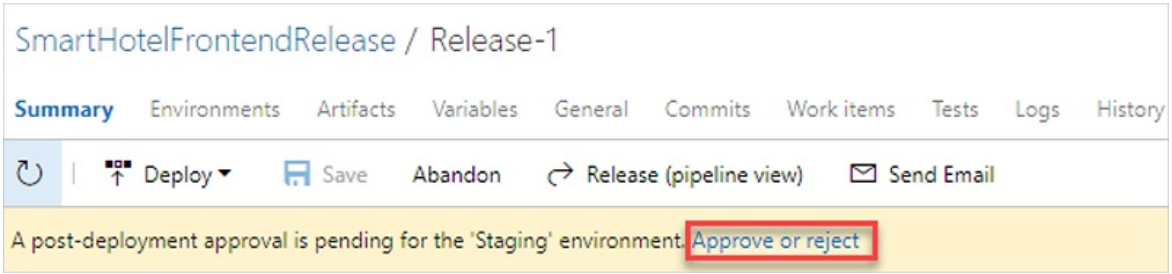


Figure 72: A pending release approval request.

22. The dev lead makes a comment and approves. This starts swapping the **staging** and **prod** slots and moves the build into production.

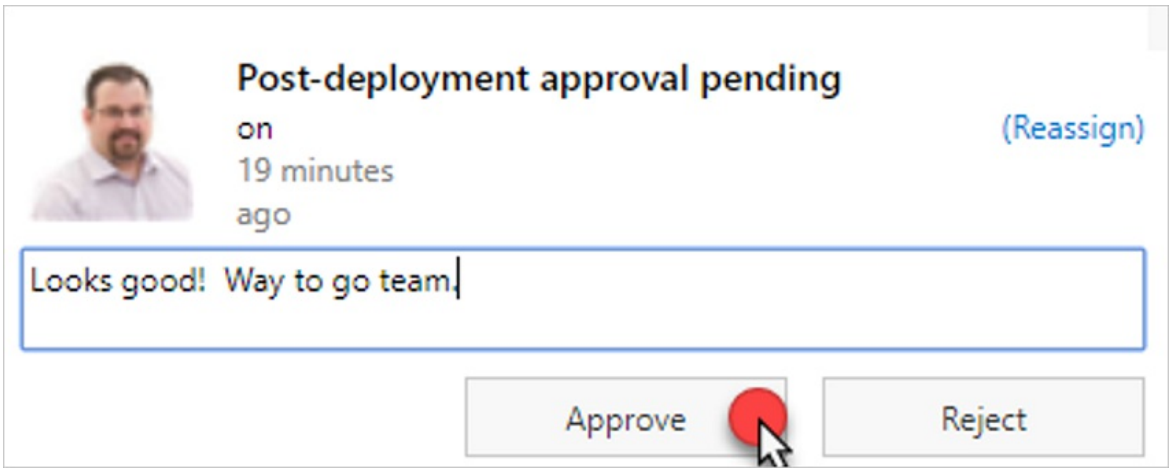


Figure 73: Moving the build into production.

23. The pipeline completes the swap.

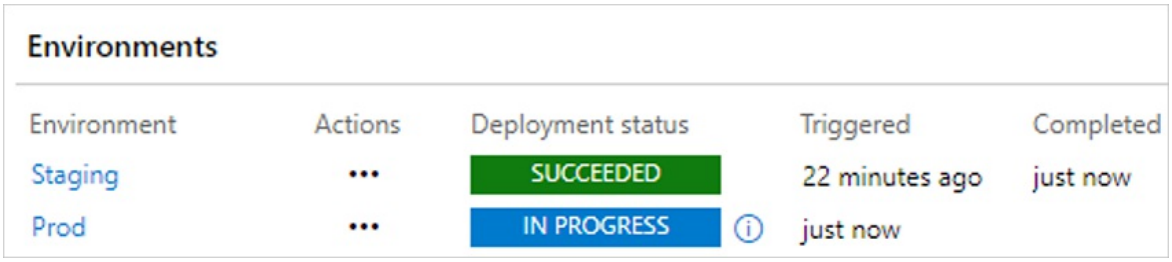


Figure 74: Completing the swap.

24. The team checks the **prod** slot to verify that the web app is in production at `https://smarthotelcontoso.azurewebsites.net/`.

Deploy the PetChecker function app

Contoso admins deploy the application by doing the following:

1. They clone the repository locally to the development machine by connecting to the Azure DevOps project.
2. In Visual Studio, they open the folder to show all the files in the repo.
3. They open the `src/PetCheckerFunction/local.settings.json` file and add the app settings for storage, the Azure Cosmos DB database, and the Computer Vision API.

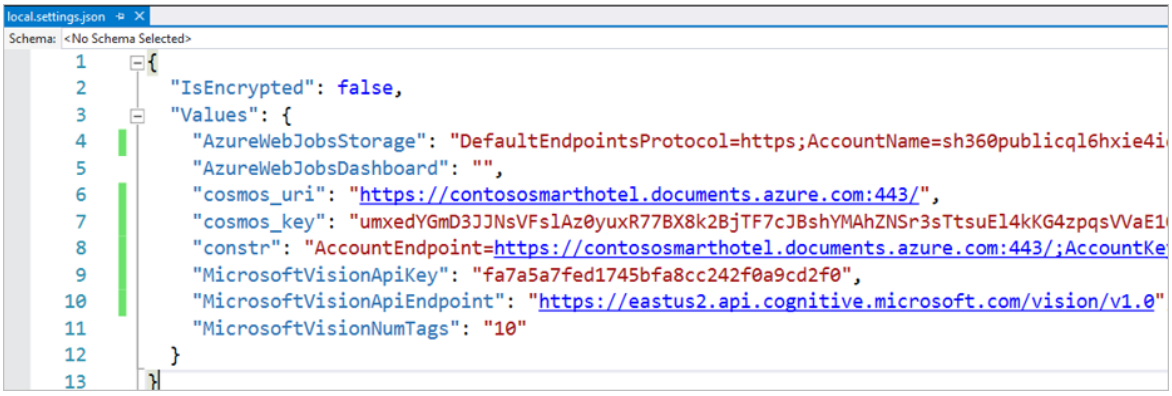


Figure 75: Deploying the function.

4. They commit the code and sync it back to Azure DevOps, pushing their changes.
5. They add a new build pipeline and then select **Azure DevOps Git** for the source.
6. They select the **ASP.NET Core (.NET Framework)** template.
7. They accept the defaults for the template.

8. Under **Triggers**, they select **Enable continuous integration** and then select **Save & Queue** to start a build.

9. After the build succeeds, they build a release pipeline, adding **Azure App Service deployment with slot**.

10. They name the environment **Prod** and then select the subscription. They set the **App type** to **Function App** and the app service name as `smarthotelpetchecker`.

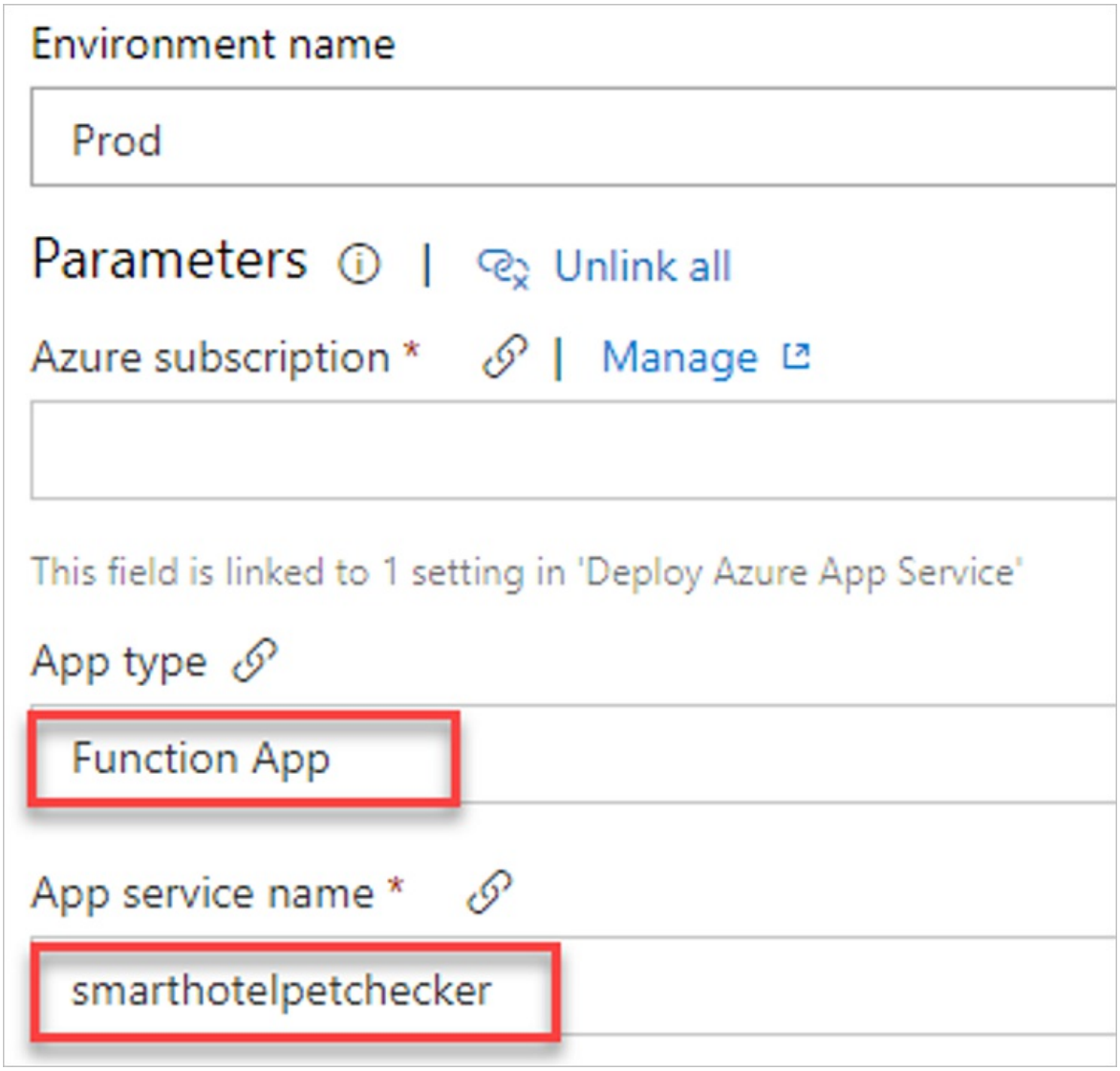


Figure 76: The function app.

11. They add an artifact, **Build**.

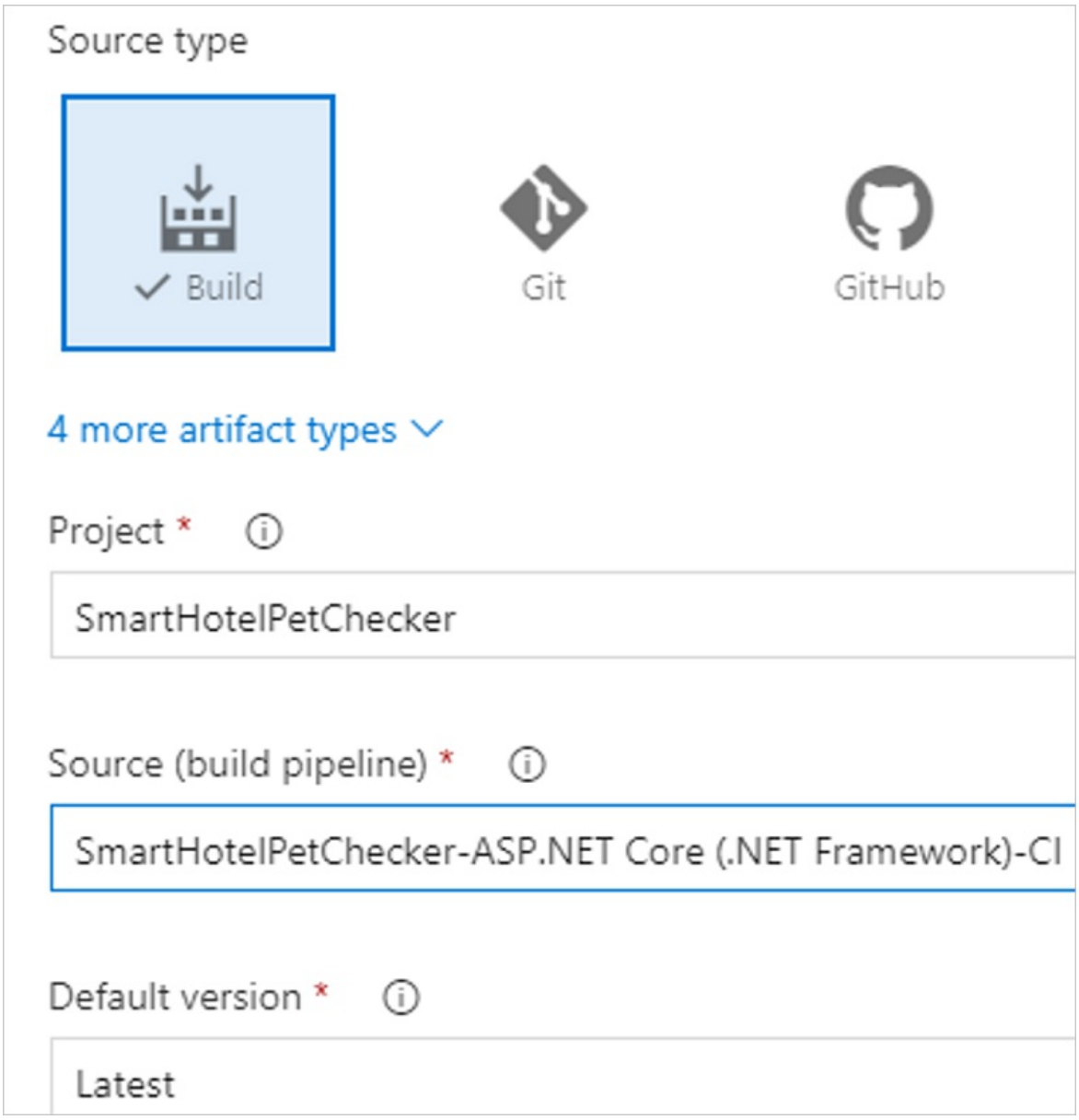


Figure 77: Adding an artifact.

12. They enable **Continuous deployment trigger** and then select **Save**.
13. They select **Queue new build** to run the full CI/CD pipeline.
14. After the function is deployed, it appears in the Azure portal with the status **Running**.

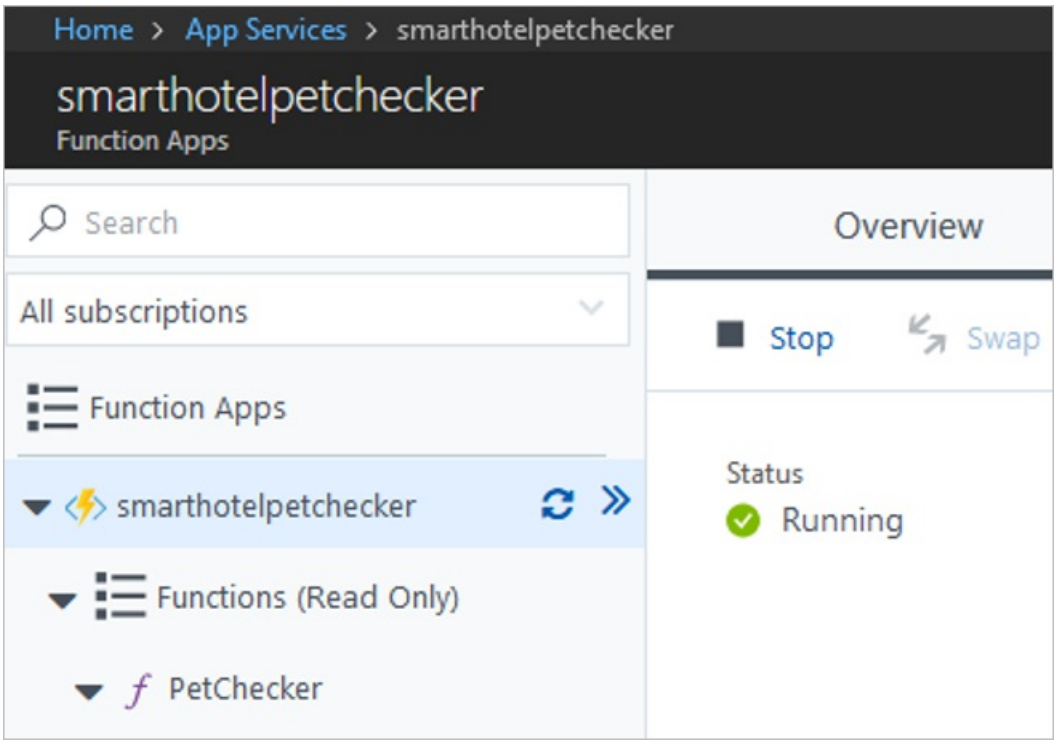


Figure 78: Updating the function's status.

15. They browse to the pet checker application, at `http://smarthotel360public.azurewebsites.net/pets`, to verify that it's working properly.

16. They select the avatar to upload a picture.

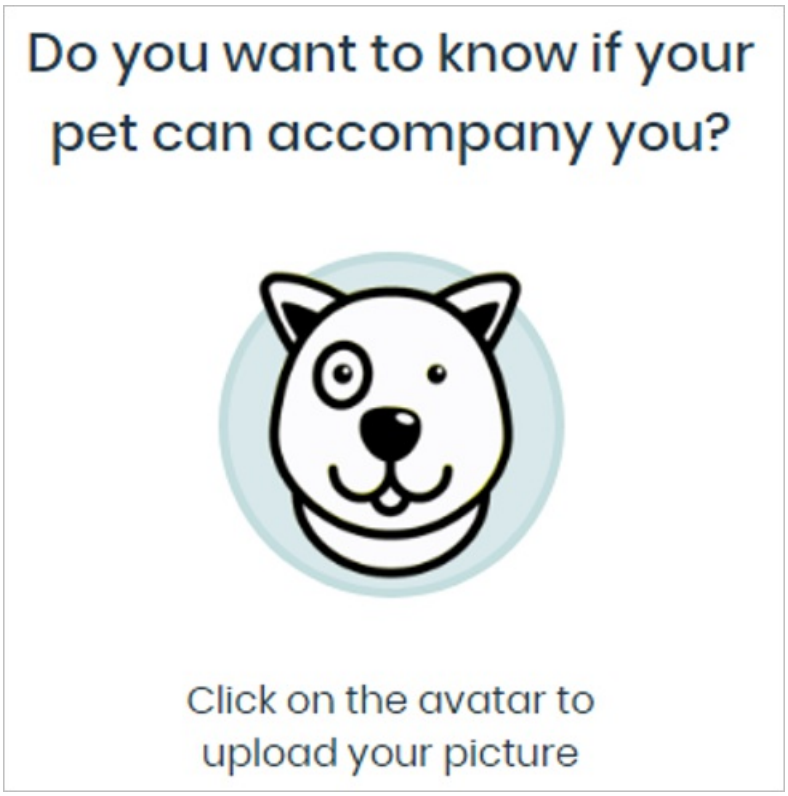


Figure 79: Assigning a picture to an avatar.

17. The first photo they want to check is of a small dog.

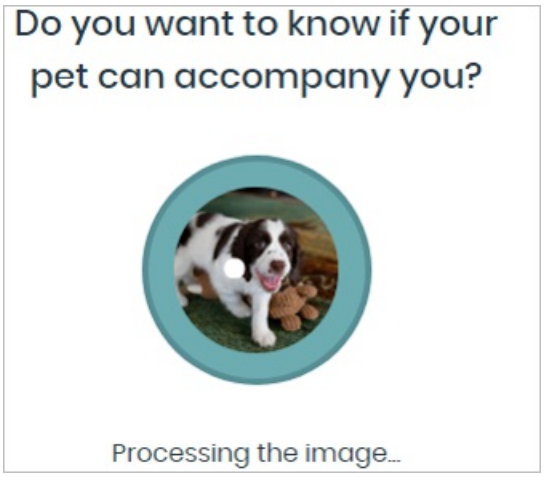


Figure 80: Checking the photo.

18. The application returns an acceptance message.

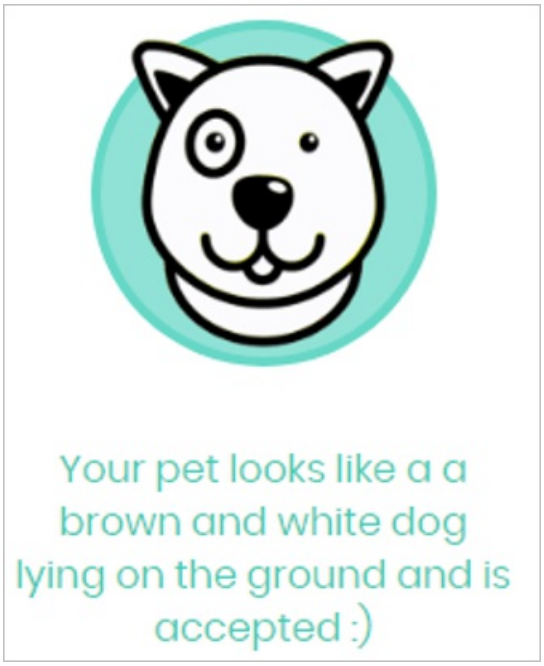


Figure 81: An acceptance message.

Review the deployment

With the migrated resources in Azure, Contoso now needs to fully operationalize and secure the new infrastructure.

Security

- Contoso needs to ensure that the new databases are secure. To learn more, see Overview of Azure SQL Database and SQL Managed Instance security capabilities.
- The application must be updated to use SSL with certificates. The container instance should be redeployed to answer on 443.
- Contoso should consider using Azure Key Vault to help protect secrets for their Service Fabric applications. To learn more, see Manage encrypted secrets in Service Fabric applications.

Backups and disaster recovery

- Contoso needs to review backup requirements for Azure SQL Database.
- Contoso should consider implementing SQL failover groups to provide regional failover for the database.
- Contoso can use geo-replication for the Azure Container Registry Premium SKU.
- Azure Cosmos DB is backed up automatically. To learn more, see Online backup and on-demand data restore in Azure Cosmos DB.

Licensing and cost optimization

- After all resources are deployed, Contoso should assign Azure tags based on their infrastructure planning.
- All licensing is built into the cost of the PaaS services that Contoso is consuming. This will be deducted from the Enterprise Agreement.
- Contoso will enable Azure Cost Management + Billing to help monitor and manage the Azure resources.

Conclusion

In this article, Contoso rebuilds the SmartHotel360 application in Azure. The on-premises application front-end VM is rebuilt for Azure App Service web apps. The application back end is built by using microservices that are deployed to containers managed by AKS. Contoso enhanced functionality with a pet photo application.

Suggested skills

Microsoft Learn is a new approach to learning. Readiness for the new skills and responsibilities that come with cloud adoption doesn't come easily. Microsoft Learn provides a more rewarding approach to hands-on learning that helps you achieve your goals faster. With Microsoft Learn, you can earn points, rise through levels, and achieve more.

Here are two examples of tailored learning paths on Microsoft Learn that align with the Contoso SmartHotel360 application in Azure.

- Deploy a website to Azure with Azure App Service: By creating web apps in Azure, you can publish and manage your website easily without having to work with the underlying servers, storage, or network assets. Instead, you can focus on your website features and rely on the robust Azure platform to help provide secure access to your site.
- Process and classify images with the Azure cognitive vision services: Azure Cognitive Services offers prebuilt functionality to enable computer vision functionality in your applications. Learn how to use the Azure cognitive vision services to detect faces, tag and classify images, and identify objects.

Refactor a Team Foundation Server deployment to Azure DevOps Services

7/15/2021 • 17 minutes to read • Edit Online

This article shows how the fictional company Contoso refactors its on-premises Visual Studio Team Foundation Server deployment by migrating it to Azure DevOps Services in Azure. The Contoso development team has used Team Foundation Server for team collaboration and source control for the past five years. Now, the team wants to move to a cloud-based solution for dev/test work and for source control. Azure DevOps Services will play a role as the Contoso team moves to an Azure DevOps model and develops new cloud-native applications.

Business drivers

The Contoso IT leadership team has worked closely with business partners to identify future goals. The partners aren't overly concerned with dev tools and technologies, but the team has captured these points:

- **Software:** Regardless of the core business, all companies are now software companies, including Contoso. Business leadership is interested in how IT can help lead the company with new working practices for users and new experiences for its customers.
- **Efficiency:** Contoso needs to streamline its processes and remove unnecessary procedures for developers and users. Doing so will allow the company to deliver on customer requirements more efficiently. The business needs IT to move quickly, without wasting time or money.
- **Agility:** To enable its success in a global economy, Contoso IT needs to be more responsive to the needs of the business. It must be able to react more quickly to changes in the marketplace. IT must not get in the way or become a business blocker.

Migration goals

The Contoso cloud team has pinned down the following goals for its migration to Azure DevOps Services:

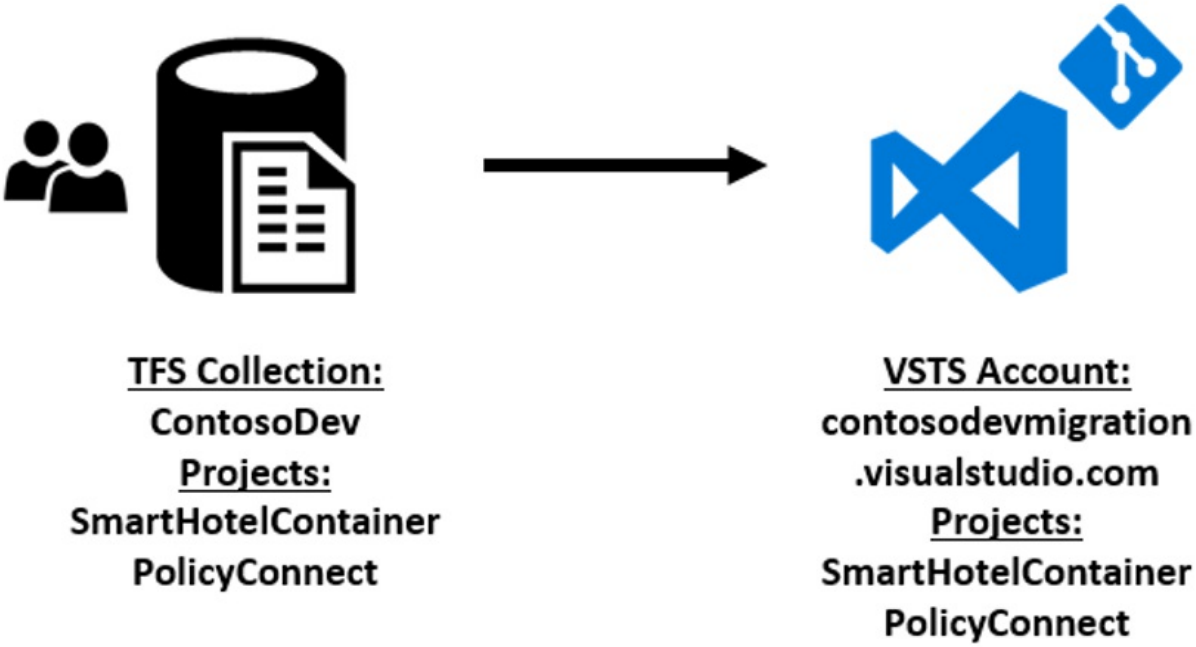
- The team needs a tool to migrate its data to the cloud. Few manual processes should be needed.
- Work item data and history for the last year must be migrated.
- The team doesn't want to set up new user names and passwords. All current system assignments must be maintained.
- The team wants to move away from Team Foundation Version Control (TFVC) to Git for source control.
- The transition to Git will be a tip migration that imports only the latest version of the source code. The transition will happen during a downtime, when all work will be halted as the code base shifts. The team understands that only the current main branch history will be available after the move.
- The team is concerned about the change and wants to test it before it does a full move. The team wants to retain access to Team Foundation Server even after the move to Azure DevOps Services.
- The team has multiple collections and, to better understand the process, it wants to start with one that has only a few projects.
- The team understands that Team Foundation Server collections are a one-to-one relationship with Azure DevOps Services organizations, so it will have multiple URLs. But this matches its current model of separation for code bases and projects.

Proposed architecture

- Contoso will move its Team Foundation Server projects to the cloud, and it will no longer host its projects or

source control on-premises.

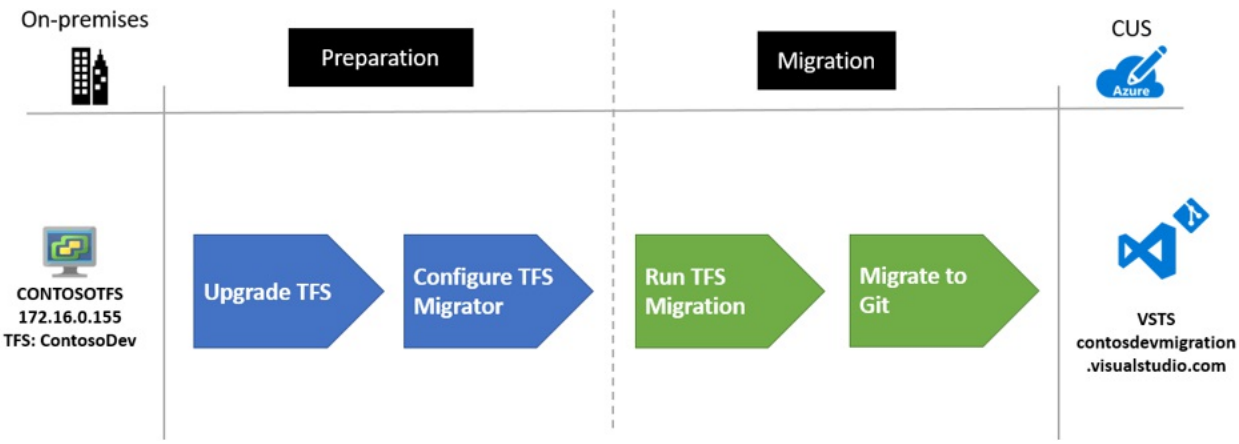
- Team Foundation Server will be migrated to Azure DevOps Services.
- Currently, Contoso has one Team Foundation Server collection, named `ContosoDev`, which will be migrated to an Azure DevOps Services organization called `contosodevmigration.visualstudio.com`.
- The projects, work items, bugs, and iterations from the last year will be migrated to Azure DevOps Services.
- Contoso will use its Azure Active Directory (Azure AD) instance, which it set up when it deployed its Azure infrastructure at the beginning of the migration planning.



Migration process

Contoso will complete the migration process as follows:

1. Significant preparation is required. First, Contoso must upgrade its Team Foundation Server implementation to a supported level. Contoso is currently running Team Foundation Server 2017 Update 3, but to use database migration it needs to run a supported 2018 version with the latest updates.
2. After Contoso upgrades, it will run the Team Foundation Server migration tool and validate its collection.
3. Contoso will build a set of preparation files and then perform a migration dry run for testing.
4. Contoso will then run another migration, this time a full migration that includes work items, bugs, sprints, and code.
5. After the migration, Contoso will move its code from TFVC to Git.



Prerequisites

To run this scenario, Contoso needs to meet the following prerequisites:

REQUIREMENTS	DETAILS
Azure subscription	Contoso created subscriptions in an earlier article in this series. If you don't have an Azure subscription, create a free account. If you create a free account, you're the administrator of your subscription and can perform all actions. If you use an existing subscription and you're not the administrator, you need to work with the admin to assign you Owner or Contributor permissions. If you need more granular permissions, see Manage Site Recovery access with Azure role-based access control (Azure RBAC).
Azure infrastructure	Contoso set up its Azure infrastructure as described in Azure infrastructure for migration.
On-premises Team Foundation Server instance	The on-premises instance needs to either run Team Foundation Server 2018 upgrade 2 or be upgraded to it as part of this process.

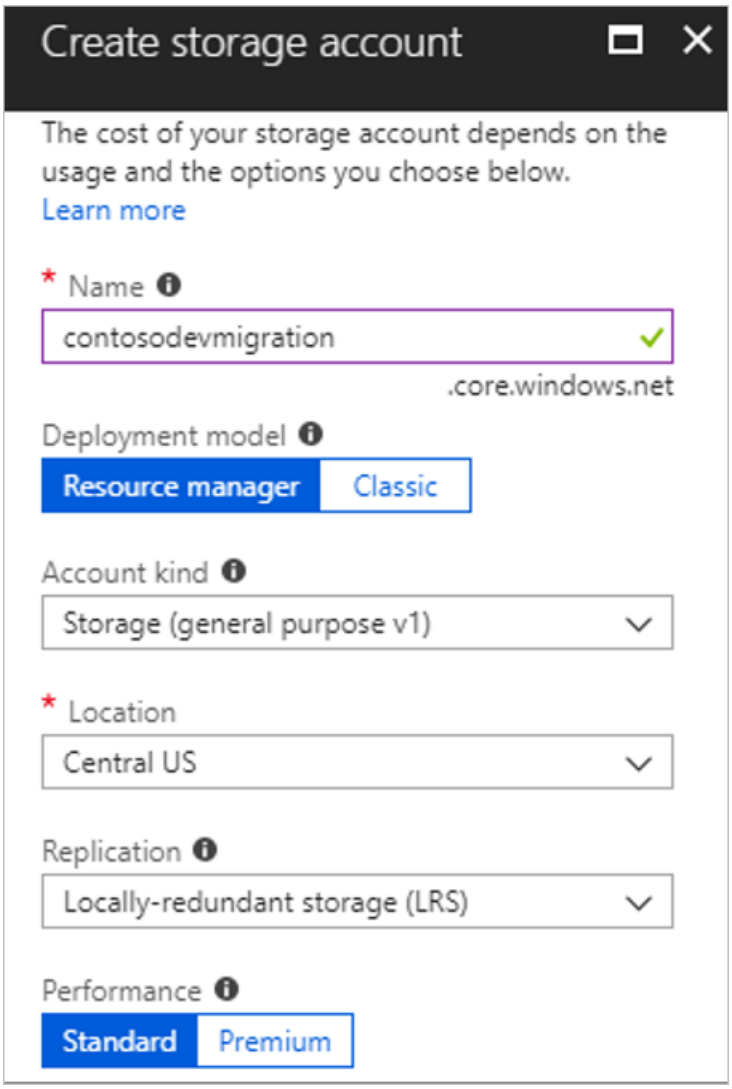
Scenario steps

Here's how Contoso will complete the migration:

- **Step 1: Create an Azure storage account**. This storage account will be used during the migration process.
- **Step 2: Upgrade Team Foundation Server**. Contoso will upgrade its deployment to Team Foundation Server 2018 upgrade 2.
- **Step 3: Validate the Team Foundation Server collection**. Contoso will validate the Team Foundation Server collection in preparation for the migration.
- **Step 4: Build the migration files**. Contoso will create the migration files by using the Team Foundation Server migration tool.

Step 1: Create an Azure Storage account

1. In the Azure portal, Contoso admins create a storage account (`contosodevmigration`).
2. They place the account in the secondary region, which they use for failover (`Central US`). They use a general-purpose standard account with locally redundant storage.



Need more help?

- Introduction to Azure Storage.
- Create a storage account.

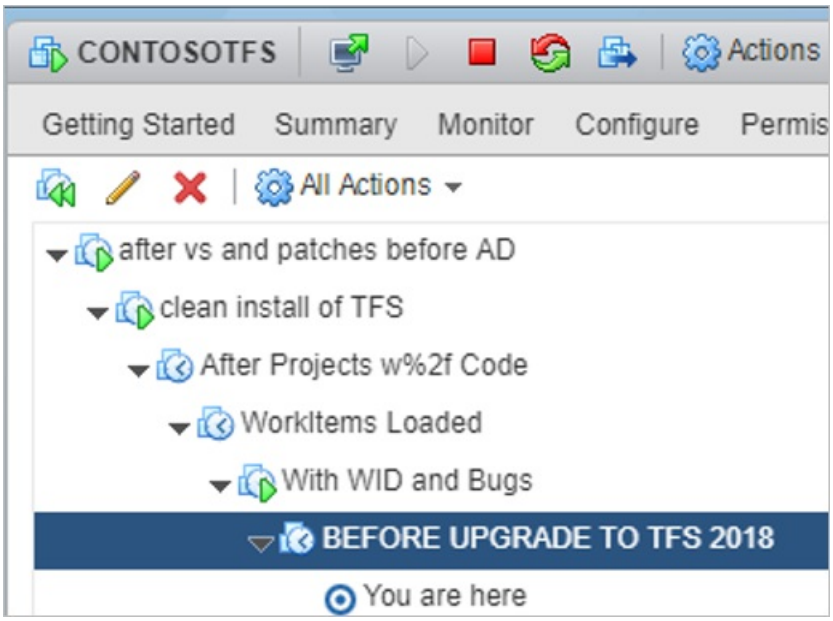
Step 2: Upgrade Team Foundation Server

Contoso admins upgrade the Team Foundation Server instance to Team Foundation Server 2018 Update 2. Before they start, they:

- Download Team Foundation Server 2018 Update 2.
- Verify the hardware requirements.
- Read the release notes and upgrade gotchas.

They upgrade as follows:

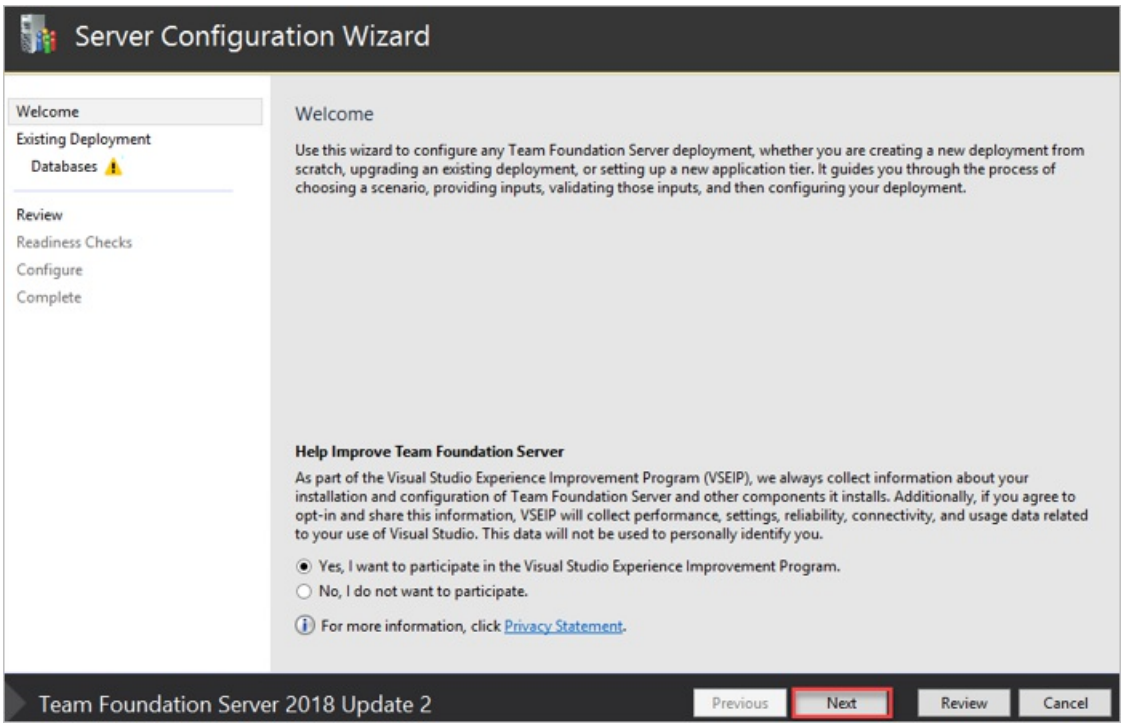
1. To start, the admins back up their Team Foundation Server instance, which is running on a VMware virtual machine (VM), and they take a VMware snapshot.



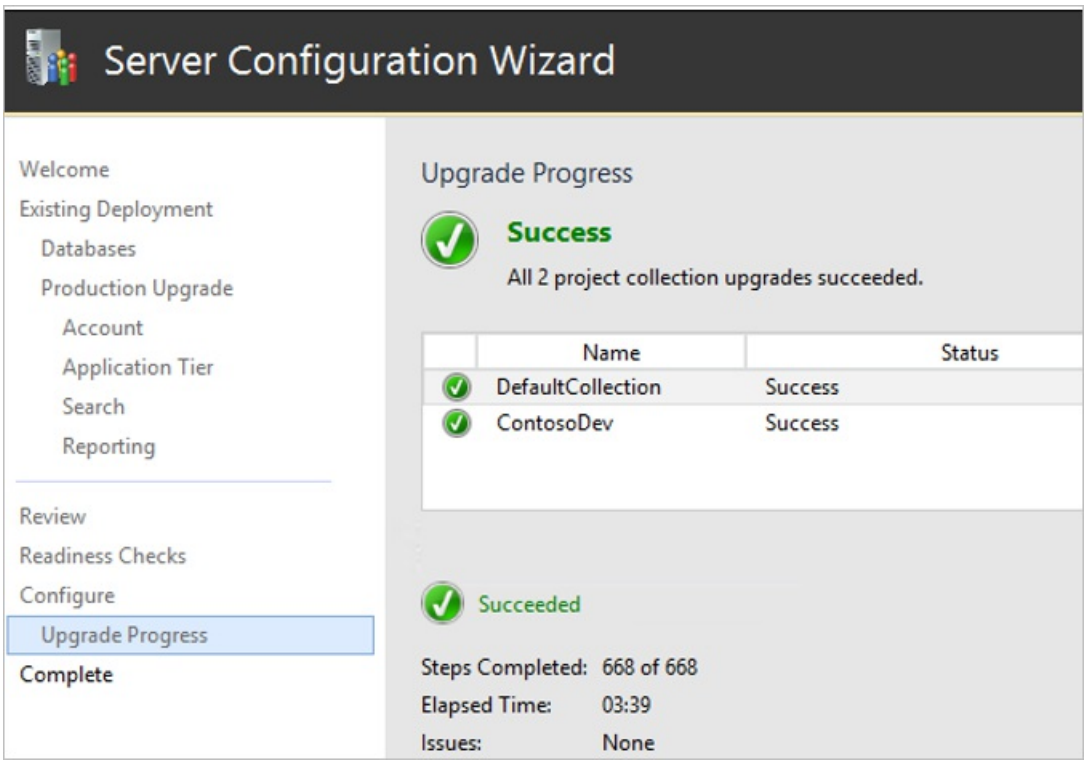
2. The Team Foundation Server installer starts, and they choose the installation location. The installer needs internet access.



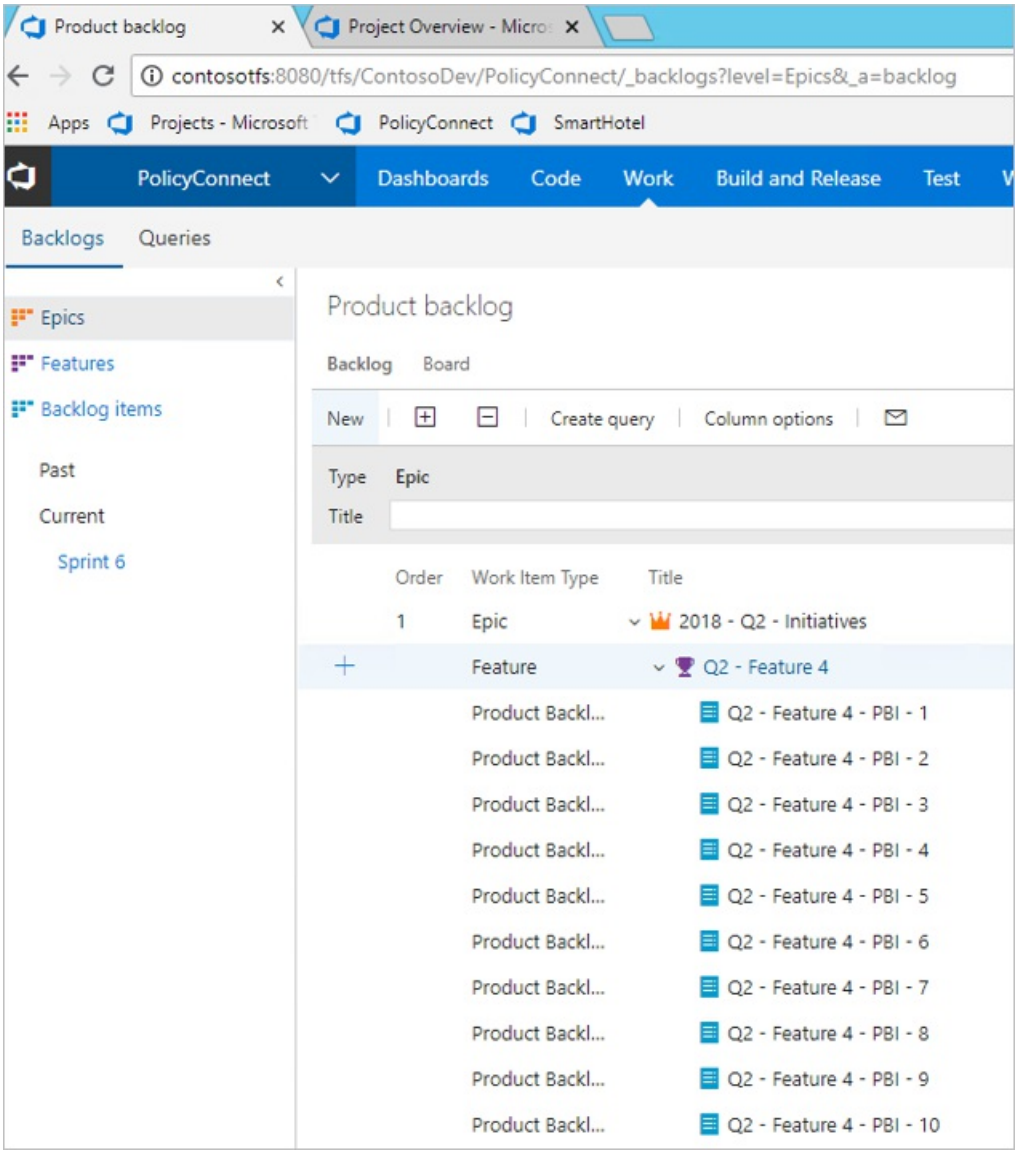
3. After the installation finishes, the Server Configuration Wizard starts.



4. After verification, the Server Configuration Wizard completes the upgrade.



5. The admins verify the Team Foundation Server installation by reviewing projects, work items, and code.



> **NOTE**
> Some Team Foundation Server upgrades need to run the Configure Features Wizard after the upgrade finishes. Learn more.

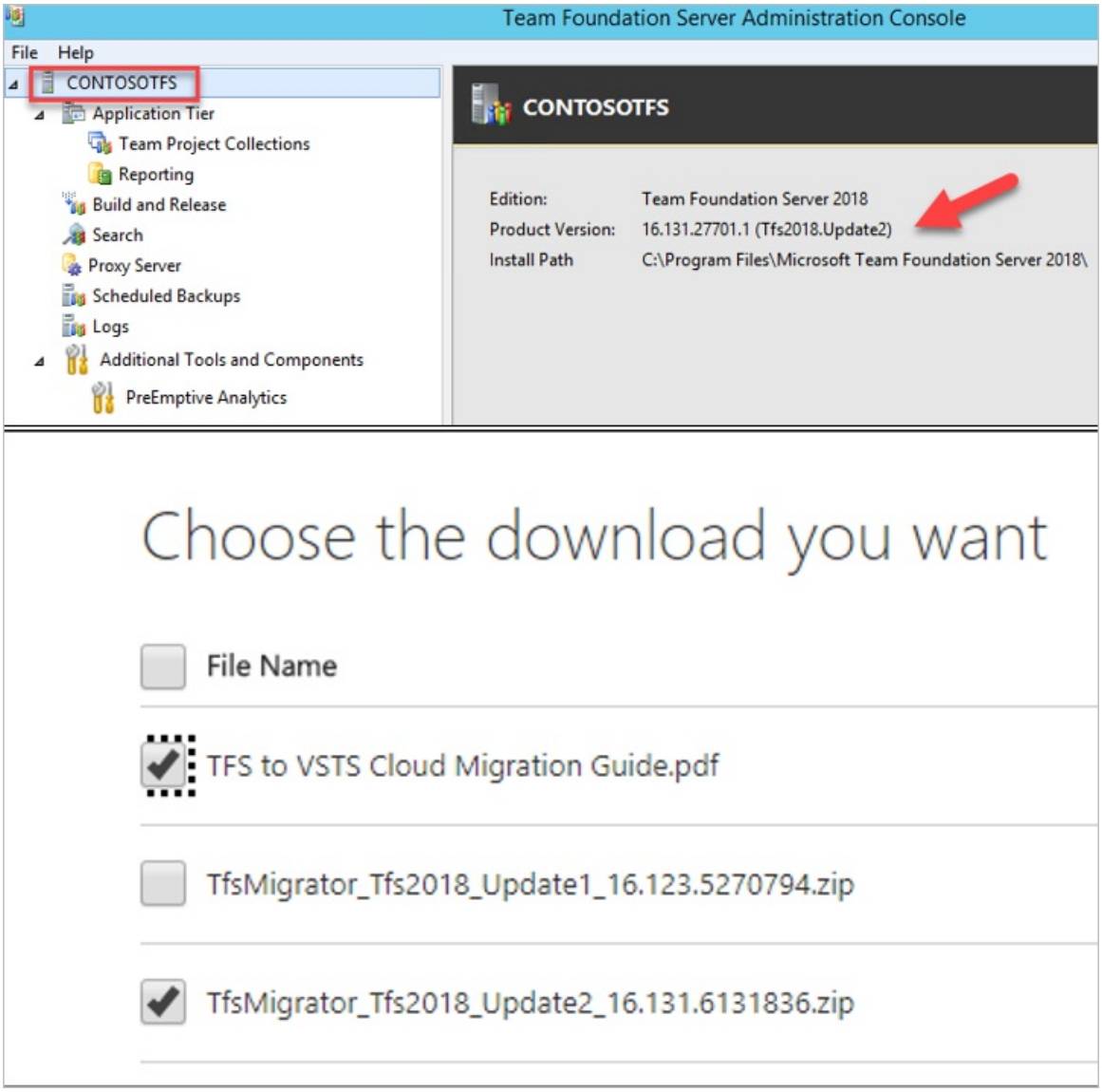
Need more help?

Learn about upgrading Team Foundation Server.

Step 3: Validate the Team Foundation Server collection

Contoso admins run the Team Foundation Server migration tool against the `contosodev` collection database to validate it before migration.

1. They download and unzip the Team Foundation Server migration tool. It's important to download the version for the Team Foundation Server update that's running. The version can be checked in the admin console.



2. They run the tool to perform the validation by specifying the URL of the project collection, as shown in the following command:

   ```
   TfsMigrator validate /collection:http://contosotfs:8080/tfs/ContosoDev
   ```

   The tool shows an error.

```
--------------------------------------------------------------------------------
  Validating ContosoDev (id = 5f815a1b-c82e-4684-918f-5d368dad733a)
--------------------------------------------------------------------------------

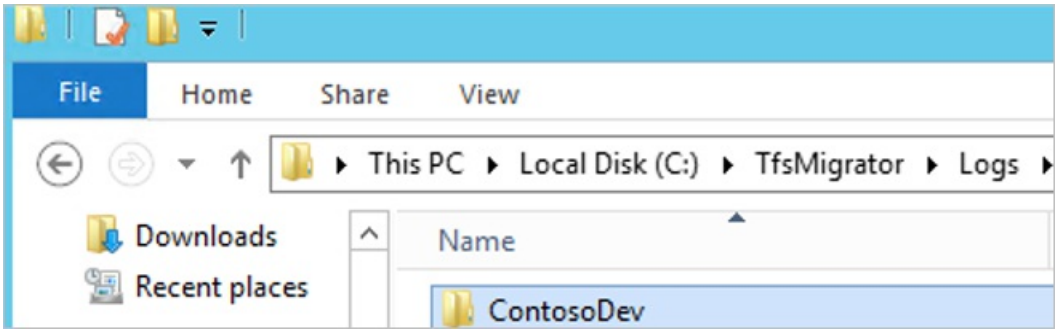
Validating that the collection is OK to be imported...

Validating Collection Metadata       Passed  (Step 15 of 15)
Validating Project Processes         Passed  (Validating Project 2 of 2)

Collection validation completed.

Results:
+ All collection validations completed with the following errors:
+ [Error] Errors encountered during identity validation, please review the logs for more details.
```

3. They locate the log files in the `Logs` folder, just before the tool location. A log file is generated for each major validation. `TfsMigration.log` holds the main information.



4. They find this entry, which is related to identity.

```
Identity Validation failed, please provide a TenantDomainName
Validation completed 'Validate Identities' with result Failed, message Errors encountered during identity validation
```

5. They run `TfsMigrator validate /help` at the command line, and they see that the command `/tenantDomainName` seems to be required to validate identities.

```
------------------------------------
  Tenant Domain Name
------------------------------------

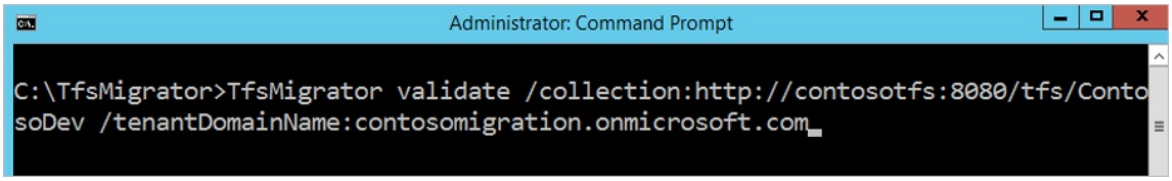
The tenant domain specifies the directory to use when validating your Identities list in the import.json

  /tenantDomainName:name -  Name of a AAD domain.
```

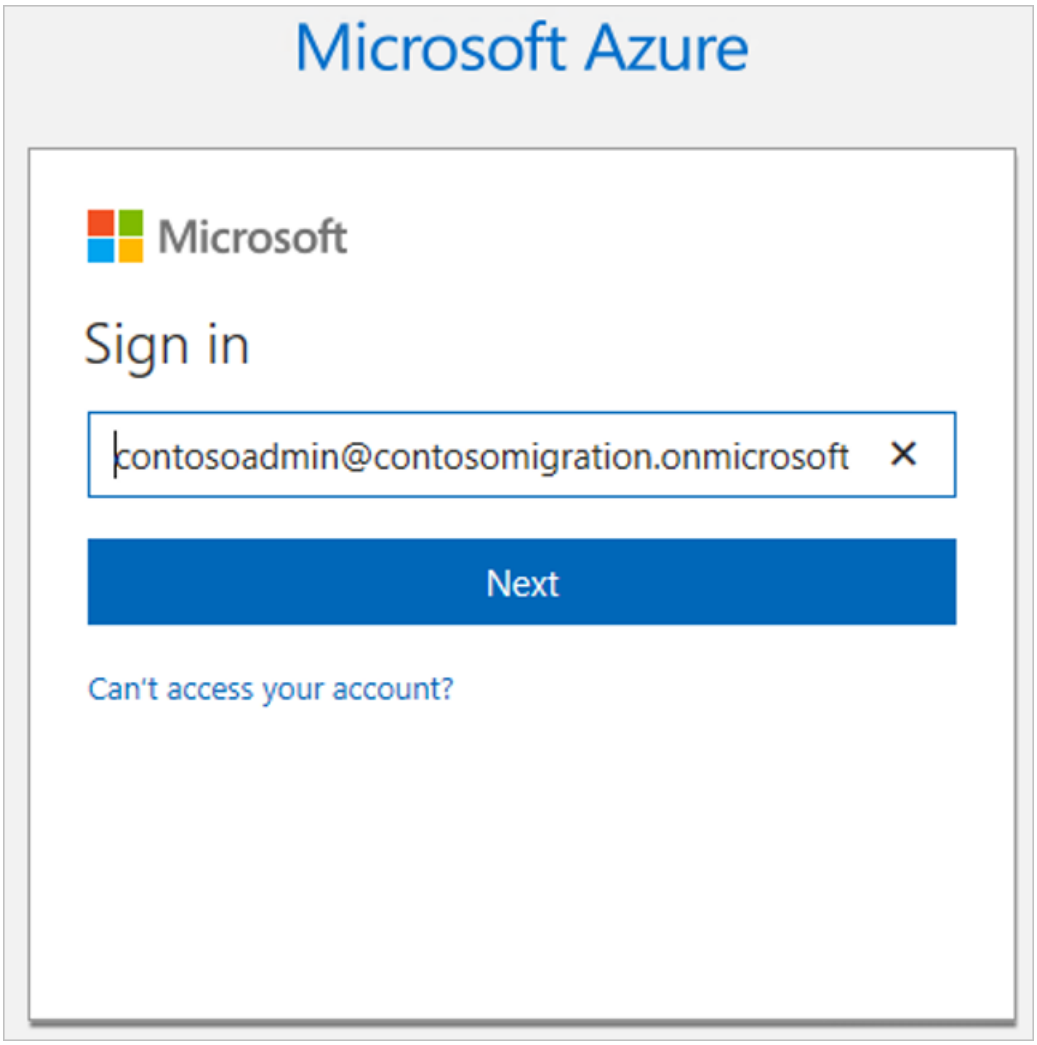
6. They run the validation command again and include this value and their Azure AD name,

```
TfsMigrator validate /collection:http://contosotfs:8080/tfs/ContosoDev
/tenantDomainName:contosomigration.onmicrosoft.com
```

.



7. In the Azure AD sign-in window that opens, they enter the credentials of a global admin user.



8. The validation passes and is confirmed by the tool.

```
------------------------------------
  Guidance
------------------------------------

Collection validation was successful. Recommended next steps:

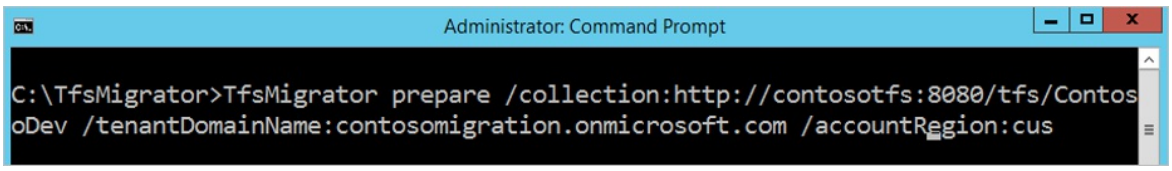
  1) If you haven't run Prepare, next you'll need to generate import files
  2) See TfsMigrator Prepare /help for guidance on how to run the command
  2) If you've run prepare and made changes to the generated files be sure to re
-validate them
  3) See TfsMigrator Validate /help for details on re-validating import files
```

Step 4: Build the migration files

With the validation complete, Contoso admins can use the Team Foundation Server migration tool to build the migration files.

1. They run the preparation step in the tool.

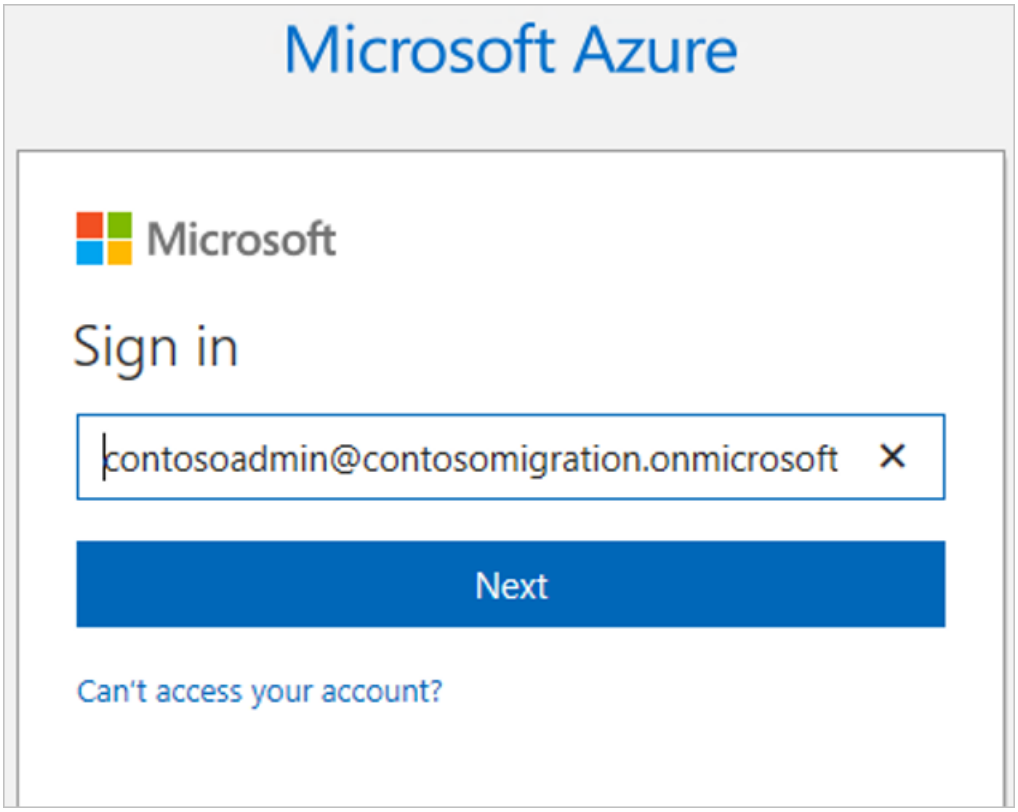
```
TfsMigrator prepare /collection:http://contosotfs:8080/tfs/ContosoDev
/tenantDomainName:contosomigration.onmicrosoft.com /accountRegion:cus
```



The preparation step does the following:

- Scans the collection to find a list of all users and then populates the identify map log (`IdentityMapLog.csv`).
- Prepares the connection to Azure AD to find a match for each identity.
- Contoso has already deployed Azure AD and synchronized it by using Azure AD Connect, so the prepare command should be able to find the matching identities and mark them as **Active**.

2. An Azure AD sign-in screen appears, and the admins enter the credentials of a global admin.



3. The preparation is completed, and the tool reports that the import files have been generated successfully.

```
-----------------------------------
  Generating Import Files
-----------------------------------

Collection validation successful.
Generating required import files...

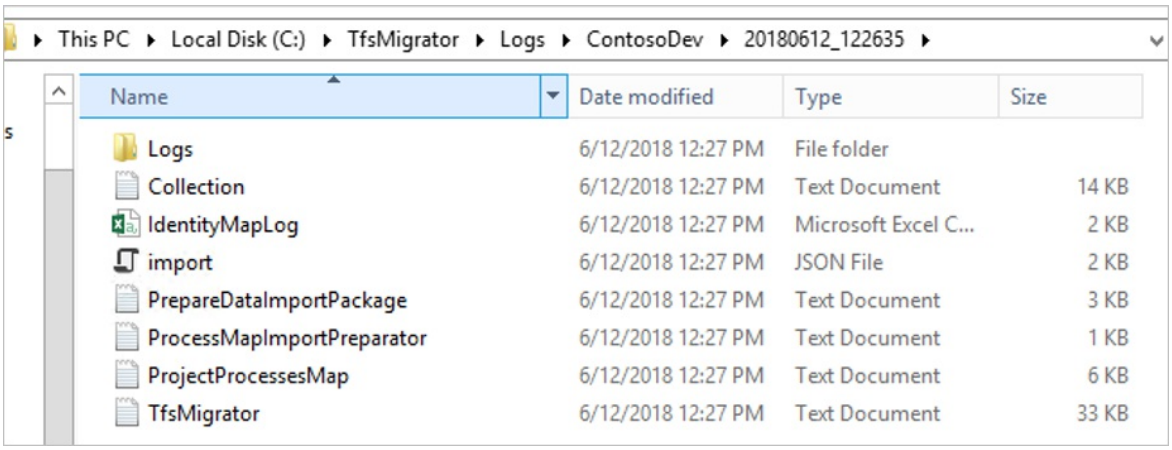
Prepare WorkItem Tracking Data Import    Passed
Prepare Import Package                   Passed

-----------------------------------
  Guidance
-----------------------------------

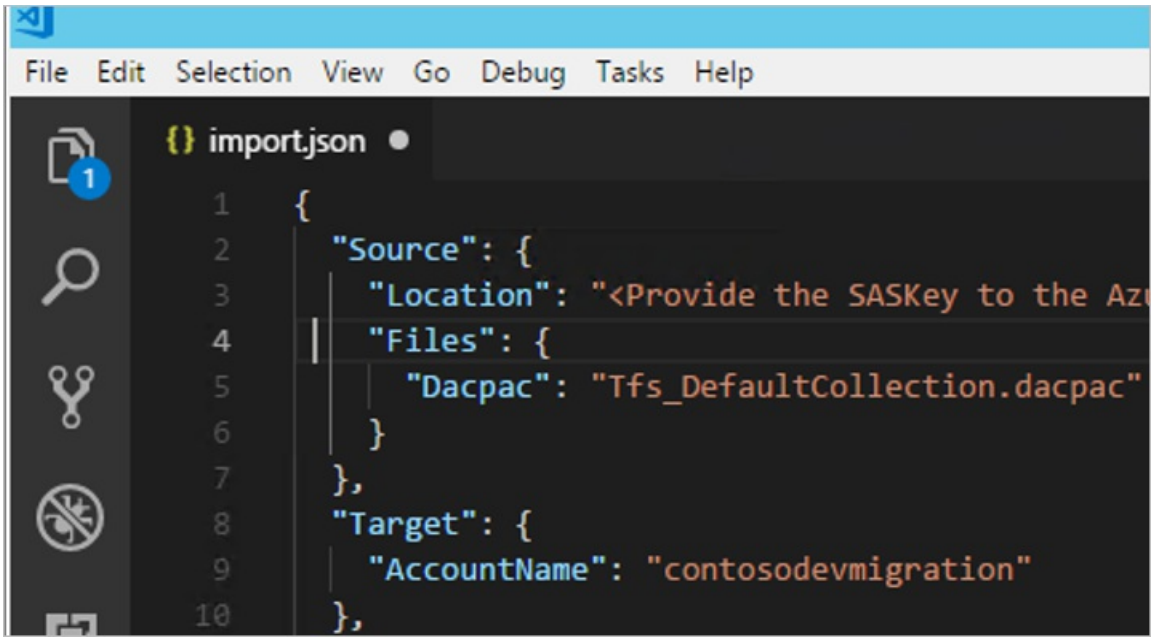
Now that the files are generated, recommended next steps are:

  1) Review each generated file and make changes or fill in blanks as needed
  2) See TfsMigrator validate /help for instructions on re-validating the files
  3) Once satisfied you'll need to ready the collection for import
```

4. The admins can now see that both the `IdentityMapLog.csv` file and the `import.json` file have been created in a new folder.



5. The `import.json` file provides import settings. It includes information such as the desired organization name, and storage account details. Most of the fields are populated automatically. Some fields require user input. The admins open the file and add the Azure DevOps Services organization name to be created, `contosodevmigration`. With this name, the Contoso Azure DevOps Services URL will be `contosodevmigration.visualstudio.com`.



> **NOTE**
> The organization must be created before the migration begins. It can be changed after the migration is completed.

6. The admins review the identity log map file, which shows the accounts that will be brought into Azure DevOps Services during the import.

   - Active identities refer to identities that will become users in Azure DevOps Services after the import.
   - In Azure DevOps Services, these identities will be licensed and displayed as users in the organization after migration.
   - The identities are marked as **Active** in the **Expected Import Status** column in the file.

	A	B	C	D
1	AD: User(TFS)	AD: Security Identifier	AAD: Expected Import User(VSTS)	Expected Import Status
2	CONTOSOTFS\Administrator	S-1-5-21-13773882-1227584077-4008836372-500	No Match Found (Check AAD Sync)	Historical
3	CONTOSO\Administrator	S-1-5-21-922643139-2624149876-687917293-500	No Match Found (Check AAD Sync)	Historical
4	CONTOSO\antf	S-1-5-21-922643139-2624149876-687917293-3946	antf@contosomigration.onmicrosoft.com	Active
5	CONTOSO\dand	S-1-5-21-922643139-2624149876-687917293-3945	dand@contosomigration.onmicrosoft.com	Active
6	CONTOSO\Frec	S-1-5-21-922643139-2624149876-687917293-3949	Frec@contosomigration.onmicrosoft.com	Active
7	CONTOSO\Kart	S-1-5-21-922643139-2624149876-687917293-3948	Kart@contosomigration.onmicrosoft.com	Active
8	CONTOSO\Mara	S-1-5-21-922643139-2624149876-687917293-3957	Mara@contosomigration.onmicrosoft.com	Active
9	CONTOSO\Rhoh	S-1-5-21-922643139-2624149876-687917293-3963	Rhoh@contosomigration.onmicrosoft.com	Active

Step 5: Migrate to Azure DevOps Services

With the preparation completed, Contoso admins can focus on the migration. After they run the migration, they'll switch from using TFVC to Git for version control.

Before they start, the admins schedule downtime with the dev team, so that they can plan to take the collection offline for migration.

Here is the migration process they'll follow:

1. **Detach the collection.** Identity data for the collection resides in the configuration database for the Team Foundation Server instance while the collection is attached and online.

   When a collection is detached from the Team Foundation Server instance, a copy of that identity data is made and then packaged with the collection for transport. Without this data, the identity portion of the import can't be executed.

   We recommended that the collection stay detached until the import has been completed, because changes that occur during the import can't be imported.

2. **Generate a backup.** The next step is to generate a backup that can be imported into Azure DevOps Services. The data-tier application component package (DACPAC) is a SQL Server feature that allows database changes to be packaged into a single file and then deployed to other instances of SQL.

   The backup can also be restored directly to Azure DevOps Services, and it's used as the packaging method for getting collection data to the cloud. Contoso will use the `sqlpackage.exe` tool to generate the DACPAC. This tool is included in SQL Server Data Tools.

3. **Upload to storage.** After the DACPAC is created, the admins upload it to Azure Storage. After they've uploaded it, they get a shared access signature (SAS) to allow the Team Foundation Server migration tool access to the storage.

4. **Fill out the import.** Contoso can then complete the missing fields in the import file, including the DACPAC setting. To ensure that everything's working properly before the full migration, the admins will specify that they want to perform a *dry-run* import.

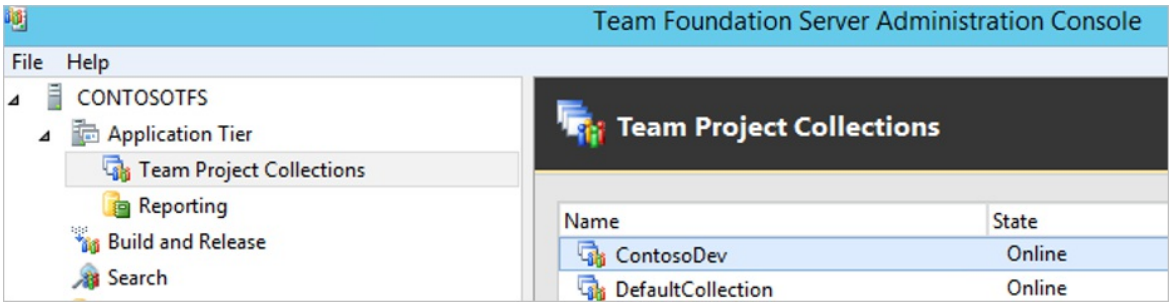
5. **Perform a dry-run import.** A dry-run import helps them test the collection migration. Dry runs have a limited life, so they're deleted before a production migration runs. They're deleted automatically after a set duration. A note that informs Contoso when the dry run will be deleted is included in the success email that's sent after the import finishes. The team takes note and plans accordingly.

6. **Complete the production migration.** With the dry-run migration completed, Contoso admins do the final migration by updating the `import.json` file and then running import again.

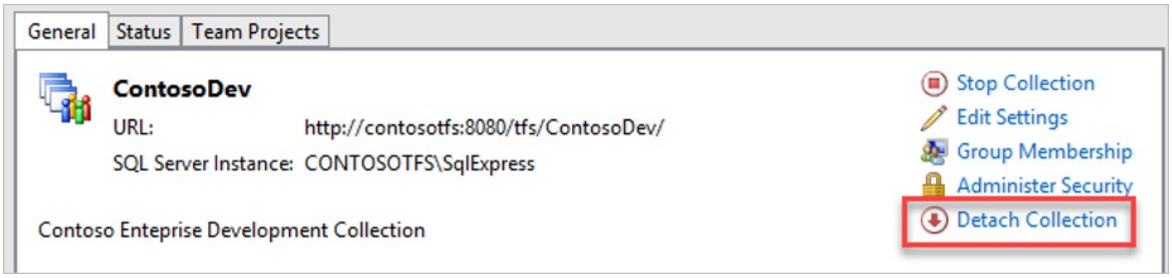
Detach the collection

Before they detach the collection, Contoso admins take a local SQL Server instance backup and a VMware snapshot of the Team Foundation Server instance.

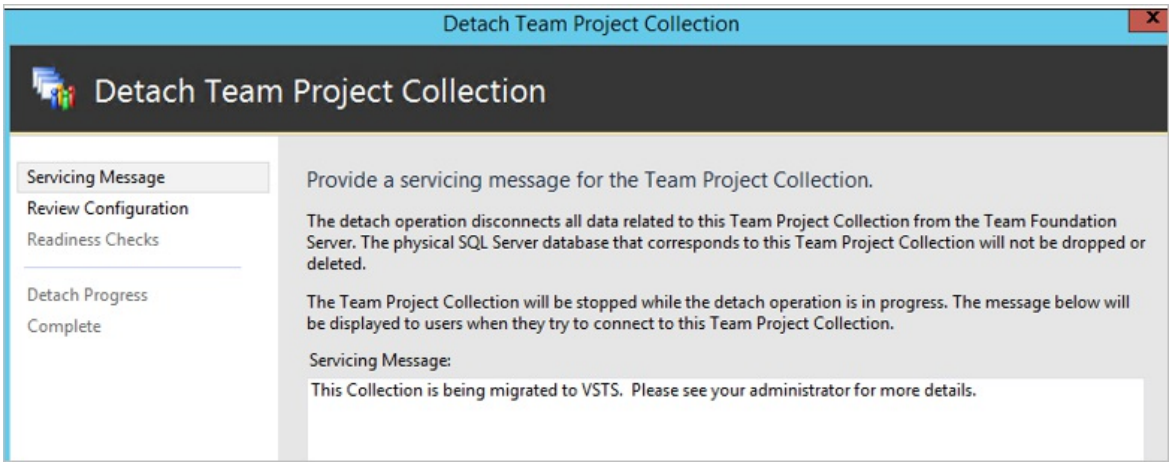
1. In the Team Foundation Server Administration Console, the admins select the collection they want to detach, **ContosoDev**.



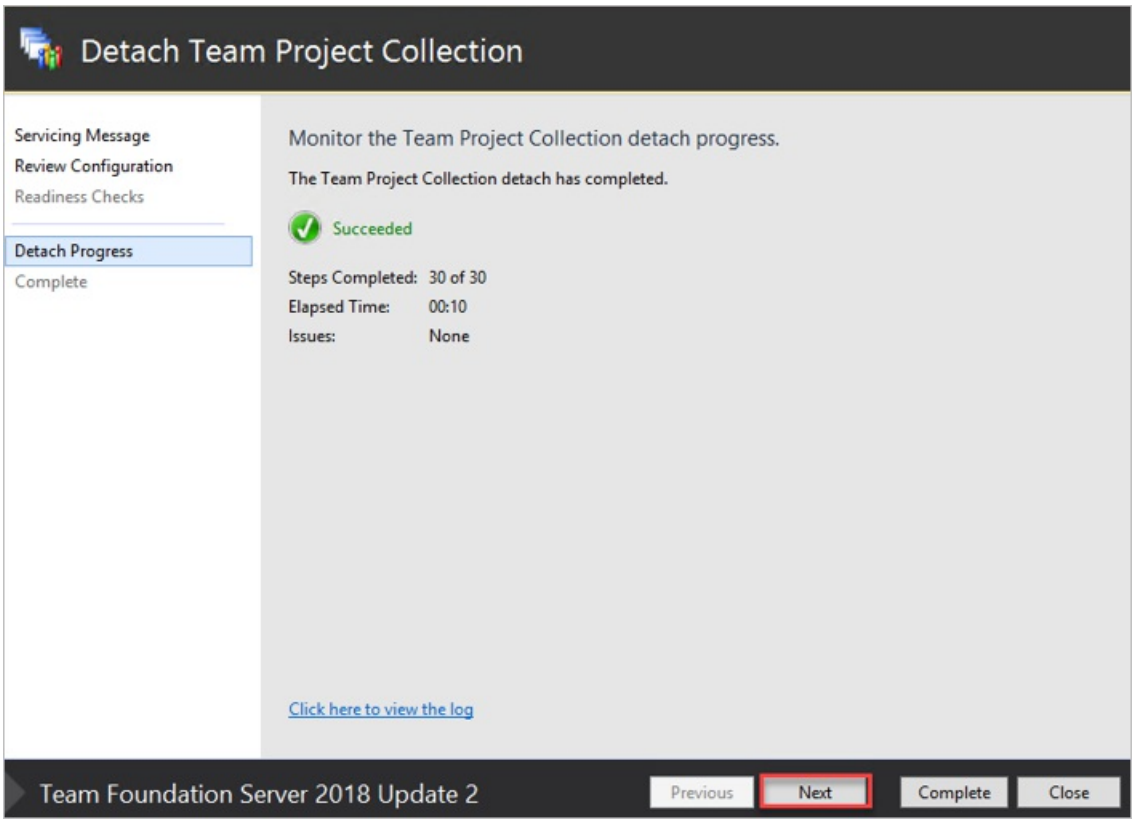
2. They select the **General** tab and then select **Detach Collection**.



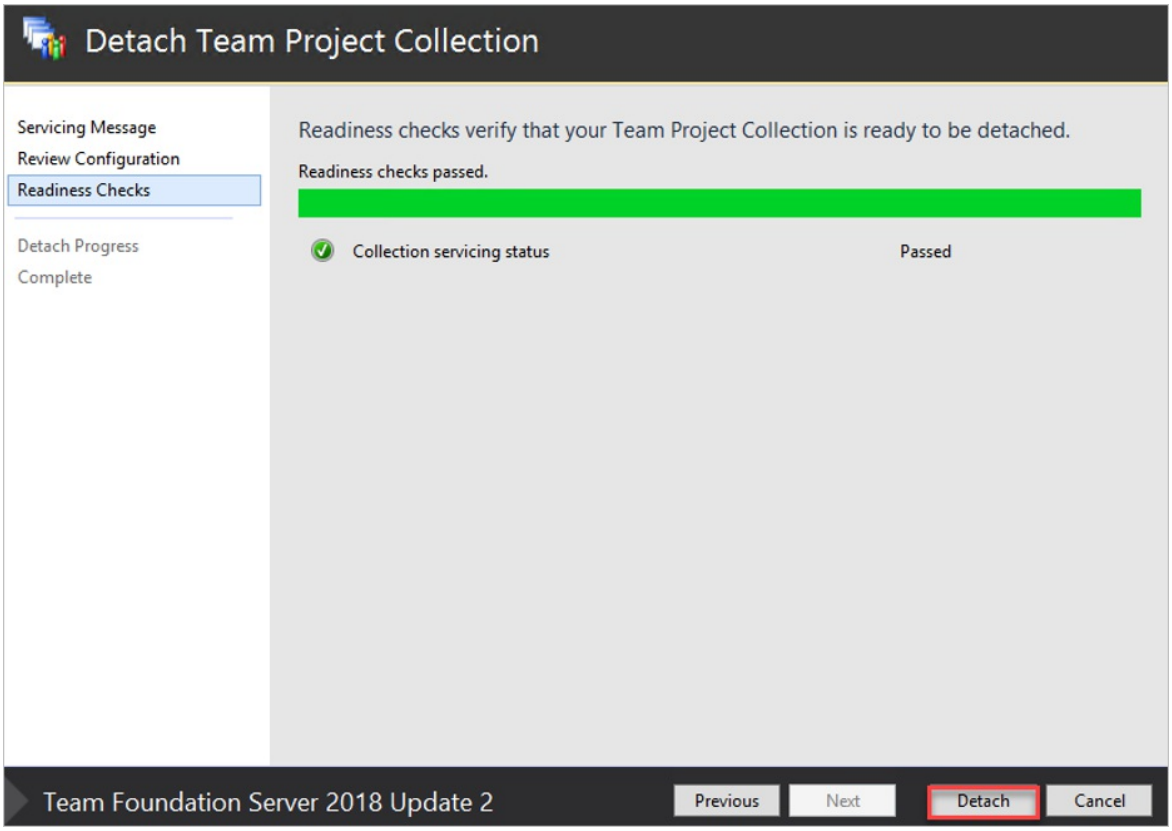
3. In the **Detach Team Project Collection** wizard, on the **Servicing Message** pane, the admins provide a message for users who might try to connect to projects in the collection.



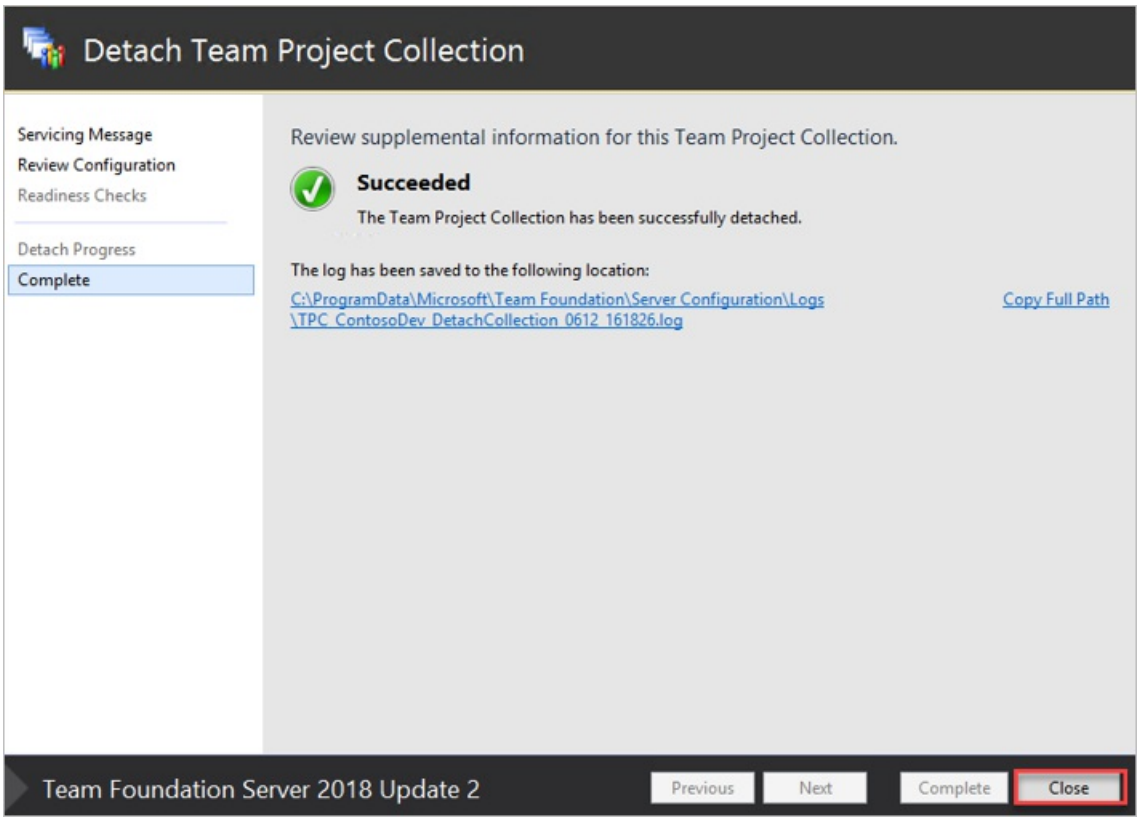
4. On the **Detach Progress** pane, they monitor progress. When the process finishes, they select **Next**.



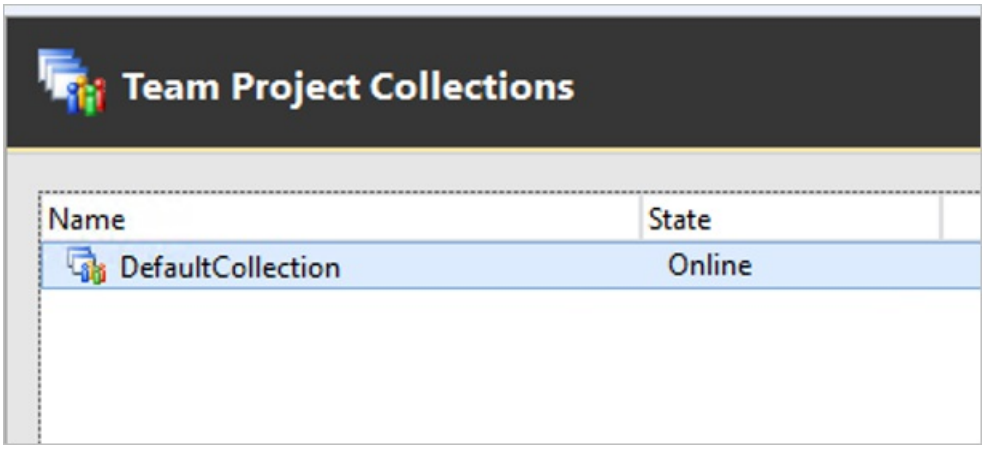
5. On the **Readiness Checks** pane, when the checks finish, they select **Detach**.



6. When the collection has been successfully detached, they select **Close** to finish up.



The collection is no longer referenced in the Team Foundation Server Administration Console.



Generate a DACPAC

Contoso admins create a backup, or DACPAC, to import into Azure DevOps Services.

- The admins use the `sqlpackage.exe` utility in SQL Server Data Tools (SSDT) to create the DACPAC. There are multiple versions of `sqlpackage.exe` installed with SQL Server Data Tools, and they're located under folders with names like `120`, `130`, and `140`. It's important to use the right version to prepare the DACPAC.
- Team Foundation Server 2018 imports need to use `sqlpackage.exe` from the *140* folder or higher. For `CONTOSOTFS`, this file is located in

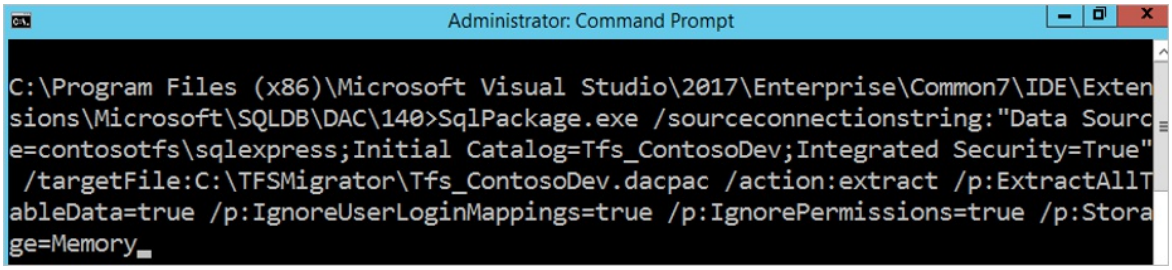
  ```
  C:\Program Files (x86)\Microsoft Visual Studio\2017\Enterprise\Common7\IDE\Extensions\Microsoft\SQLDB\DAC\140
  ```

  .

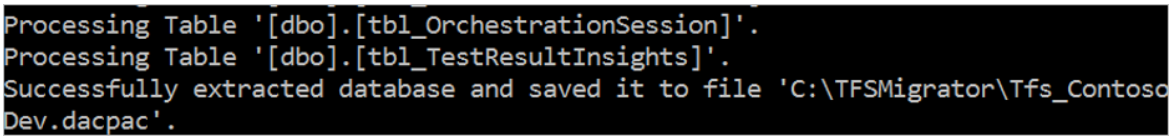
Contoso admins generate the DACPAC as follows:

1. They open a command prompt and go to the `sqlpackage.exe` location. To generate the DACPAC, they run the following command:

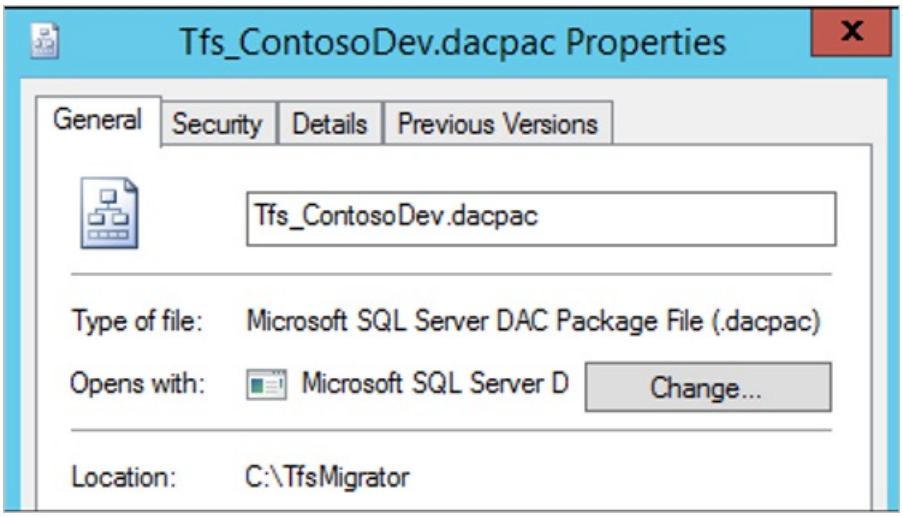
```
SqlPackage.exe /sourceconnectionstring:"Data Source=SQLSERVERNAME\INSTANCENAME;Initial
Catalog=Tfs_ContosoDev;Integrated Security=True" /targetFile:C:\TFSMigrator\Tfs_ContosoDev.dacpac
/action:extract /p:ExtractAllTableData=true /p:IgnoreUserLoginMappings=true /p:IgnorePermissions=true
/p:Storage=Memory
```



The following message is displayed:



2. They verify the properties of the DACPAC file.



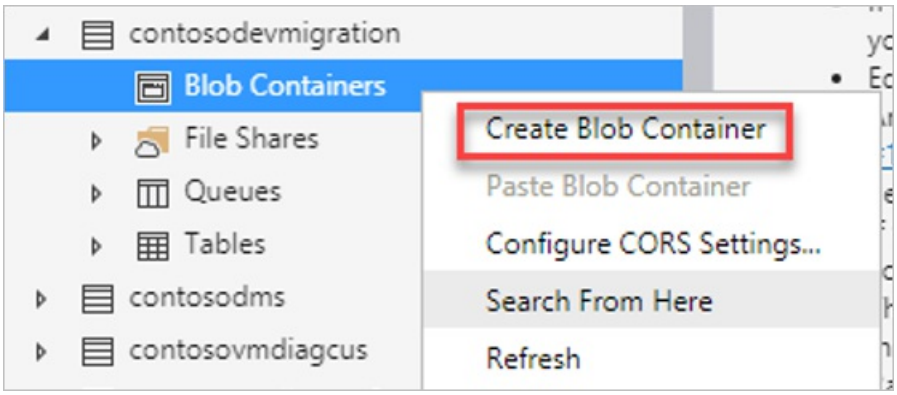
Upload the file to storage

After the admins create the DACPAC file, they upload it to the Azure Storage account.

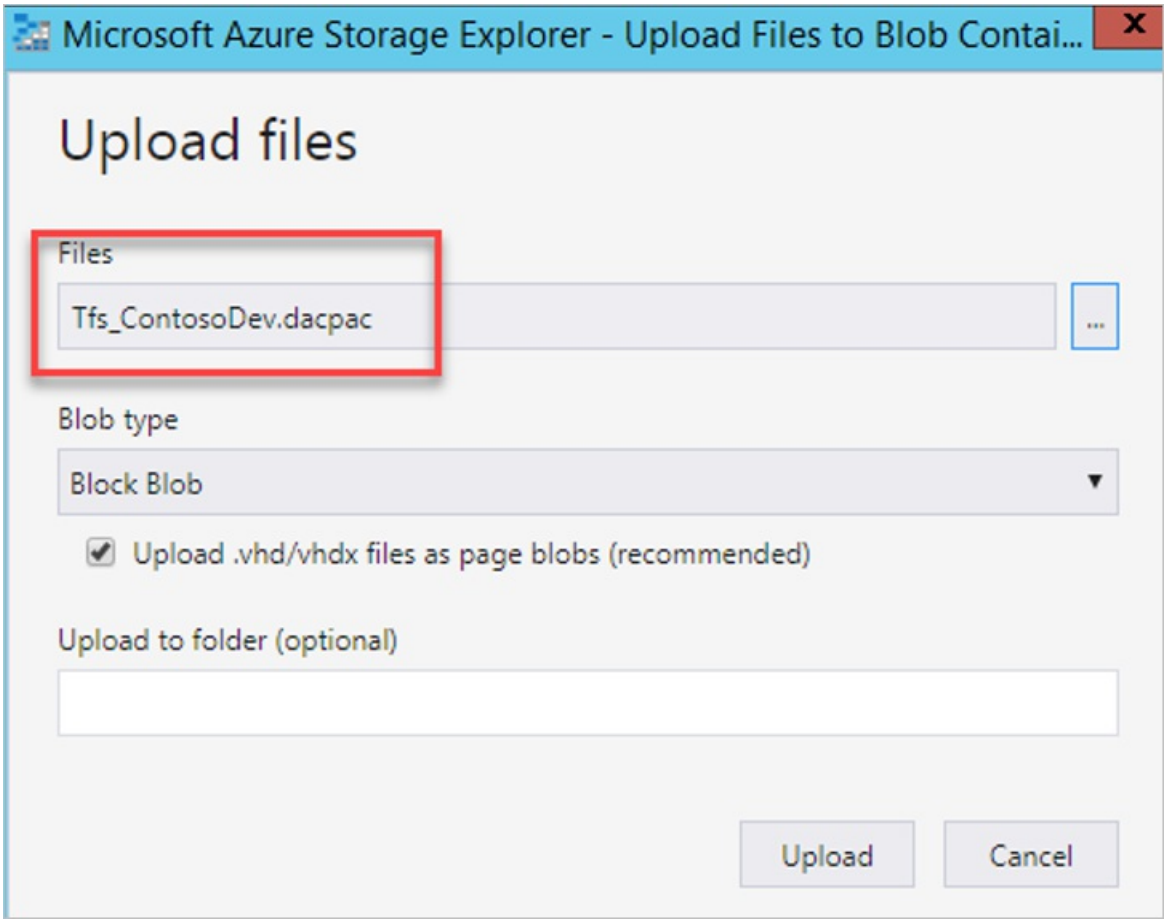
1. They download and install Azure Storage Explorer.



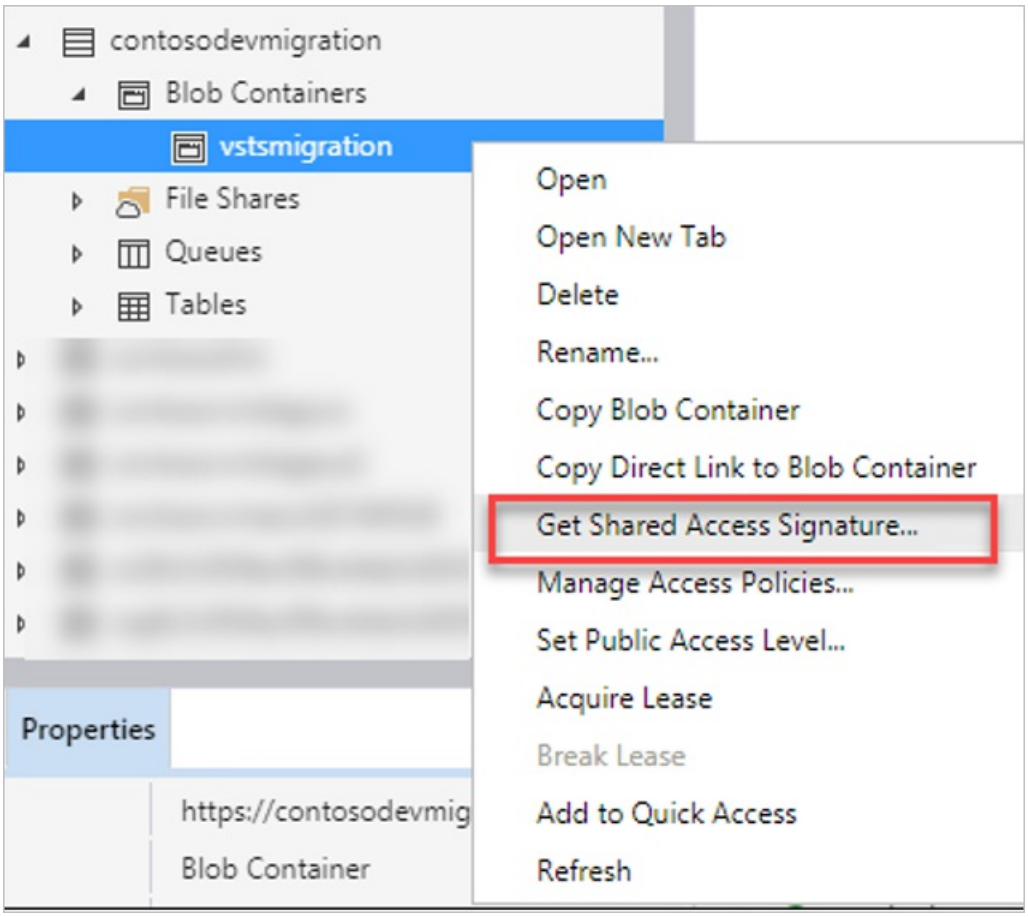
2. In Storage Explorer, the admins connect to their subscription and then search for and select the storage account they created for the migration (`contosodevmigration`). They create a new blob container, `azuredevopsmigration`.



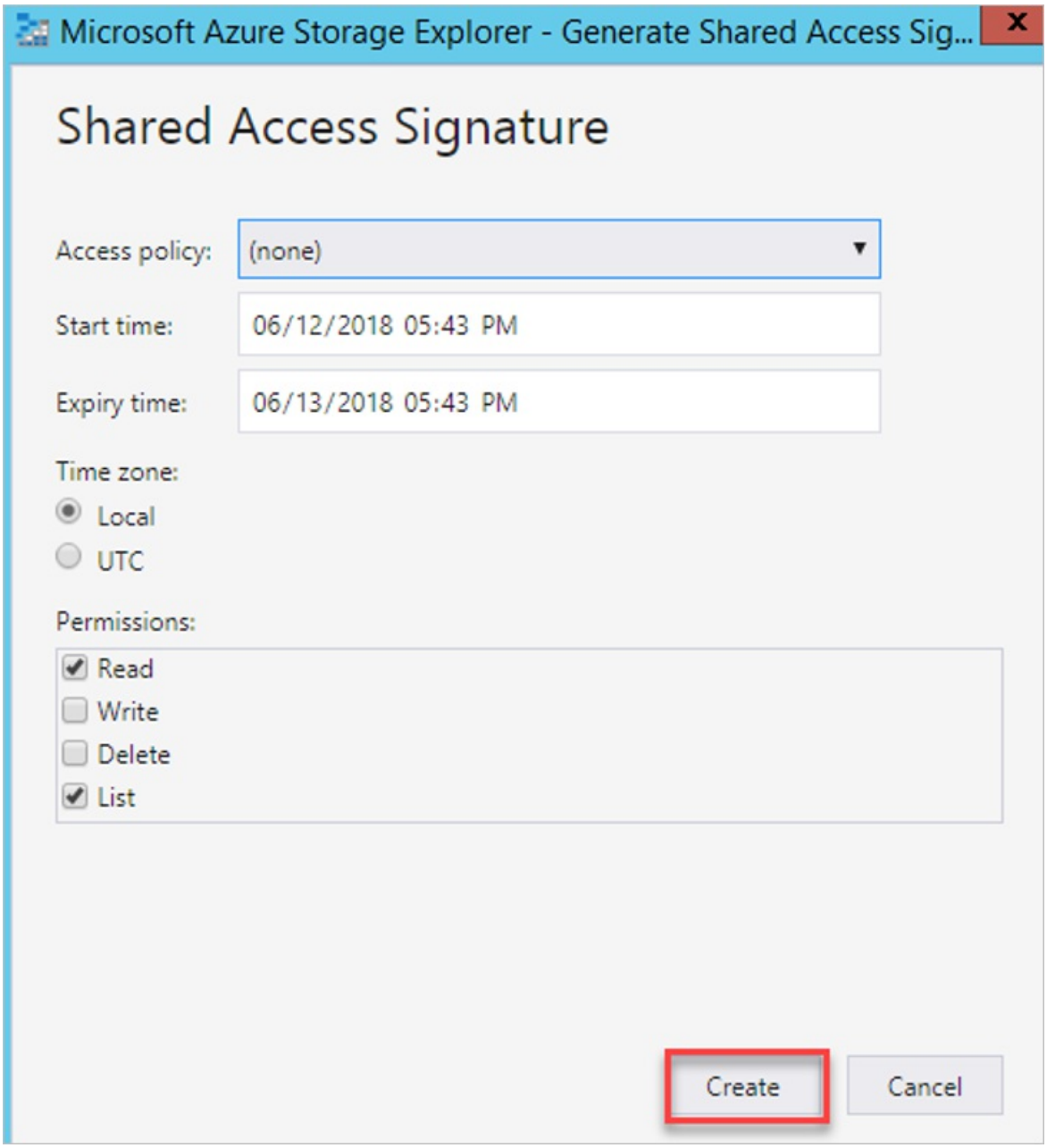
3. On the **Upload files** pane, in the **Blob type** drop-down list, the admins specify **Block Blob** for the DACPAC file upload.



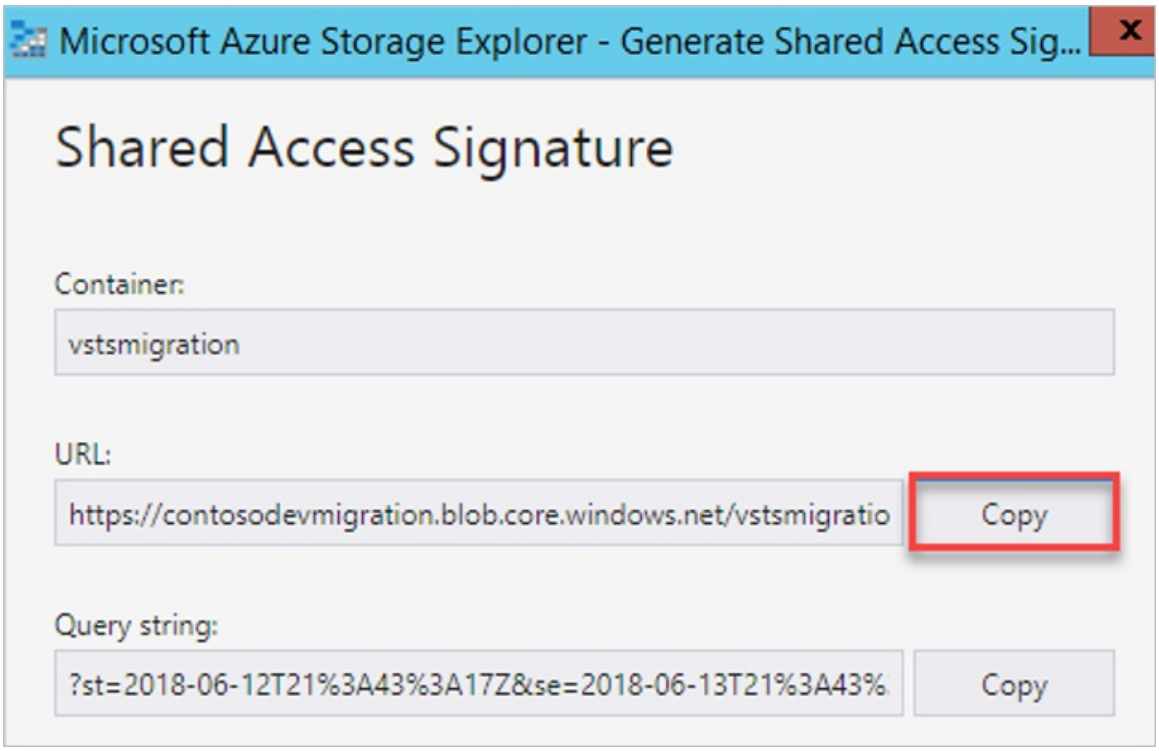
4. After they upload the file, they select the file name and then select **Generate SAS**. They expand the **Blob Containers** list under the storage account, select the container with the import files, and then select **Get Shared Access Signature**.



5. On the **Shared Access Signature** pane, they accept the default settings and then select **Create**. This enables access for 24 hours.



6. They copy the shared access signature URL, so that it can be used by the Team Foundation Server migration tool.



> **NOTE**
>
> The migration must happen within the allowed time window or the permissions will expire. Do *not* generate an SAS key from the Azure portal. Keys that are generated from the portal are account-scoped and won't work with the import.

Fill in the import settings

Earlier, Contoso admins partially filled in the import specification file, `import.json`. Now, they need to add the remaining settings.

They open the `import.json` file and complete the following fields:

- **Location:** They enter the location of the SAS key that was generated previously.
- **DACPAC:** They enter the name of the DACPAC file that they uploaded earlier to the storage account, making sure to include the *.dacpac* extension.
- **ImportType:** They enter **DryRun** for now.



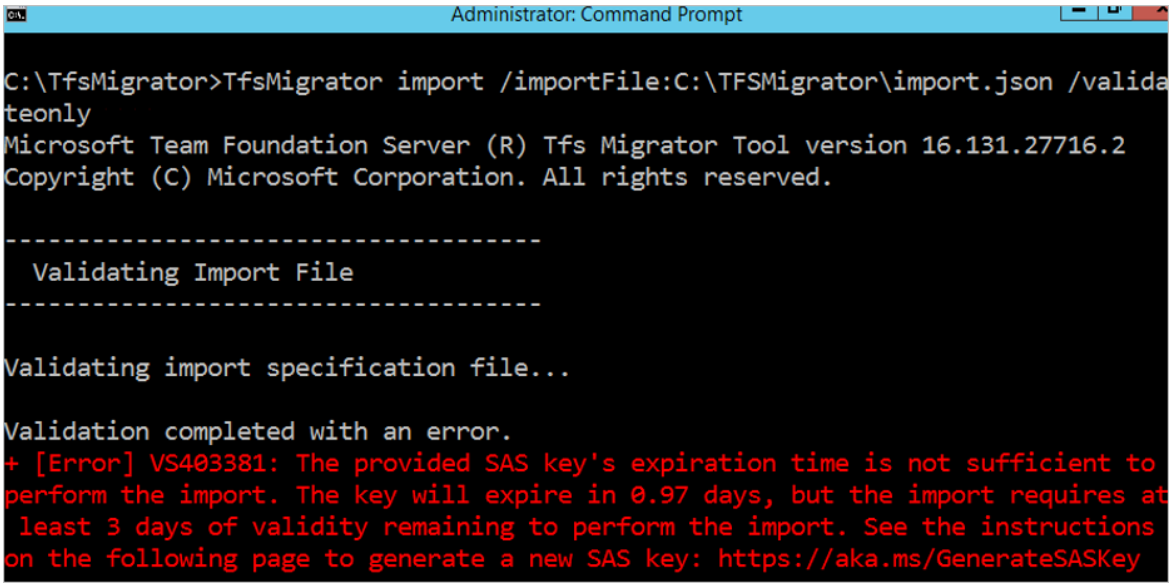
Perform a dry-run migration

Contoso admins perform a dry-run migration to make sure that everything's working as expected.

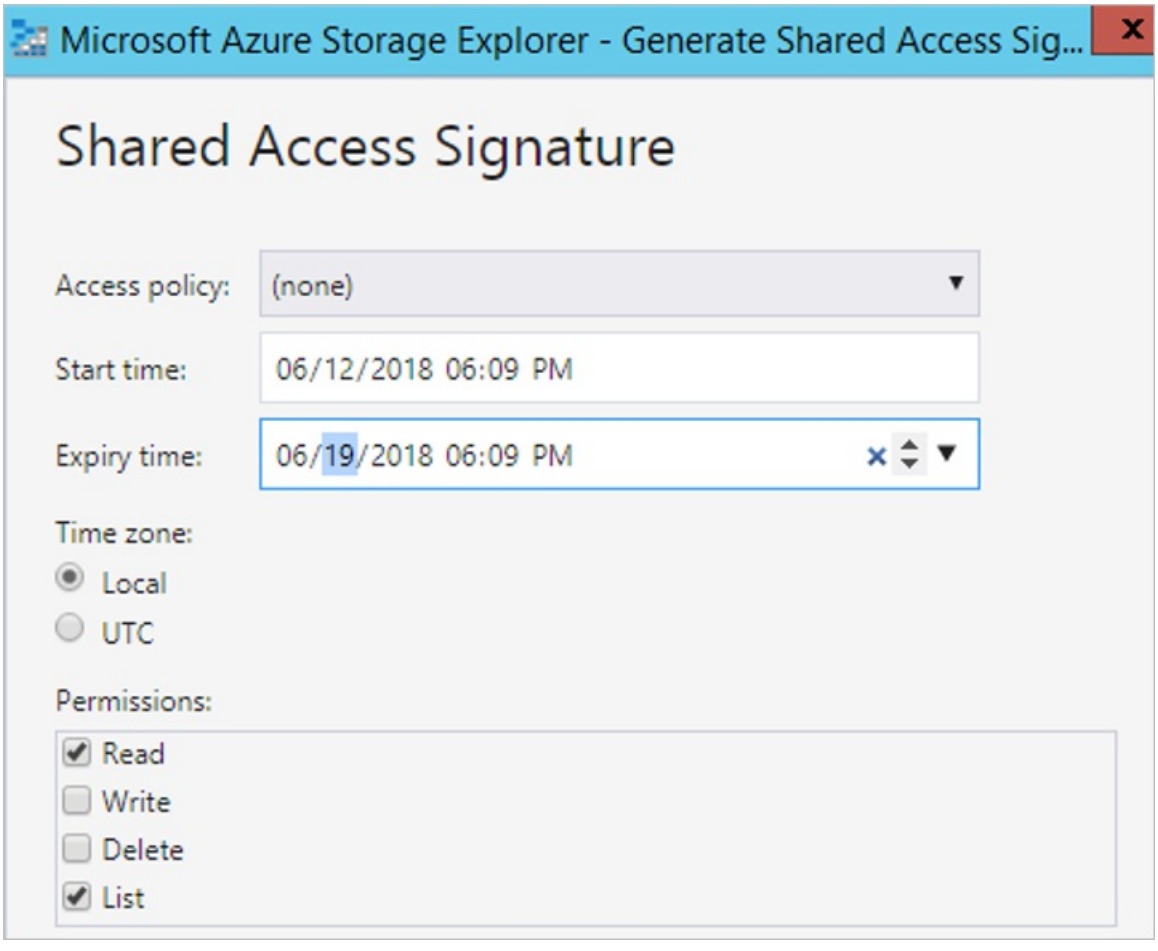
1. They open a command prompt and then go to the `TfsMigrator` location (`C:\TFSMigrator`).
2. They want to make sure that the file is formatted properly and that the SAS key is working. They validate the import file by running the following command:

   ```
   TfsMigrator import /importFile:C:\TFSMigrator\import.json /validateonly
   ```

   The validation returns an error saying that the SAS key needs a longer period before it expires.



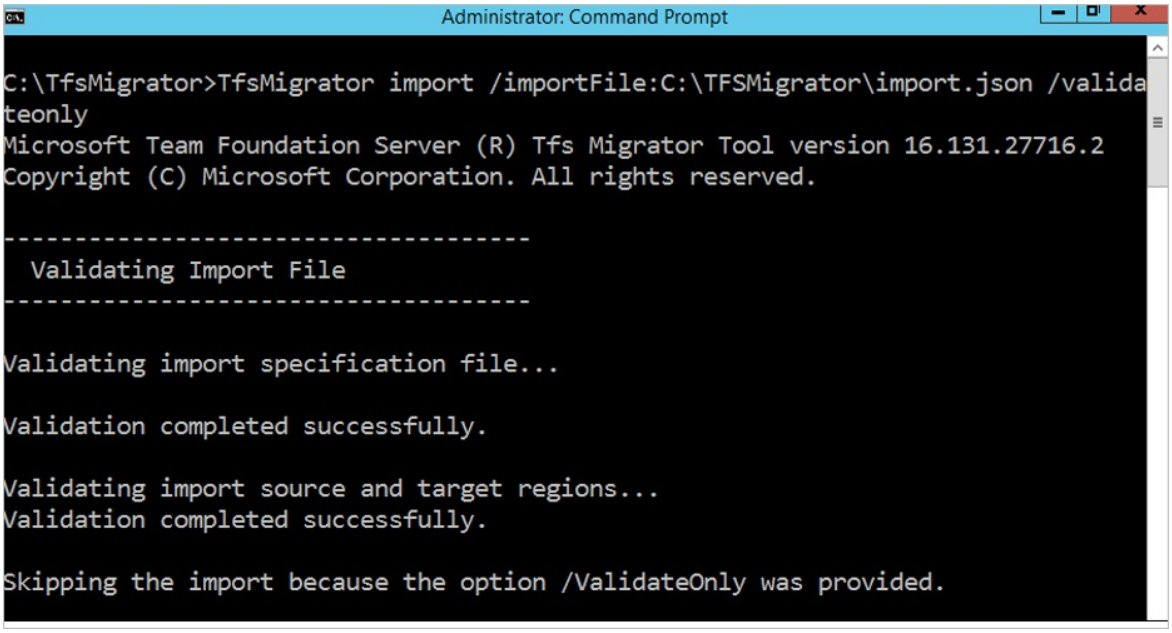
3. They use Azure Storage Explorer to create a new SAS key with the period before expiration set to seven days.



4. They update the `import.json` file and rerun the command. This time, the validation is completed

successfully.

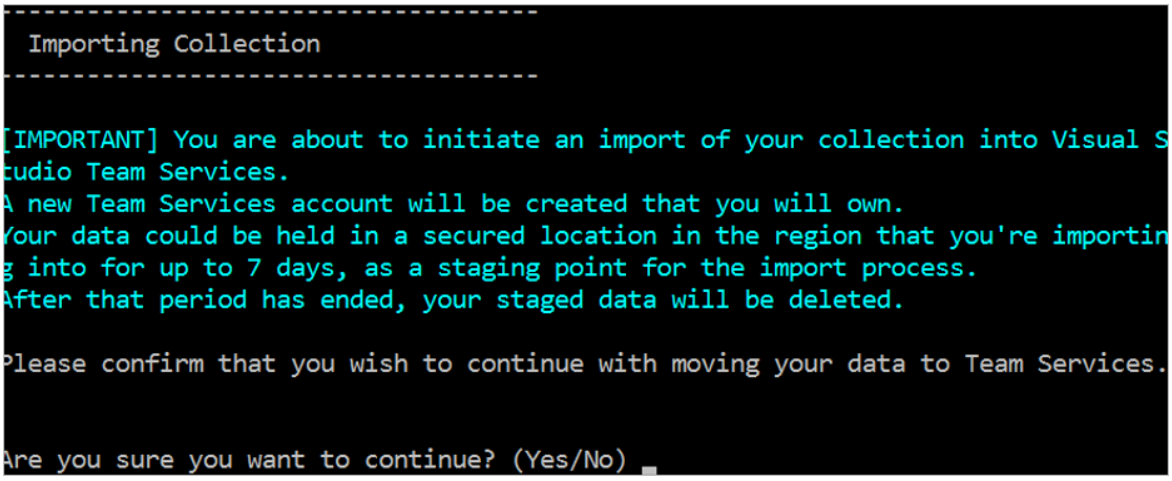
```
TfsMigrator import /importFile:C:\TFSMigrator\import.json /validateonly
```



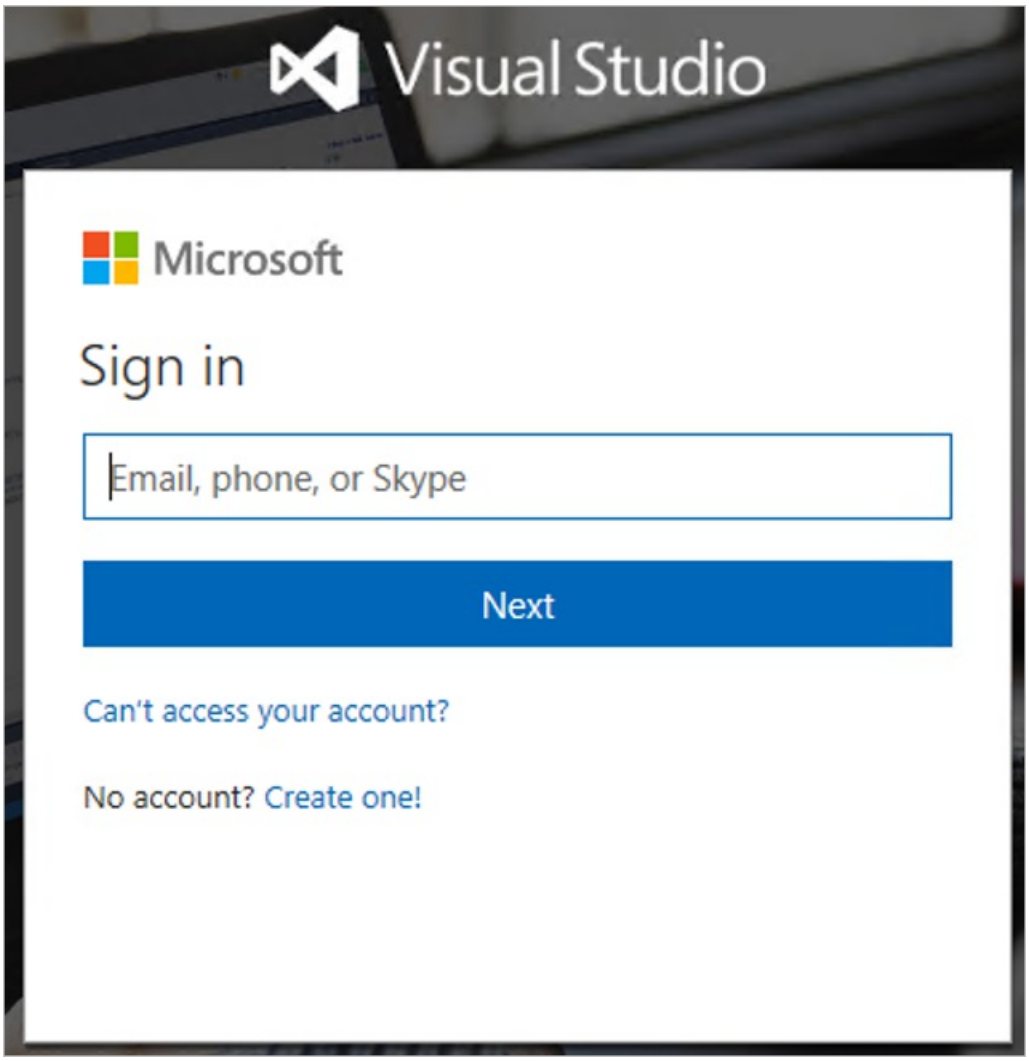
5. They start the dry run by running the following command:

   ```
   TfsMigrator import /importFile:C:\TFSMigrator\import.json
   ```

   A message is displayed asking them to confirm that they want to continue with the migration. Note the seven-day period after the dry run during which the staged data will be maintained.



6. The Azure AD sign-in window opens. Contoso admins sign in to Azure AD with admin permissions.



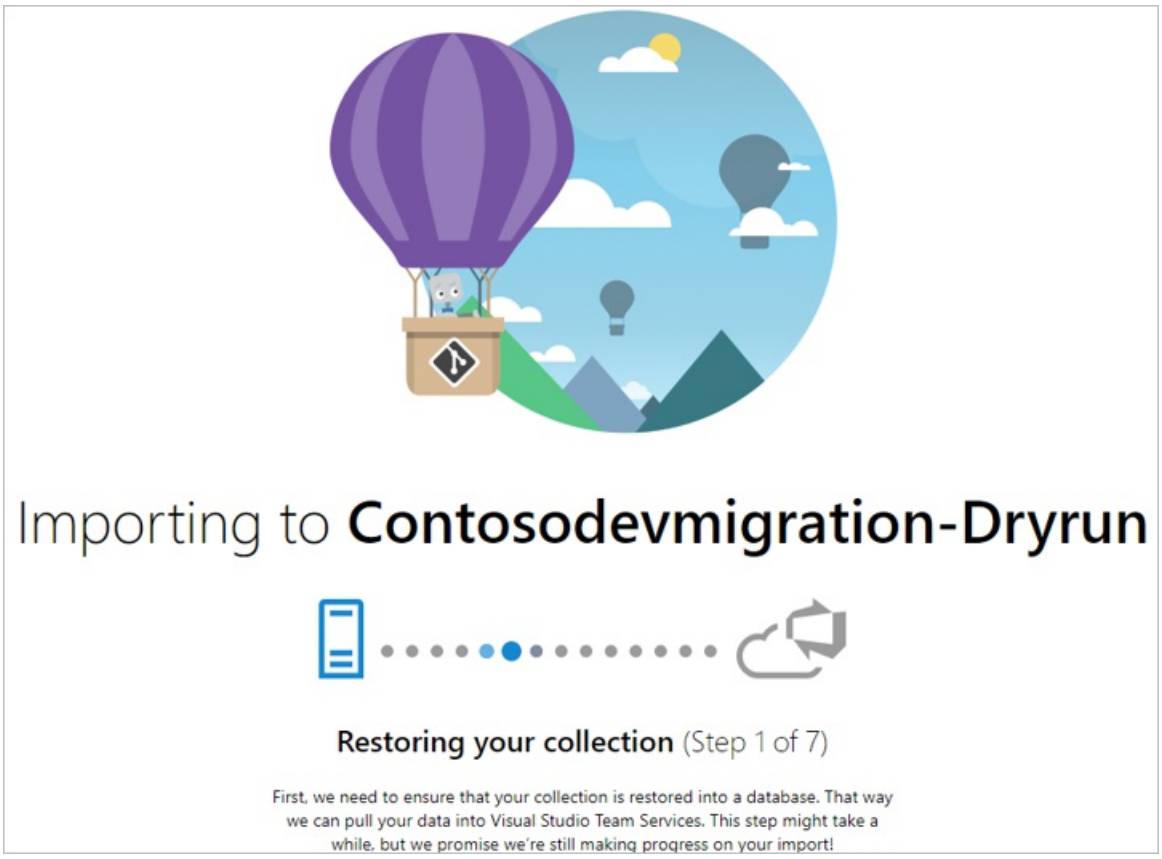
A message is displayed confirming that the import has been started successfully.

```
Starting the Import ...

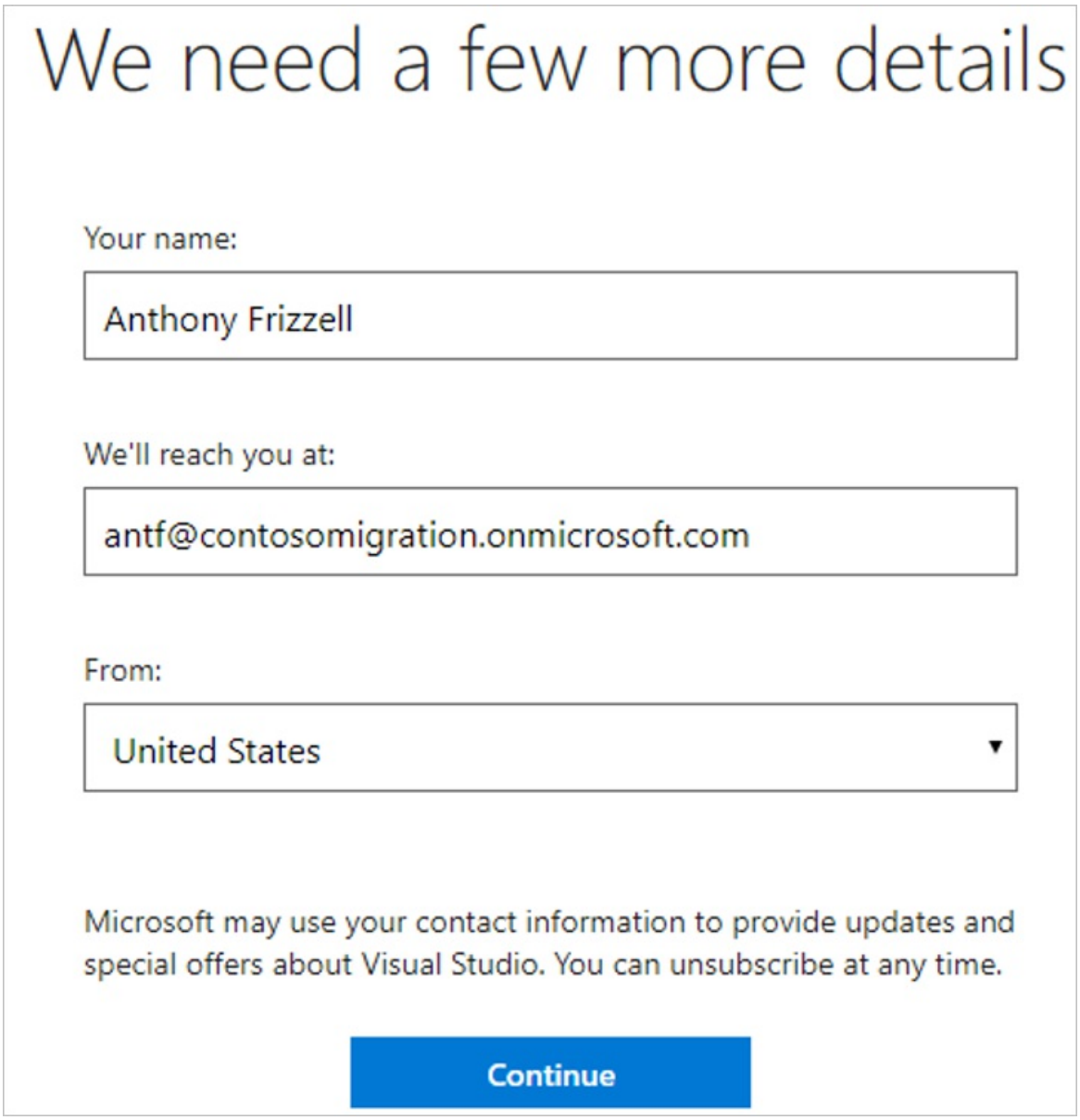
Import has been successfully started!

Monitor import: https://contosodevmigration-dryrun.visualstudio.com
Import ID: b954df07-4965-42c6-8439-
```

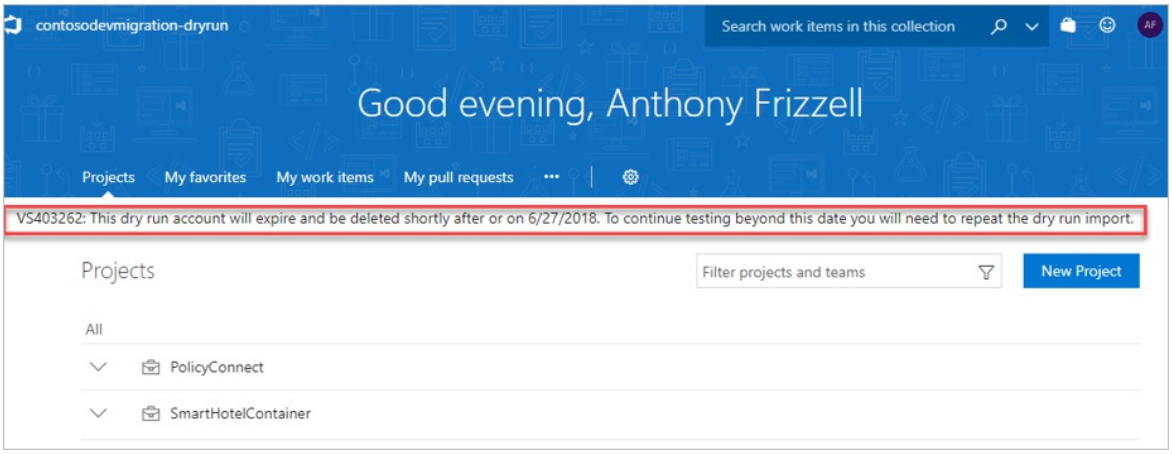
7. After about 15 minutes, the admins go to the website and see the following information:



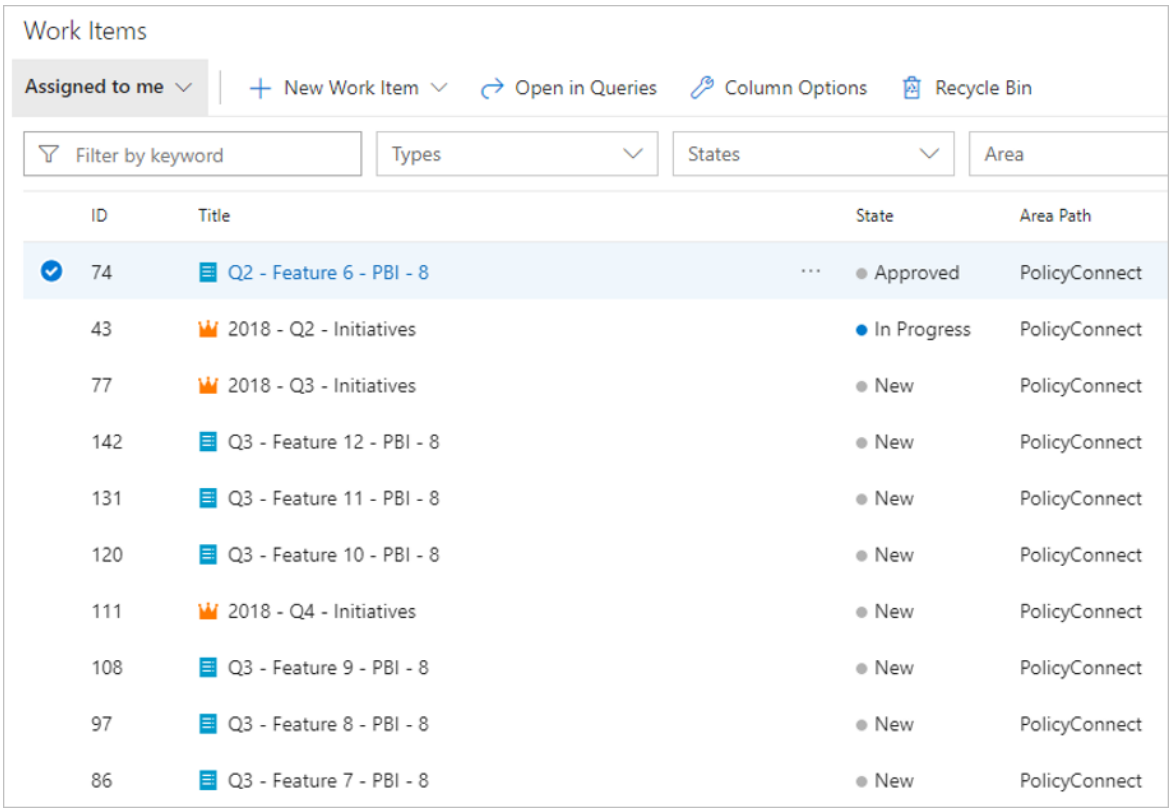
8. After the migration finishes, a Contoso dev lead signs in to Azure DevOps Services to ensure that the dry run worked properly. After authentication, Azure DevOps Services needs a few details to confirm the organization.



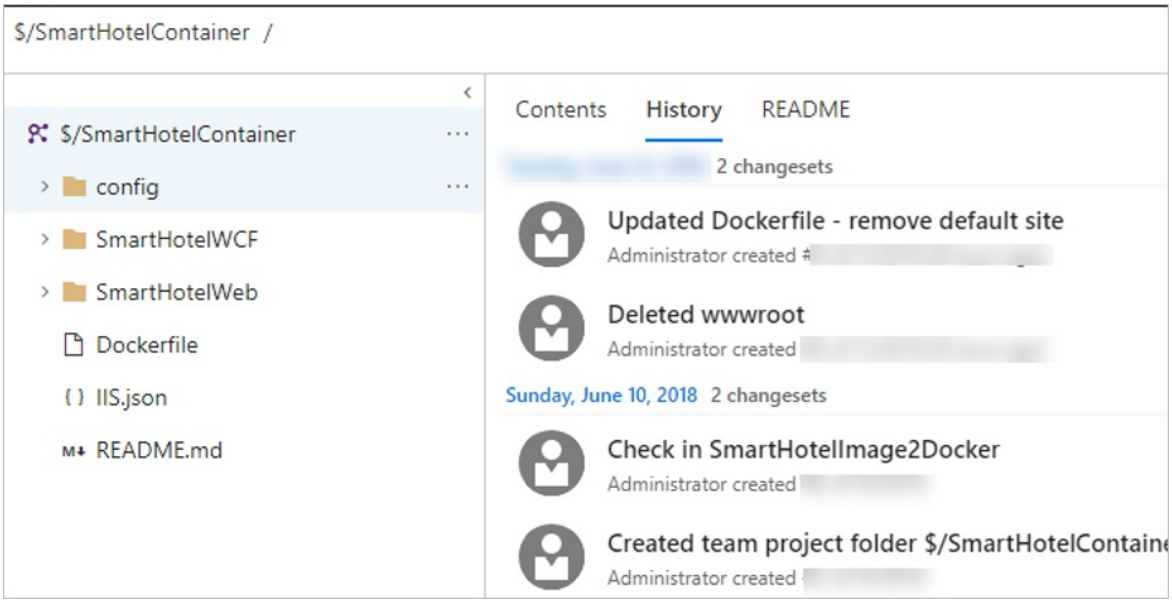
The dev lead can see that the projects have been migrated successfully. A notice near the top of the page warns that the dry run account will be deleted in 15 days.



9. The dev lead opens one of the projects and then selects **Work Items** > **Assigned to me**. This page verifies that the work item data has been migrated successfully, along with the identity.



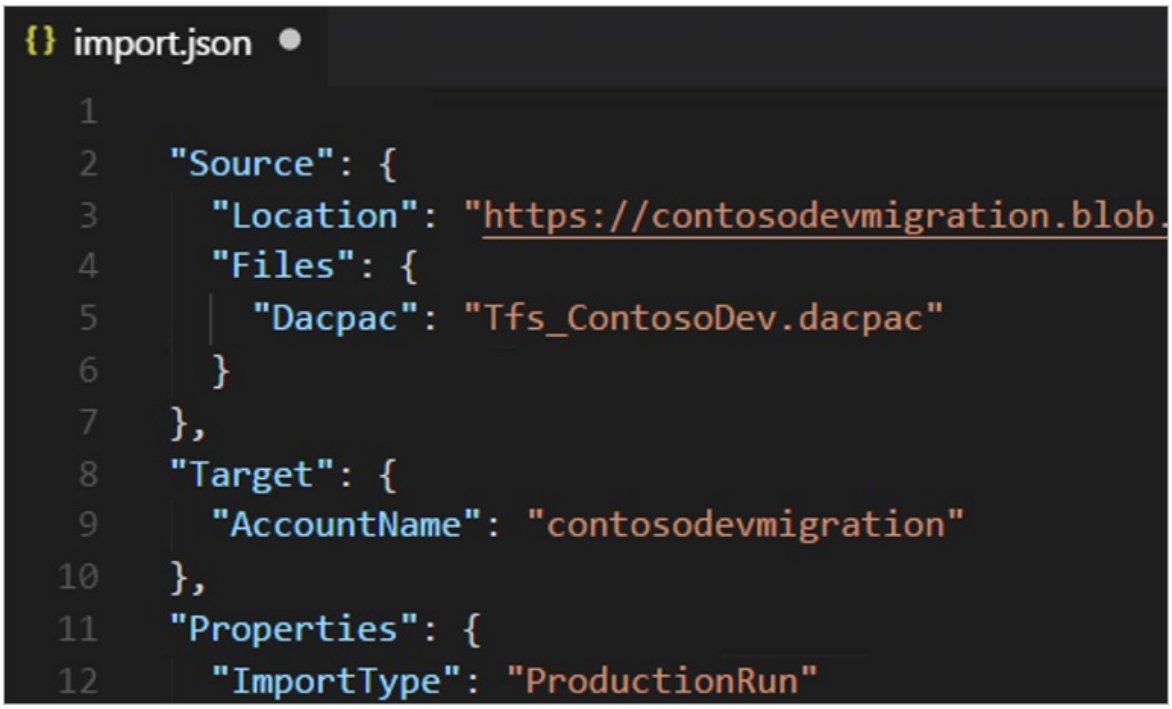
10. To confirm that the source code and history have been migrated, the dev lead checks other projects and code.



Run the production migration

Now that the dry run is complete, Contoso admins move on to the production migration. They delete the dry run, update the import settings, and run import again.

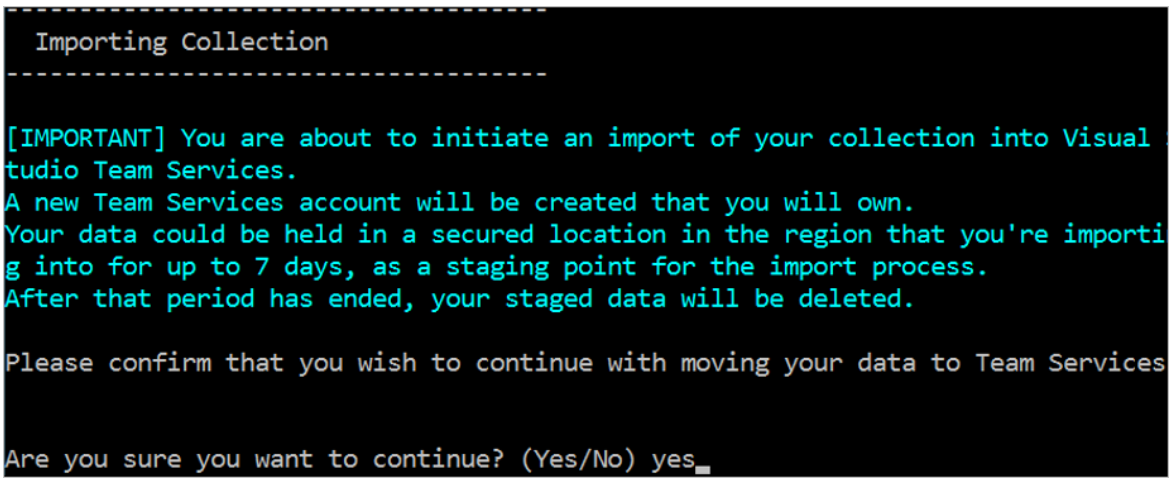
1. In the Azure DevOps Services portal, they delete the dry-run organization.
2. They update the `import.json` file to set the **ImportType** to **ProductionRun**.



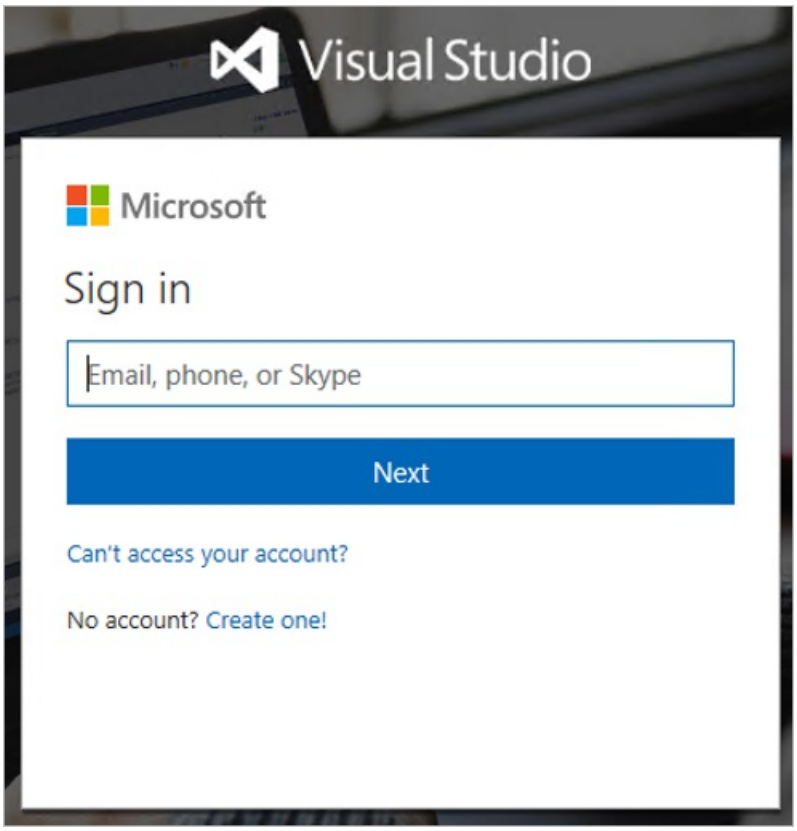
3. As they did for the dry run, they start the migration by running the following command:

   `TfsMigrator import /importFile:C:\TFSMigrator\import.json`.

   A message is displayed asking the admins to confirm the migration. It warns that data could be held in a secure location as a staging area for up to seven days.



4. In the Azure AD sign-in window, they specify a Contoso admin sign-in.



A message is displayed that the import has started successfully.

```
Starting the Import ...

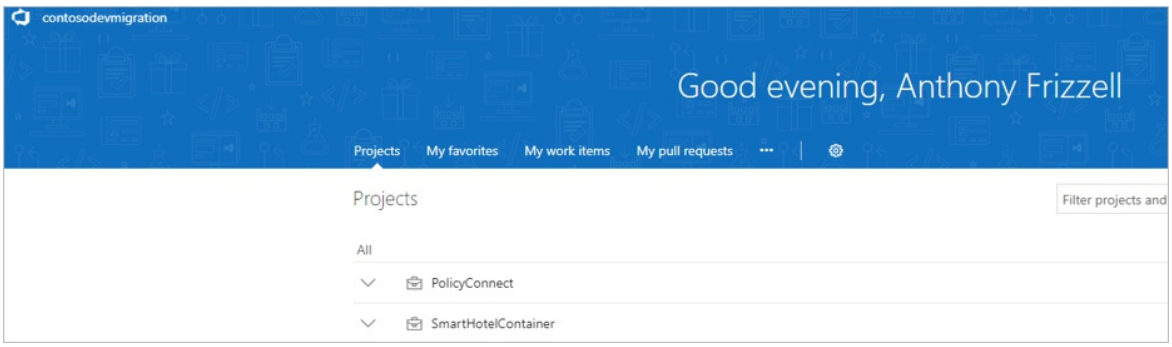
Import has been successfully started!

Monitor import: https://contosodevmigration.visualstudio.com
Import ID: 715a1aa1-82f0-45ca-843d-a0a0e8720388
```

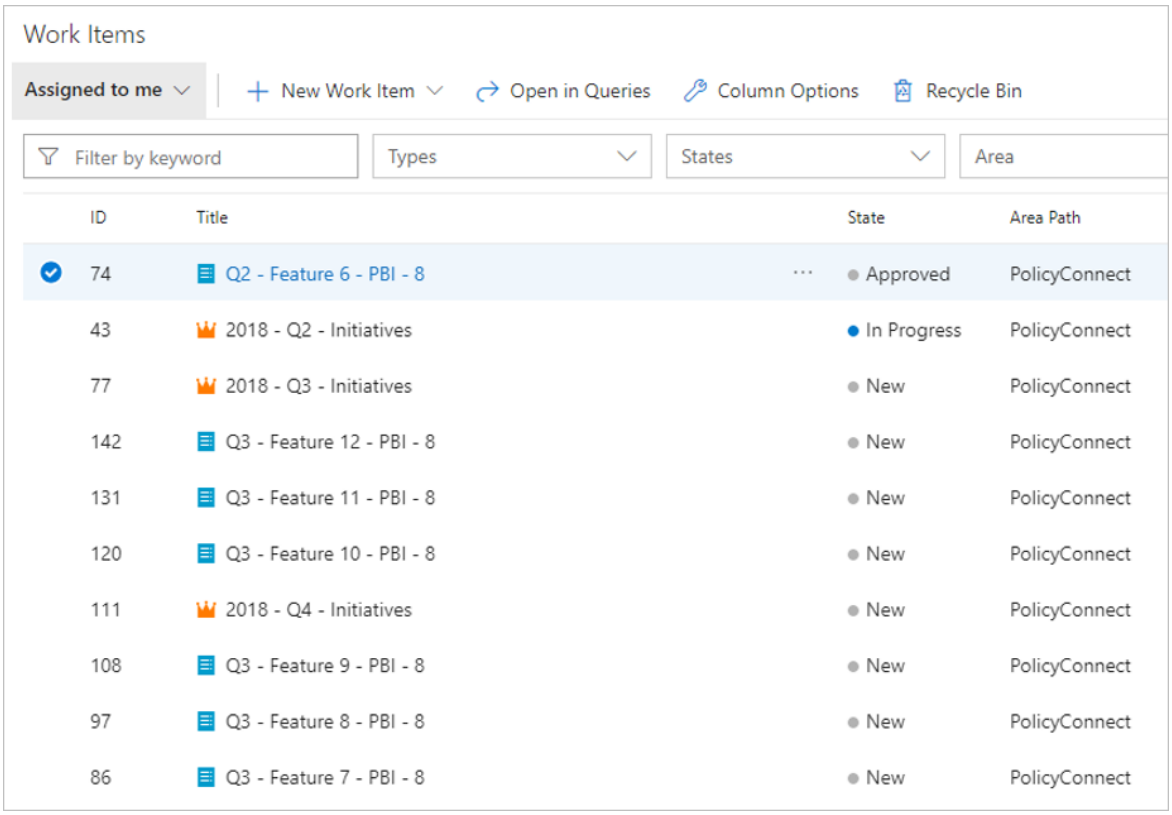
5. After about 15 minutes, the admins go to the website and see the following information:



6. After the migration finishes, a dev lead signs into Azure DevOps Services to ensure that the migration worked properly. After signing in, the dev lead can see that projects have been migrated.



7. The dev lead opens one of the projects and selects **Work Items** > **Assigned to me**. This shows that the work item data has been migrated, along with the identity.

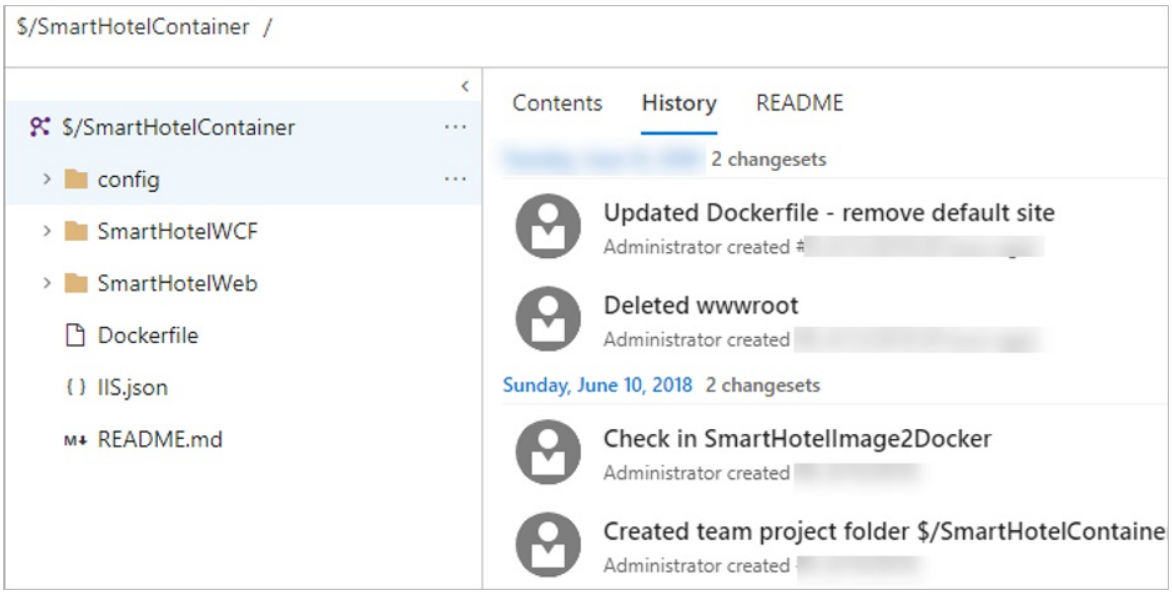


ID	Title	State	Area Path
74	Q2 - Feature 6 - PBI - 8	Approved	PolicyConnect
43	2018 - Q2 - Initiatives	In Progress	PolicyConnect
77	2018 - Q3 - Initiatives	New	PolicyConnect
142	Q3 - Feature 12 - PBI - 8	New	PolicyConnect
131	Q3 - Feature 11 - PBI - 8	New	PolicyConnect
120	Q3 - Feature 10 - PBI - 8	New	PolicyConnect
111	2018 - Q4 - Initiatives	New	PolicyConnect
108	Q3 - Feature 9 - PBI - 8	New	PolicyConnect
97	Q3 - Feature 8 - PBI - 8	New	PolicyConnect
86	Q3 - Feature 7 - PBI - 8	New	PolicyConnect

8. The dev lead checks to confirm that other work item data has been migrated.

Order	Work Item Type	Title	State	Effort	Assigned To
1	Epic	2018 - Q2 - Initiatives	In Progress		Anthony Frizzell
	Feature	Q2 - Feature 4	Done		Dan Drayton
	Product Backl...	Q2 - Feature 4 - PBI - 1	Done		Freda Conley
	Product Backl...	Q2 - Feature 4 - PBI - 2	Done		Freda Conley
	Product Backl...	Q2 - Feature 4 - PBI - 3	Done		Margo Ayers
	Product Backl...	Q2 - Feature 4 - PBI - 4	Done		Rhonda Hughes
	Product Backl...	Q2 - Feature 4 - PBI - 5	Done		Margo Ayers
	Product Backl...	Q2 - Feature 4 - PBI - 6	Done		Margo Ayers
	Product Backl...	Q2 - Feature 4 - PBI - 7	Done		Dan Drayton
	Product Backl...	Q2 - Feature 4 - PBI - 8	Done		Anthony Frizzell
	Product Backl...	Q2 - Feature 4 - PBI - 9	Done		Kari Tran
	Product Backl...	Q2 - Feature 4 - PBI - 10	Done		Kari Tran
	Feature	Q2 - Feature 5	Done		Dan Drayton
	Feature	Q2 - Feature 6	In Progress		Dan Drayton
2	Epic	2018 - Q3 - Initiatives	New		Anthony Frizzell
3	Epic	2018 - Q4 - Initiatives	New		Anthony Frizzell

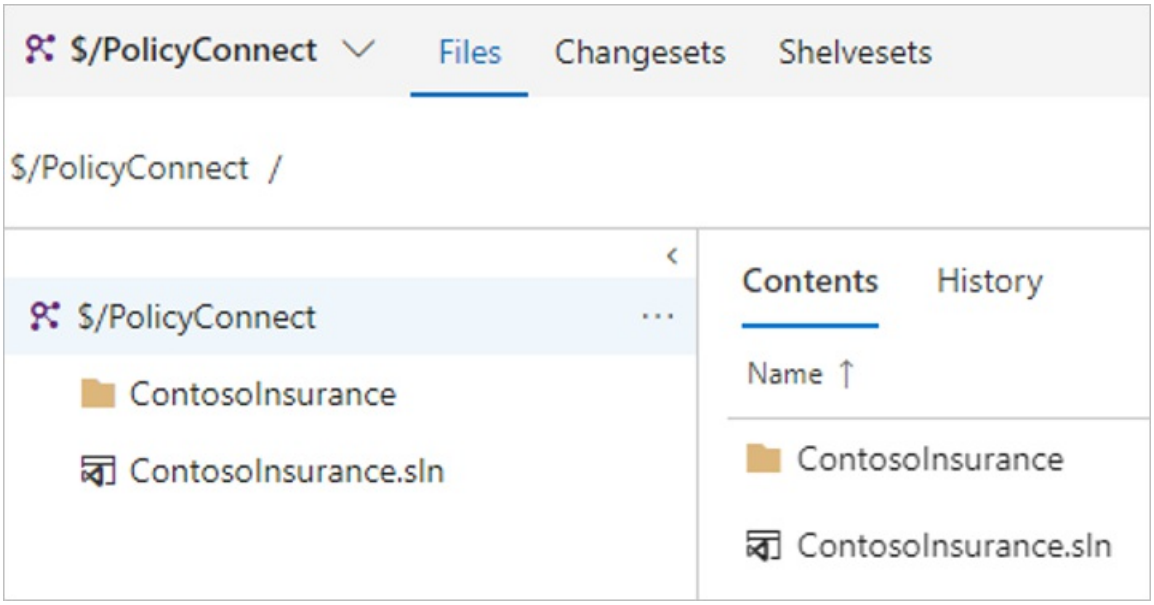
9. To confirm that the source code and history have been migrated, the dev lead checks other projects and code.



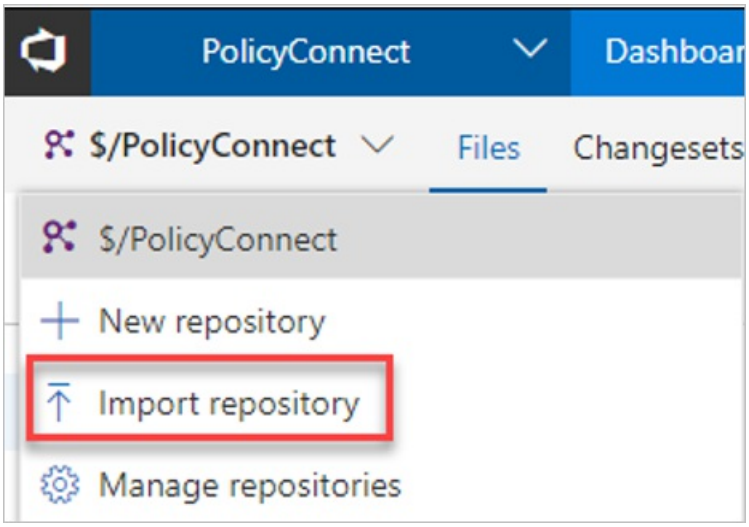
Move source control from TFVC to Git

With the migration now completed, Contoso admins want to move source code management from TFVC to Git. The admins need to import the source code that's currently in their Azure DevOps Services organization as Git repos in the same organization.

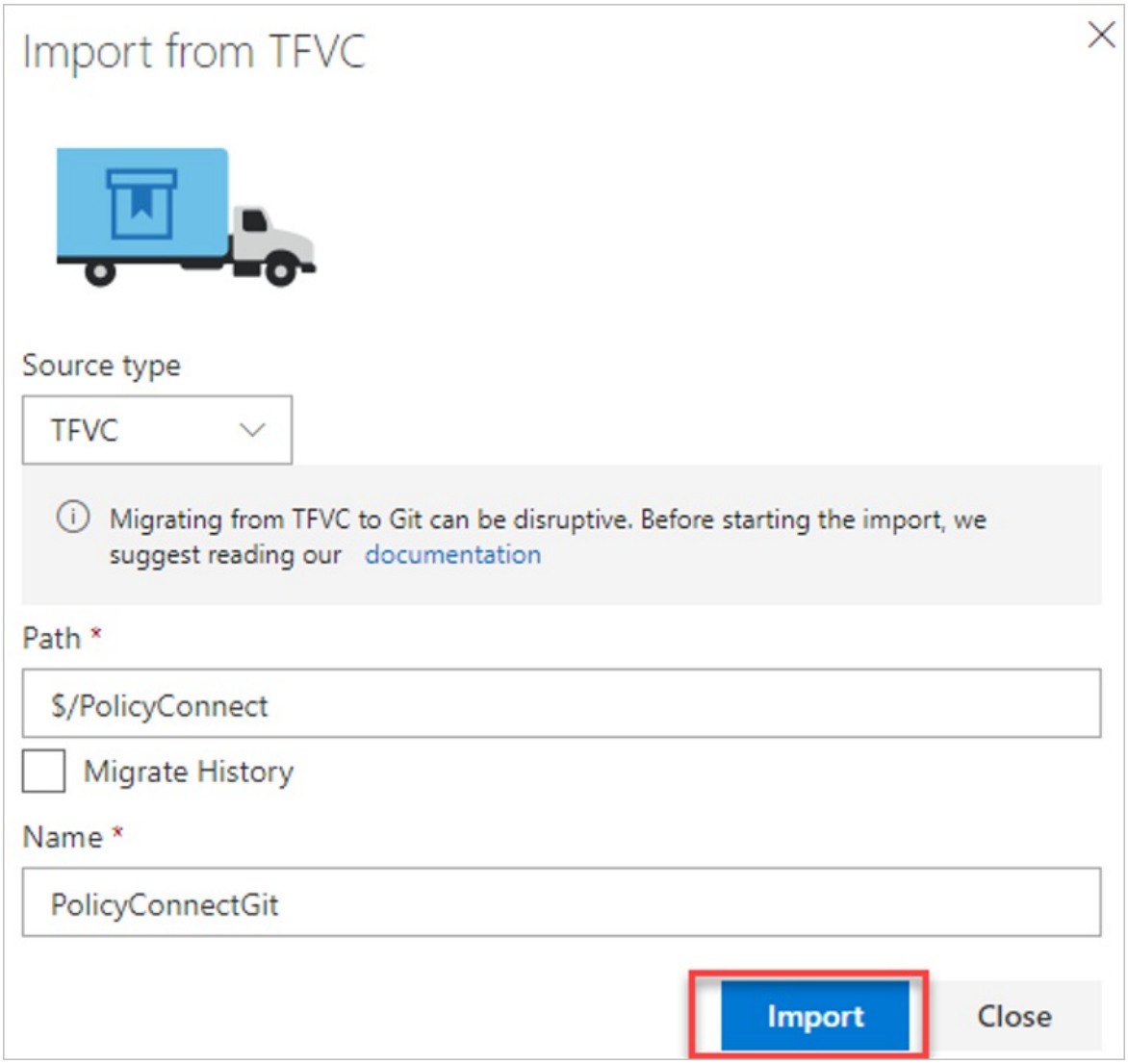
1. In the Azure DevOps Services portal, they open one of the TFVC repos, `$/PolicyConnect`, and review it.



2. In the source **$/PolicyConnect** drop-down list, they select **Import repository**.

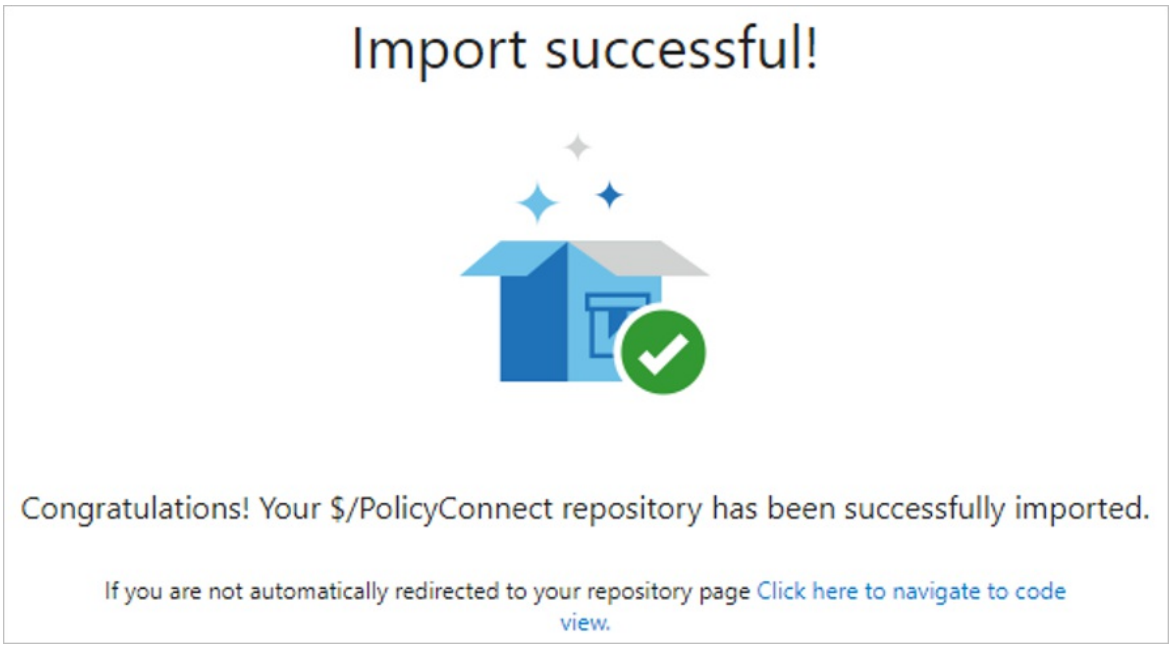


3. In the **Source type** drop-down list, they select **TFVC**. In the **Path** box, they specify the path to the repo, and then select **Import**. They decide to leave the **Migrate History** check box cleared.

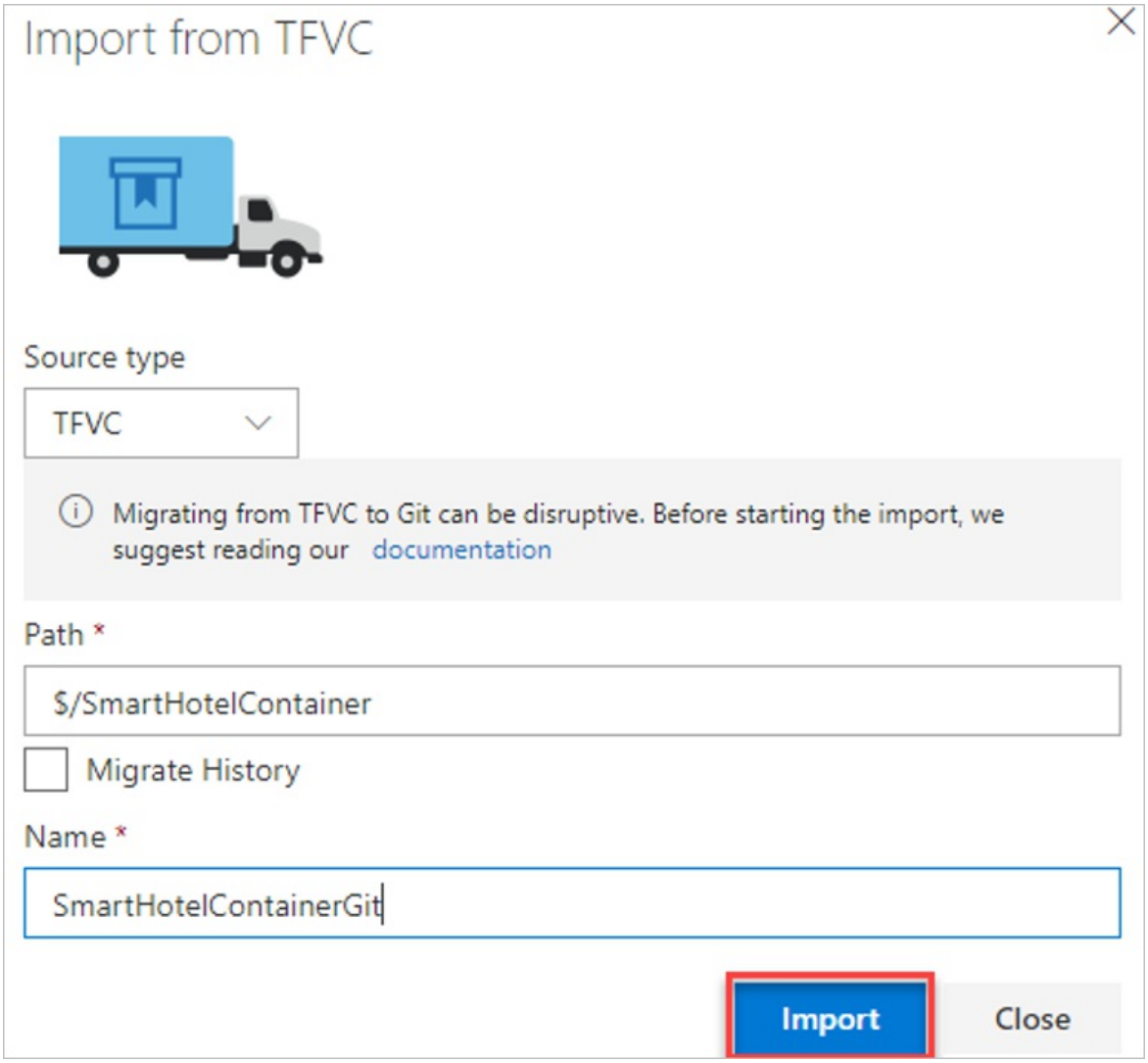


> **NOTE**
>
> Because TFVC and Git store version control information differently, we recommend that Contoso *not* migrate its repository history. This is the approach that Microsoft took when we migrated Windows and other products from centralized version control to Git.

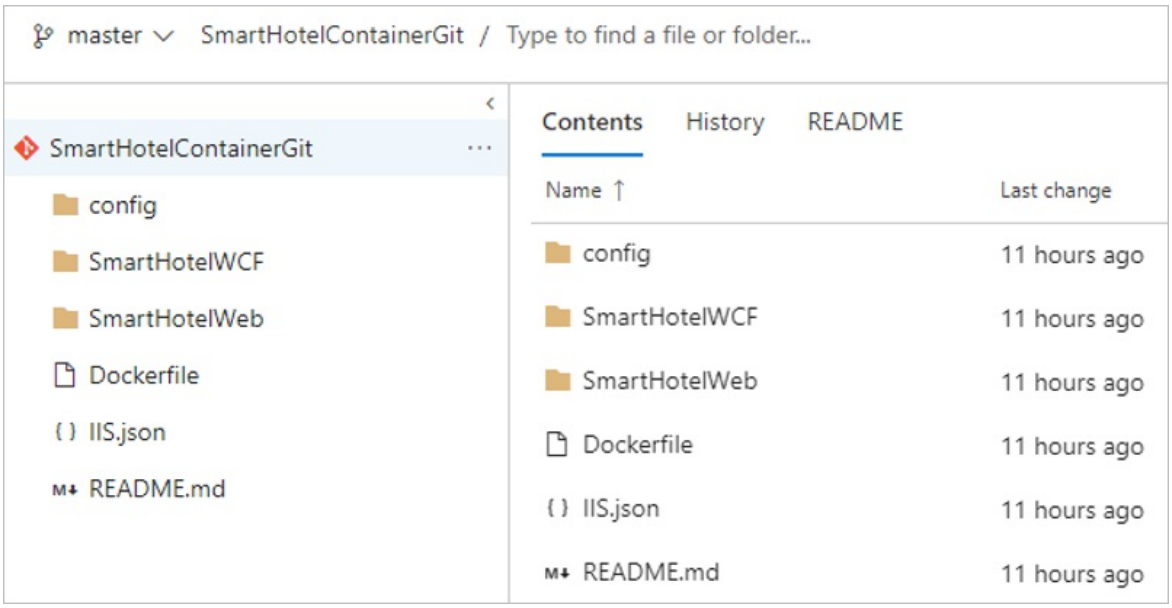
4. After the import finishes, the admins review the code.



5. They repeat the process for the second repository, `$/smarthotelcontainer`.



6. After the dev lead reviews the source, they agree that the migration to Azure DevOps Services is done. Azure DevOps Services now becomes the source for all development within the teams involved in the migration.



Need more help?

For more information, see Import repositories from TFVC to Git.

Clean up after migration

With the migration now complete, the Contoso team needs to do the following:

- Review the post-import article for information about additional import activities.
- Either delete the TFVC repos or place them in read-only mode. The code bases must not be used, but they can be referenced for their history.

Post-migration training

The Contoso team will need to provide Azure DevOps Services and Git training for relevant team members.

Move on-premises VMware infrastructure to Azure

5/18/2021 • 10 minutes to read • Edit Online

When fictional company Contoso migrates its VMware virtual machines (VMs) from an on-premises datacenter to Azure, two options are available to the team. This article focuses on Azure VMware Solution, which Contoso has determined to be the better migration option.

MIGRATION OPTIONS	OUTCOME
Azure Migrate	• Assess and migrate on-premises VMs. • Run workloads using Azure infrastructure as a service (IaaS). • Manage VMs with Azure Resource Manager.
Azure VMware Solution	• Use VMware Hybrid Cloud Extension (HCX) or vMotion to move on-premises VMs. • Run native VMware workloads on Azure bare-metal hardware. • Manage VMs using vSphere.

In this article, Contoso uses Azure VMware Solution to create a private cloud in Azure with native access to VMware vCenter and other tools that are supported by VMware for workload migration. Contoso can confidently use Azure VMware Solution, knowing that it's a first-party Microsoft offering backed by VMware.

Business drivers

Working closely with business partners, the Contoso IT team defines the business drivers for a VMware migration to Azure. These drivers can include:

- **Datacenter evacuation or shutdown:** Seamlessly move VMware-based workloads when they consolidate or retire existing datacenters.
- **Disaster recovery and business continuity:** Use a VMware stack deployed in Azure as a primary or secondary on-demand disaster recovery site for on-premises datacenter infrastructure.
- **Application modernization:** Tap into the Azure ecosystem to modernize Contoso applications without having to rebuild VMware-based environments.
- **Implementing DevOps:** Bring Azure DevOps tool chains to VMware environments and modernize applications at its own pace.
- **Ensure operational continuity:** Redeploy vSphere-based applications to Azure while avoiding hypervisor conversions and application refactoring. Extend support for legacy applications that run Windows and SQL Server.

Goals for migrating VMware on-premises to VMware in the cloud

With its business drivers in mind, Contoso has pinned down a few goals for this migration:

- Continue managing its existing environments with VMware tools that are familiar to its teams, while modernizing the applications with native Azure services.
- Seamlessly move Contoso VMware-based workloads from its datacenter to Azure, and integrate the VMware environment with Azure.
- After migration, the application in Azure should have the same performance capabilities as it does today in VMware. The application remains as critical in the cloud as it is on-premises.

These goals support Contoso's decision to use Azure VMware Solution and validate it as the best migration method.

Benefits of running VMware workloads in Azure

By using Azure VMware Solution, Contoso can now seamlessly run, manage, and secure applications across VMware environments and Azure with a common operating framework.

Contoso will capitalize on existing VMware investments, skills, and tools, including VMware vSphere, vSAN, and vCenter. At the same time, Contoso gets the scale, performance, and innovation of Azure. Additionally, it can:

- Set up VMware infrastructure in the cloud in minutes.
- Modernize applications at its own pace.
- Enhance VMware applications with dedicated, isolated, high-performance infrastructure and unique Azure products and services.
- Move or extend on-premises VMs to Azure without having to refactor its applications.
- Get scale, automation, and fast provisioning for VMware workloads on global Azure infrastructure.
- Benefit from a solution that's delivered by Microsoft, verified by VMware, and run on Azure infrastructure.

The solutions design

After Contoso pins down its goals and requirements, the company designs and reviews a deployment solution and identifies the migration process.

Current architecture

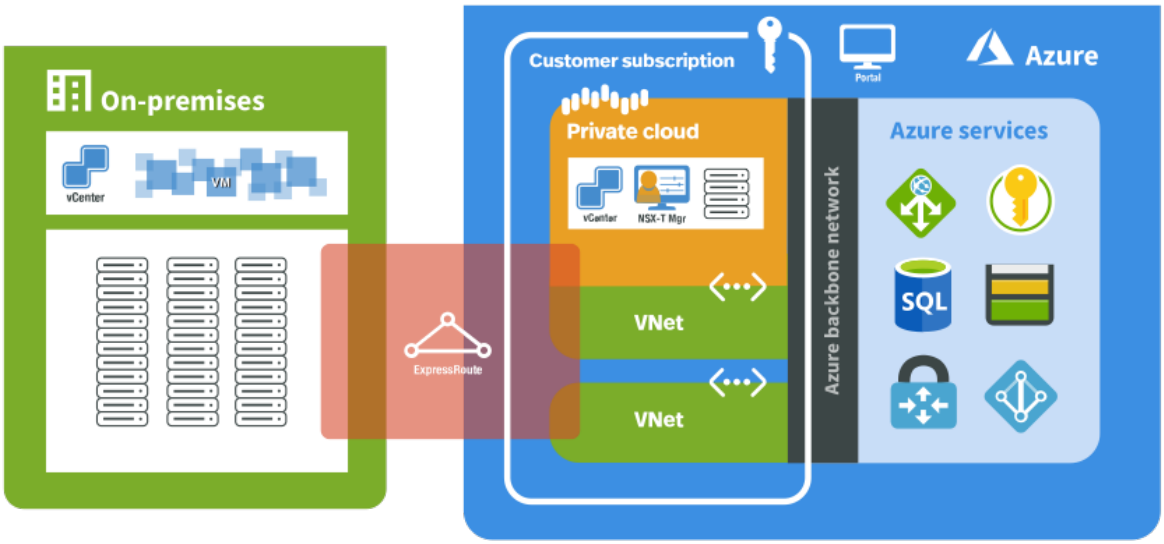
Contoso's current architecture features:

- VMs deployed to an on-premises datacenter that's managed by vSphere.
- Workloads deployed on a VMware ESXi host cluster that's managed with vCenter, vSAN, and NSX.

Proposed architecture

In its proposed architecture, Contoso will:

- Deploy an Azure VMware Solution private cloud to the West US Azure region.
- Connect the on-premises datacenter to Azure VMware Solution running in West US by using virtual networks and ExpressRoute with Global Reach enabled.
- Migrate VMs to dedicated Azure VMware Solution by using VMware Hybrid Cloud Extension (HCX).



Solution review

Contoso evaluates its proposed design by putting together a pros and cons list, as shown in the following table:

CONSIDERATION	DETAILS
Pros	• Bare-metal VMware infrastructure with high performance. • Infrastructure that's fully dedicated to Contoso and is physically isolated from the infrastructure of other customers. • Because Contoso is using a rehost that uses VMware, there's no special configuration or migration complexity. • Contoso can take advantage of its investment in Software Assurance by using the Azure Hybrid Benefit and extended security updates for legacy Windows and SQL platforms. • Contoso will retain full control of the application VMs in Azure.
Cons	• Contoso will need to continue supporting the application as VMware VMs rather than move them to a managed service such as Azure App Service and Azure SQL Database. • Azure VMware Solution is set up and priced based on a minimum of three large nodes rather than individual VMs in Azure IaaS. Contoso will need to plan its capacity needs, because the company currently uses an on-premises environment that restricts it from the on-demand nature of other services in Azure.

> **NOTE**
>
> For information about pricing, see Azure VMware Solution pricing.

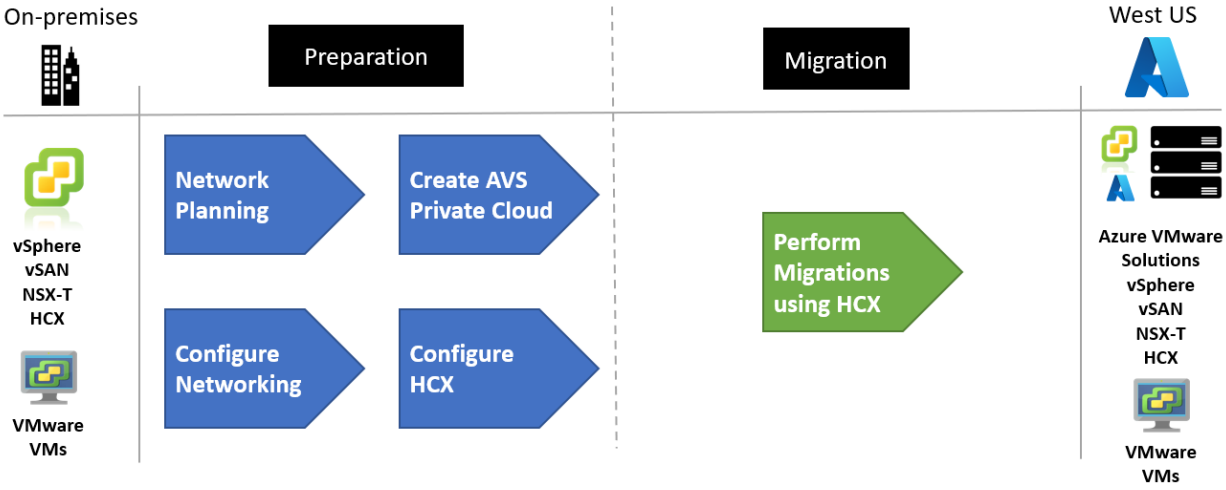
Migration process

Contoso will move its VMs to Azure VMware Solution by using the VMware HCX tool. The VMs will run in an Azure VMware Solution private cloud. VMware HCX migration methods include running a bulk or cold

migration. VMware vMotion or Replication-assisted vMotion (RAV) is a method reserved for workloads that run through a live migration.

To complete the process, the Contoso team:

- Plans its networking in Azure and ExpressRoute.
- Creates the Azure VMware Solution private cloud by using the Azure portal.
- Configures the network to include the ExpressRoute circuits.
- Configures the HCX components to connect its on-premises vSphere environment to the Azure VMware Solution private cloud.
- Replicates the VMs and then moves them to Azure by using VMware HCX.



Scenarios steps

- Step 1: Network planning
- Step 2: Create an Azure VMware Solution private cloud
- Step 3: Configure networking
- Step 4: Migrate VMs using HCX

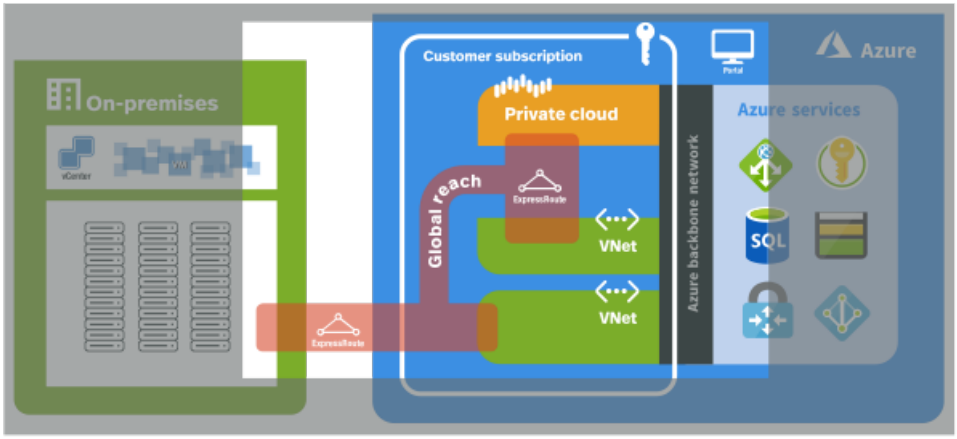
Step 1: Network planning

Contoso needs to plan out its networking to include Azure Virtual Network and connectivity between on-premises and Azure. The company needs to provide a high-speed connection between its on-premises and Azure-based environments, along with a connection to the Azure VMware Solution private cloud.

This connectivity is delivered through Azure ExpressRoute and will require some specific network address ranges and firewall ports for enabling the services. This high-bandwidth, low-latency connection allows Contoso to access services that run in its Azure subscription from the Azure VMware Solution private cloud environment.

Contoso will need to plan an IP address scheme that includes non-overlapping address space for its virtual networks. The company will need to include a gateway subnet for the ExpressRoute gateway.

The Azure VMware Solution private cloud is connected to Contoso's virtual network in Azure by using another Azure ExpressRoute connection. ExpressRoute Global Reach will be enabled to allow direct connection from on-premises VMs to VMs running on the Azure VMware Solution private cloud. The ExpressRoute Premium SKU is required to enable Global Reach.



Azure VMware Solution private clouds require, at minimum, a `/22` CIDR network address block for subnets. To connect to on-premises environments and virtual networks, this must be a non-overlapping network address block.

> **NOTE**
> To learn about network planning for Azure VMware Solution, see Networking checklist for Azure VMware Solution.

Step 2: Create an Azure VMware Solution private cloud

With its network and IP address planning completed, Contoso will next focus on setting up the Azure VMware Solution service in the West US Azure region. By using Azure VMware Solution, Contoso can deploy a vSphere cluster in Azure.

An Azure VMware Solution private cloud is an isolated, VMware software-defined datacenter that supports ESXi hosts, vCenter, vSAN, and NSX. The stack runs on dedicated and isolated bare-metal hardware nodes in an Azure region. The minimum initial deployment for an Azure VMware Solution private cloud is three hosts. Additional hosts can be added one at a time, up to a maximum of 16 hosts per cluster.

For more information, see Azure VMware Solution preview private cloud and cluster concepts.

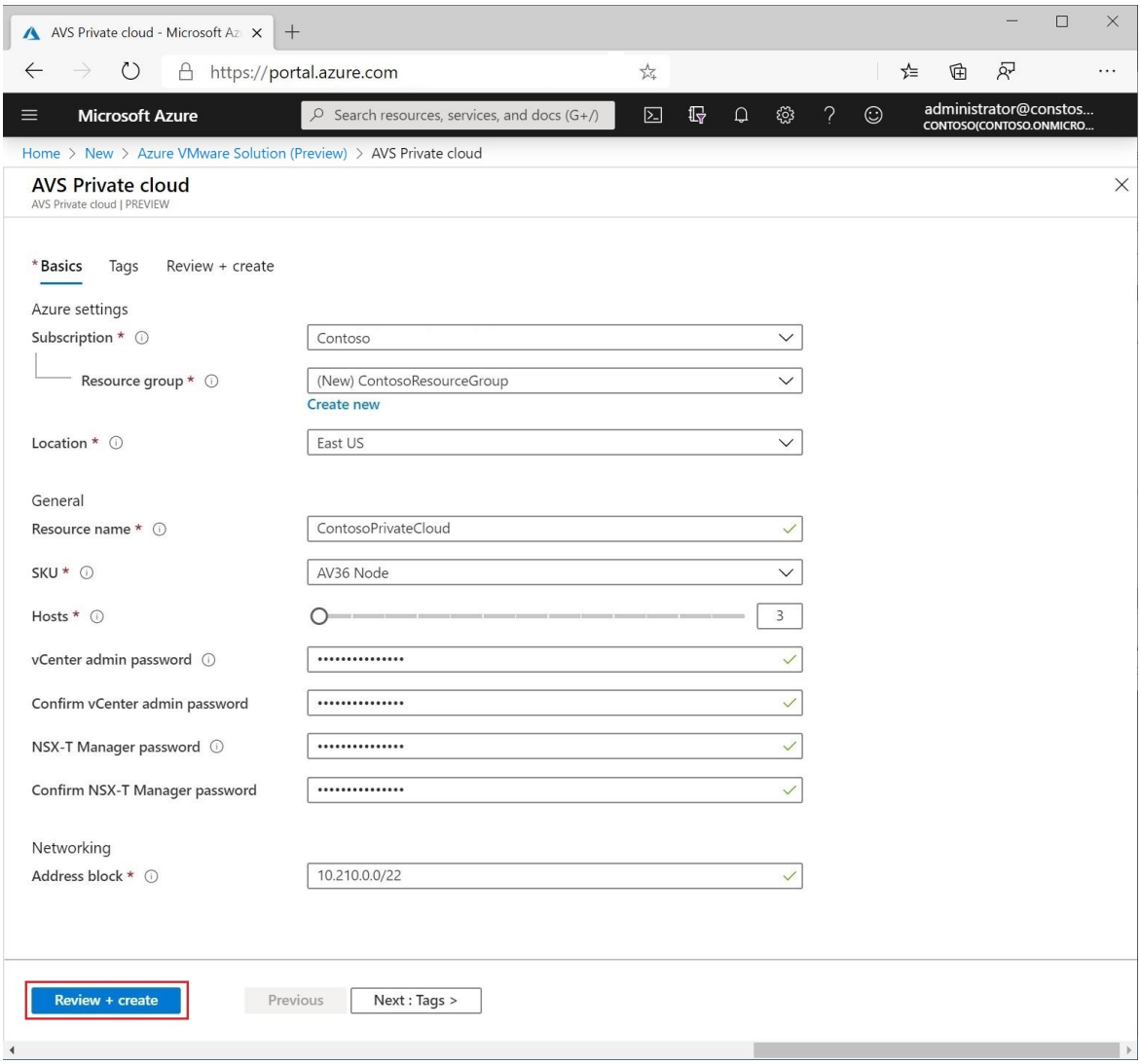
Azure VMware Solution private clouds are managed through the Azure VMware Solution portal. Contoso has its own vCenter Server in its own management domain.

To learn how to create Azure VMware Solution private clouds, see Deploy an Azure VMware Solution private cloud in Azure.

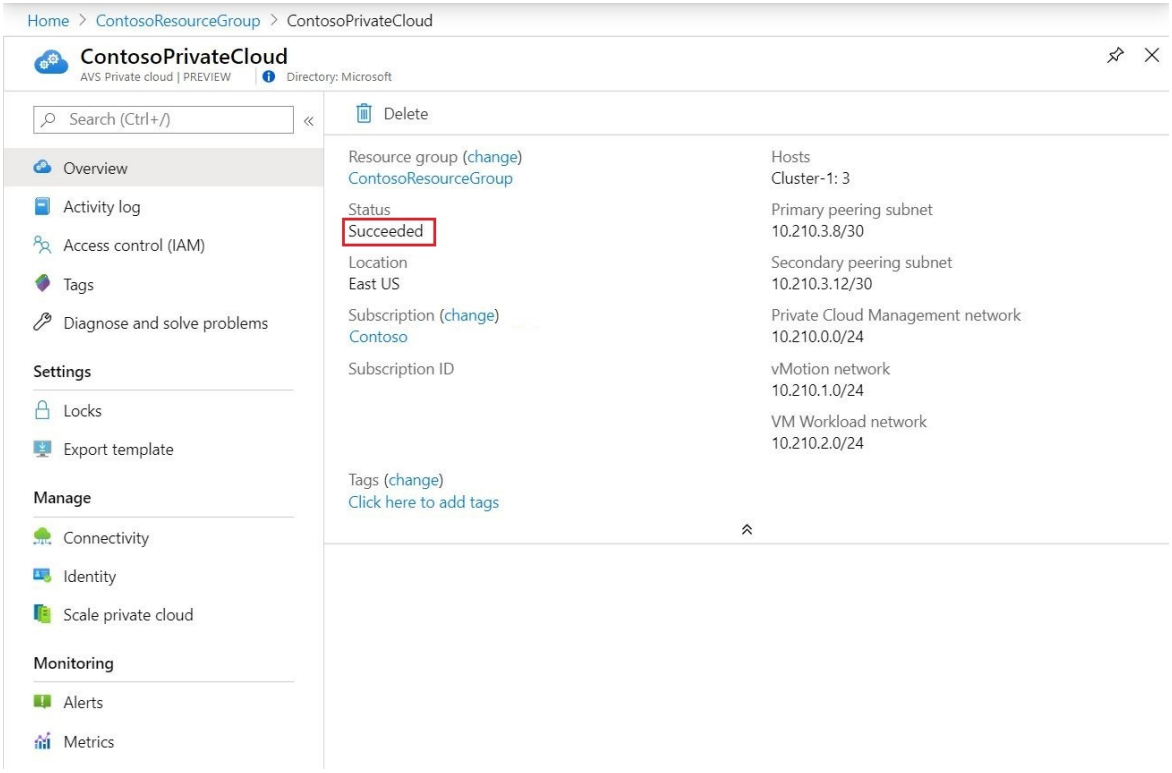
1. The Contoso team first registers the Azure VMware Solution provider with Azure by running the following command:

   ```
   az provider register -n Microsoft.AVS --subscription <your subscription ID>
   ```

2. In the Azure portal, the team creates the Azure VMware Solution private cloud by providing the networking information from the plan. The team then selects **Review + create**. This step takes about two hours.



3. The team verifies that the Azure VMware Solution private cloud deployment is complete by going to the resource group and selecting the private cloud resource. The status is displayed as **Succeeded**.



Step 3: Configure networking

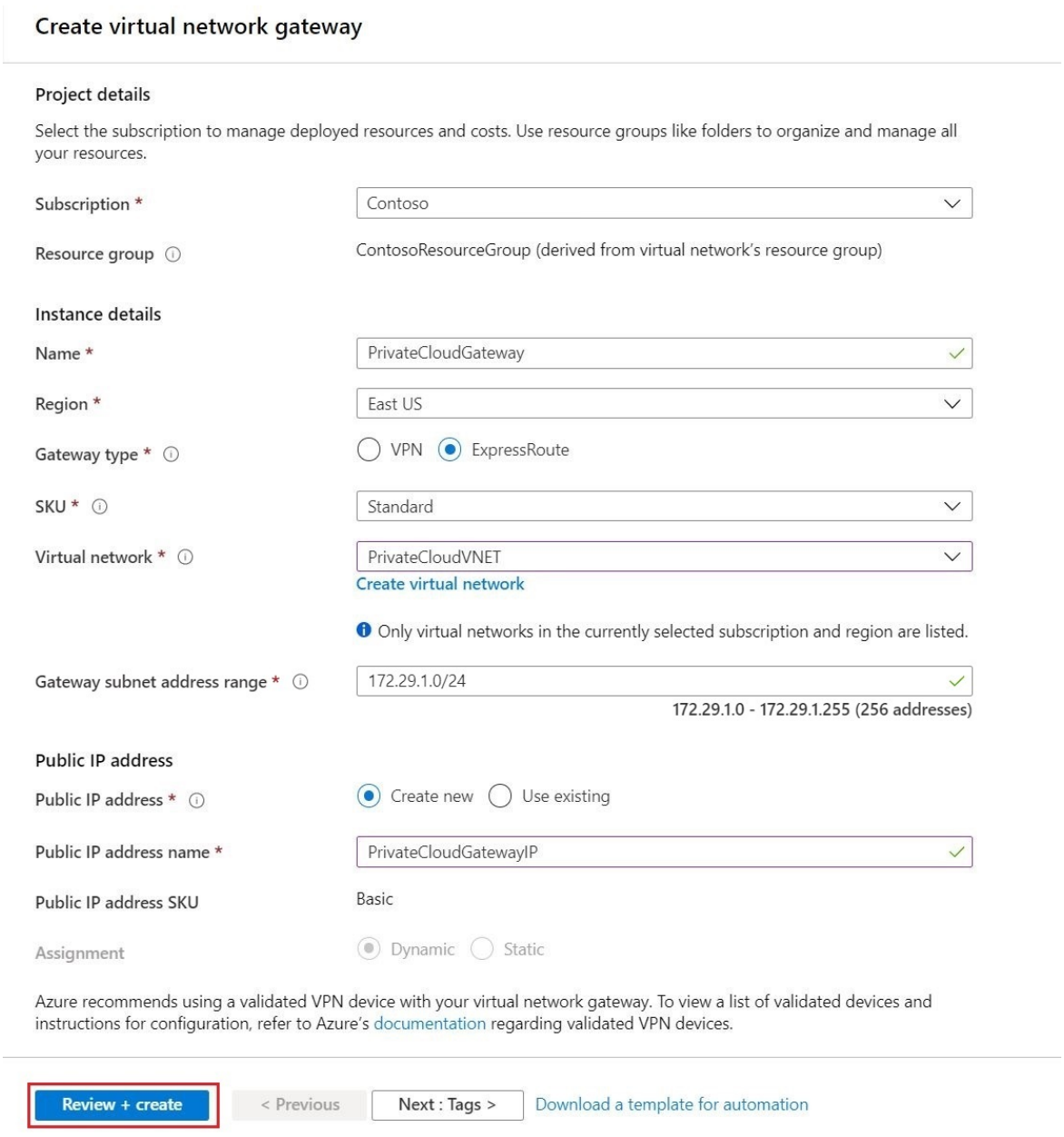
An Azure VMware Solution private cloud requires a virtual network. Because Azure VMware Solution doesn't support an on-premises vCenter during preview, Contoso requires additional steps for integration with its on-premises environment. By setting up an ExpressRoute circuit and a virtual network gateway, the team connects its virtual networks to the Azure VMware Solution private cloud.

For more information, see Configure networking for your VMware private cloud in Azure.

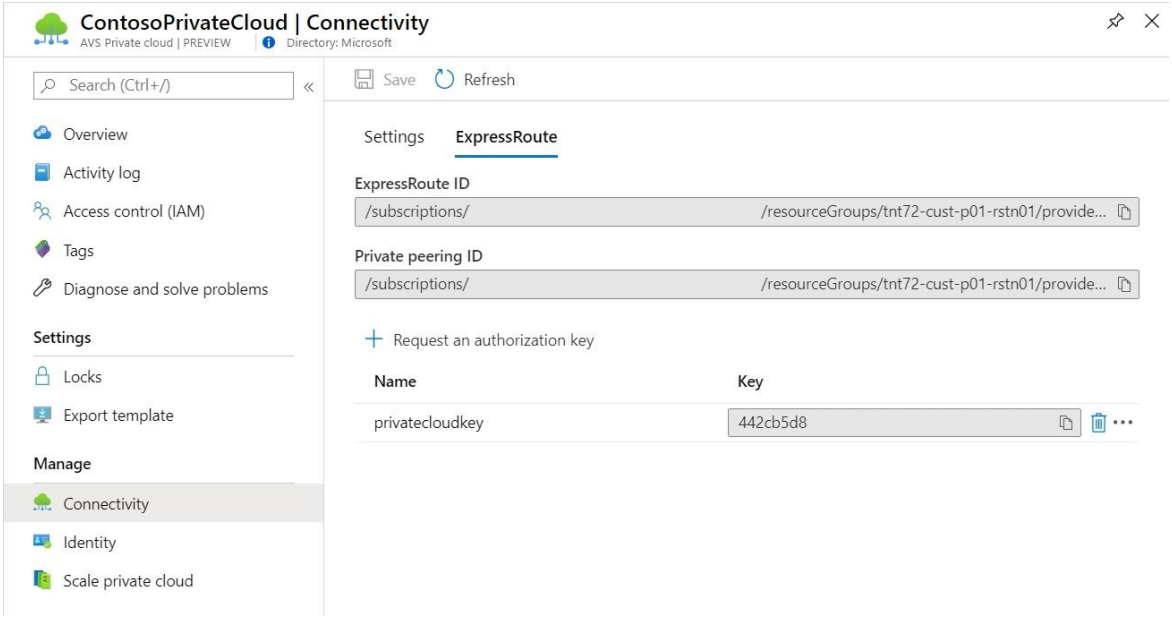
1. The Contoso team first creates a virtual network with a gateway subnet.

   > **IMPORTANT**
   > The team must use an address space that *does not* overlap with the address space that it used when it created the private cloud.

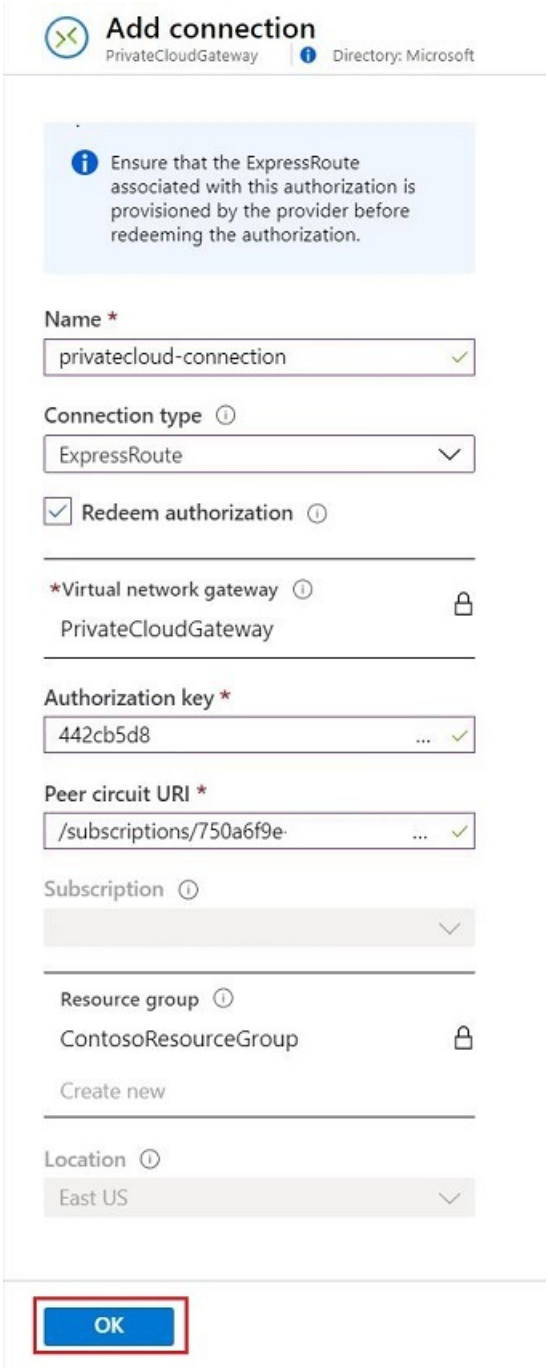
2. The team creates the ExpressRoute VPN gateway, making sure to select the correct SKU, and then selects **Review + create**.



3. The team gets the authorization key to connect ExpressRoute to the virtual network. The key is found on the connectivity screen of the Azure VMware Solution private cloud resource in the Azure portal.



4. The team connects the ExpressRoute to the VPN gateway that connects the Azure VMware Solution private cloud to the Contoso virtual network. It does this by creating a connection in Azure.



For more information, see Learn how to access an Azure VMware Solution private cloud.

Step 4: Migrate by using VMware HCX

To move VMware VMs to Azure using HCX, the Contoso team will need to follow these high-level steps:

- Install and configure VMware HCX.
- Perform migrations to Azure by using HCX.

For more information, see Install HCX for Azure VMware Solution.

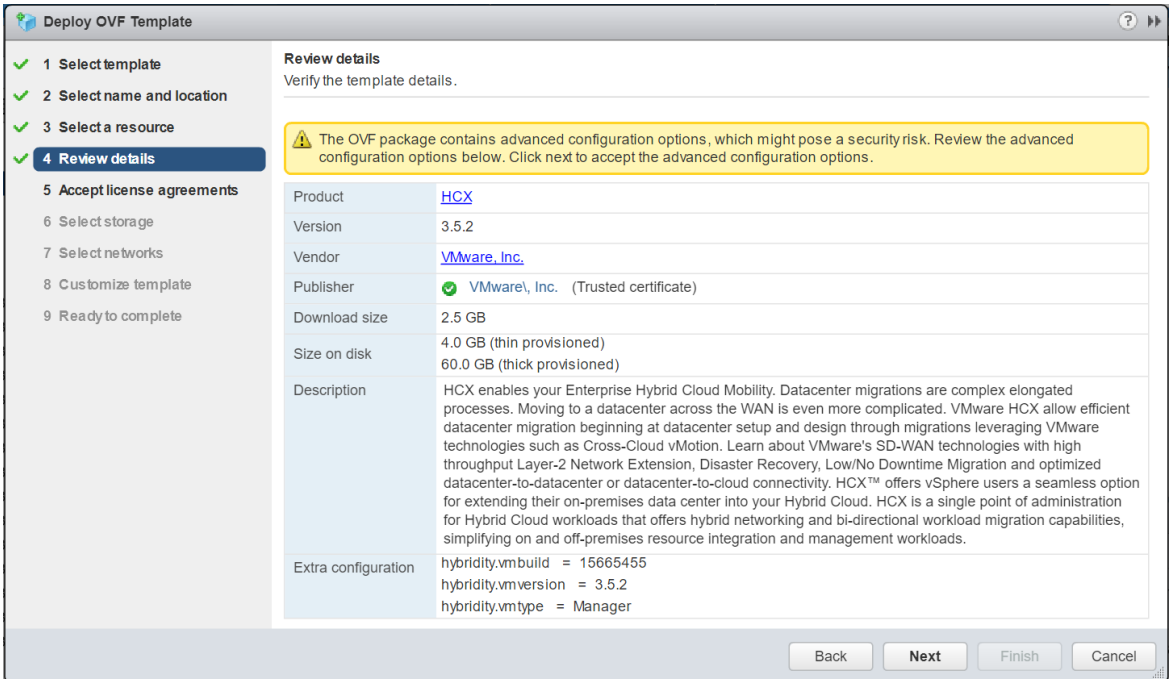
Install and configure VMware HCX for the public cloud

VMware HCX is a VMware product that's part of the Azure VMware Solution default installation. HCX Advanced is installed by default, but it can be upgraded to HCX Enterprise as additional features and functionality are required.

Azure VMware Solution automates the cloud manager component of HCX in Azure VMware Solution. It provides the customer activation keys and download link to the connector HCX appliance that must be configured on the on-premises side and in a customer's vCenter domain. These elements are then paired with the Azure VMware Solution cloud appliance, so that customers can take advantage of services such as migration and Layer 2

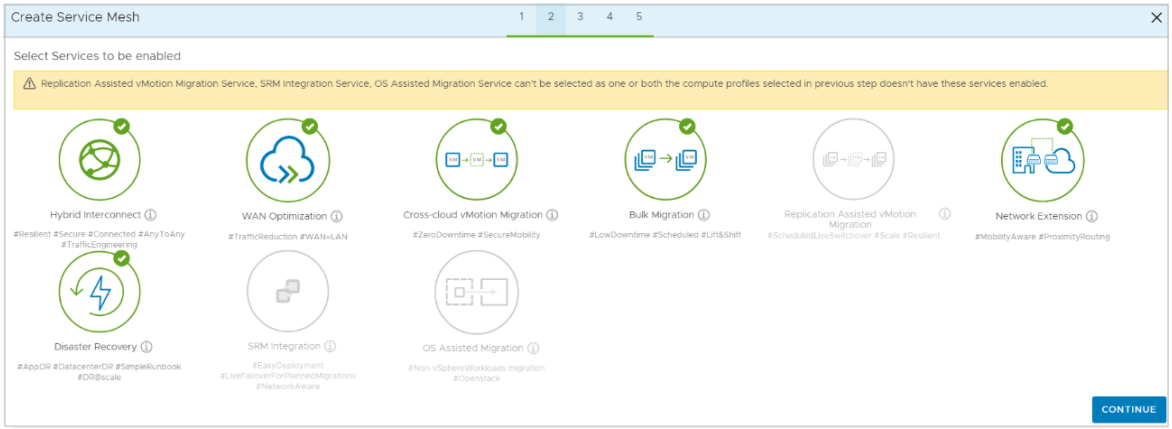
stretch.

- The Contoso team is deploying the HCX by using an OVF package that's provided by VMware.



To install and configure HCX for your Azure VMware Solution private cloud, see Install HCX for Azure VMware Solution.

- As the team is configuring HCX, it has chosen to enable migration and other options, including disaster recovery.

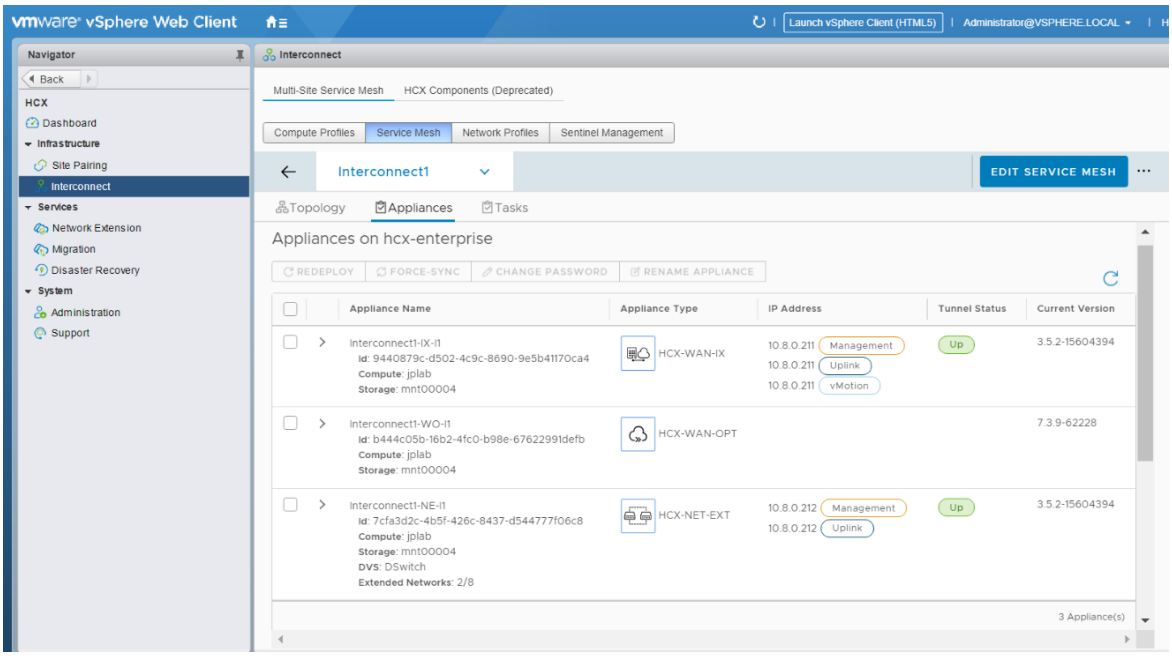


For more information, see HCX installation workflow for HCX public clouds.

Migrate VMs to Azure by using HCX

When both the on-premises datacenter (source) and the Azure VMware Solution private cloud (destination) are configured with the VMware cloud and HCX, Contoso can begin migrating its VMs. The team can move VMs to and from VMware HCX-enabled datacenters by using multiple migration technologies.

- Contoso's HCX application is online, and the status is green. The team is now ready to migrate and protect Azure VMware Solution VMs by using HCX.



VMware HCX Bulk Migration

This migration method uses the VMware vSphere replication protocols to move multiple VMs simultaneously to a destination site. Benefits include:

- This method is designed to move multiple VMs in parallel.
- The migration can be set to finish on a predefined schedule.
- The VMs run at the source site until failover begins. The service interruption is equivalent to a reboot.

VMware HCX vMotion live migration

This migration method uses the VMware vMotion protocol to move a single VM to a remote site. Benefits include:

- This method is designed to move one VM at a time.
- There's no service interruption when the VM state is migrated.

VMware HCX Cold Migration

This migration method uses the VMware near-field communication protocol. The option is automatically selected when the source VM is powered off.

VMware HCX Replication-assisted vMotion

VMware HCX RAV combines the benefits of VMware HCX Bulk Migration, which include parallel operations, resiliency, and scheduling, with the benefits of VMware HCX vMotion migration, which include zero downtime during VM state migration.

Additional resources

For additional VMware technical documentation, see:

- VMware HCX documentation
- Migrate virtual machines by using VMware HCX

Move on-premises Remote Desktop Services to Azure Virtual Desktop scenario

8/25/2021 • 11 minutes to read • Edit Online

Azure Virtual Desktop is a comprehensive desktop and application virtualization service running in the cloud. It's the only virtual desktop infrastructure (VDI) that delivers simplified management, Windows 10 Enterprise multi-session optimizations for Microsoft 365 Apps for enterprise, and support for Remote Desktop Services (RDS) environments. Deploy and scale Windows desktops and applications on Azure in minutes, and get built-in security and compliance features.

MIGRATION OPTIONS	OUTCOME
Azure Migrate	Assess and migrate on-premises RDS environments. Run workloads in an Azure Virtual Desktop environment. Manage Azure Virtual Desktop with Azure Virtual Desktop management UX.

> **NOTE**
> This article focuses on using Azure Virtual Desktop to move an on-premises RDS environment to Azure.

Business drivers

Working closely with business partners, the Contoso IT team will define the business drivers for a VDI migration to Azure. These drivers might include:

- **Current environment end-of-life:** A datacenter is out of capacity when it reaches the end of a lease or is closing down. Migrating to the cloud provides virtually unlimited capacity. Current software might also be reaching its end of life where it has become necessary to upgrade the software running Contoso's current VDI solution.
- **Multi-session Windows 10 VDI:** Provide Contoso users with the only multi-session Windows 10 desktop virtualized in the cloud that's highly scalable, up to date, and available on any device.
- **Optimize for Microsoft 365 Apps for enterprise:** Deliver the best Microsoft 365 Apps for enterprise experience, with multi-session virtual desktop scenarios providing the most productive virtualized experience for Contoso's users.
- **Deploy and scale in minutes:** Quickly virtualize and deploy modern and legacy desktop applications to the cloud in minutes with unified management in the Azure portal.
- **Secure and productive on Azure and Microsoft 365:** Deploy a complete, intelligent solution that enhances creativity and collaboration for everyone. Shift to Microsoft 365 and get Office 365, Windows 10, and Enterprise Mobility + Security.

RDS on-premises to Azure Virtual Desktop in the cloud goals

With the business drivers in mind, Contoso has pinned down goals for this migration:

- Modernize the virtual desktop environment for the cloud.
- Take advantage of existing Microsoft 365 licenses.

- Improve security of corporate data when users work remotely.
- Optimize the new environment for cost and growth.

These goals support the decision to use Azure Virtual Desktop and validate it as the best migration method for Contoso.

Benefits of running Azure Virtual Desktop

Using Azure Virtual Desktop, Contoso can now seamlessly run, manage, and scale its VDI solution quickly and easily. The company also can provide an optimized multi-session Windows 10 environment to its users.

Contoso will capitalize on existing Microsoft 365 licenses while using the scale, performance, security, and innovation of Azure.

Additional benefits might include:

- Access to Azure Virtual Desktop from anywhere.
- Optimized Microsoft 365 Apps for enterprise environment.
- Azure Virtual Desktop for dev/test environments.

Solutions design

After pinning down goals and requirements, Contoso designs and reviews a deployment solution and identifies the migration process.

Current architecture

RDS is deployed to an on-premises datacenter. Microsoft 365 is licensed and in use by the organization.

Proposed architecture

- Sync Active Directory or Azure Active Directory Domain Services.
- Deploy Azure Virtual Desktop.
- Migrate on-premises RDS servers to Azure.
- Convert user profile disks (UPDs) to FSLogix profile containers.

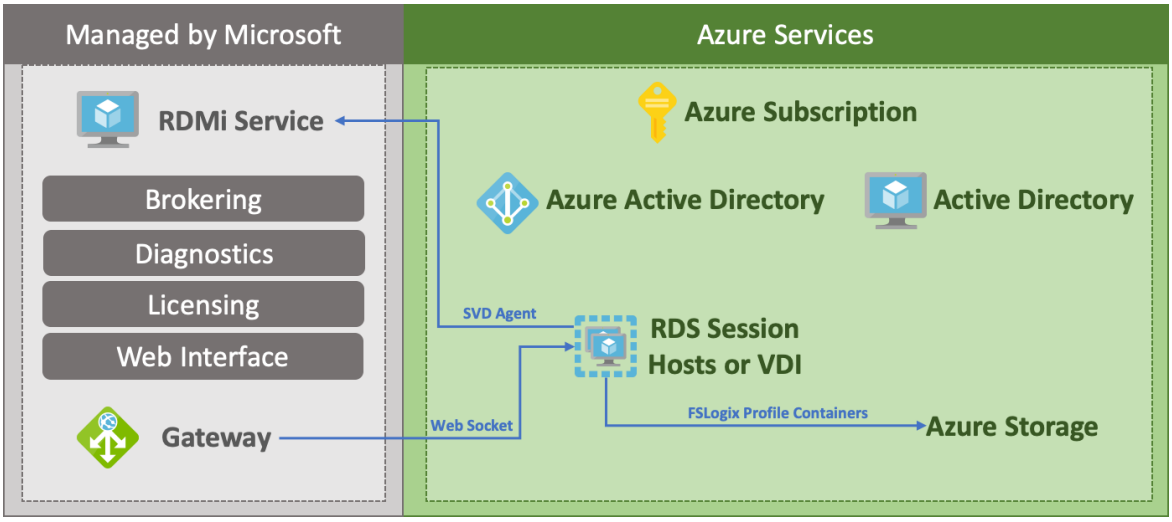


Figure 1: Proposed architecture.

Solution review

Contoso evaluates the proposed design by putting together a list of pros and cons.

CONSIDERATION	DETAILS
Pros	Windows 10 Enterprise multi-session environment. Cloud-based, allowing access from anywhere. Take advantage of other Azure services like Azure Files within the Azure Virtual Desktop environment. Optimized for the Microsoft modern desktop.
Cons	To fully optimize for Azure, Contoso will have to rebuild Windows 10 images optimized for multiuser sessions. Azure Virtual Desktop doesn't support user profile disks, so UPDs must be migrated to FSLogix profile containers.

Migration process

Contoso will move VMs to Azure Virtual Desktop by using the Lakeside assessment tool and Azure Migrate. Contoso will need to:

- Run the assessment tool against its on-premises RDS infrastructure to establish the scale of the Azure Virtual Desktop deployment in Azure.
- Migrate to Azure Virtual Desktop via either Windows 10 Enterprise multi-session or persistent virtual machines.
- Optimize the Azure Virtual Desktop multi-session by scaling up and down as needed to manage costs.
- Virtualize applications and assign users as needed to continue to secure and manage the Azure Virtual Desktop environment.

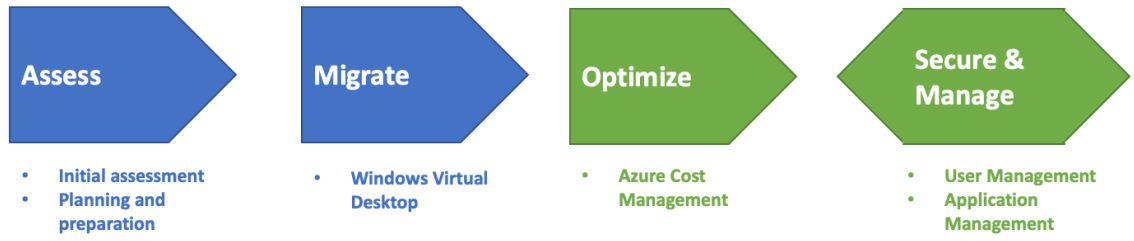


Figure 2: The migration process.

Scenario steps

1. Assess the current RDS environment.
2. Create the VDI and new images in Azure and migrate and persist VMs to Azure.
3. Convert UPDs to FSLogix profile containers.
4. Replicate any persistent VMs to Azure.

Step 1: Assess the current on-premises environment

Contoso will provision the Azure Virtual Desktop service in the `East US 2` Azure region. With Azure Virtual Desktop, Contoso can provision virtual machines, host pools, and create application groups. Azure Virtual Desktop also configures an availability set for all the servers in the Azure Virtual Desktop solution. Azure Virtual Desktop allows Contoso to create a high-available VDI environment and to scale up and down quickly as needed.

> **NOTE**
>
> Contoso reviewed two scenarios during the assessment: multi-session (shared) instances of RDS and persistent (or user-dedicated) virtual machines.

1. Make sure that domain services, either Active Directory or Azure Active Directory Domain Services, are synchronized with Azure Active Directory (Azure AD). Ensure the domain service is accessible from the Azure subscription and virtual network to be connected where Azure Virtual Desktop will be deployed.

   > **NOTE**
   >
   > Learn more about Azure AD Connect for synchronizing Active Directory on-premises with Azure AD.

   > **NOTE**
   >
   > Learn about provisioning Azure Active Directory Domain Services and synchronizing Azure AD to it.

2. Create a new Azure Migrate project.

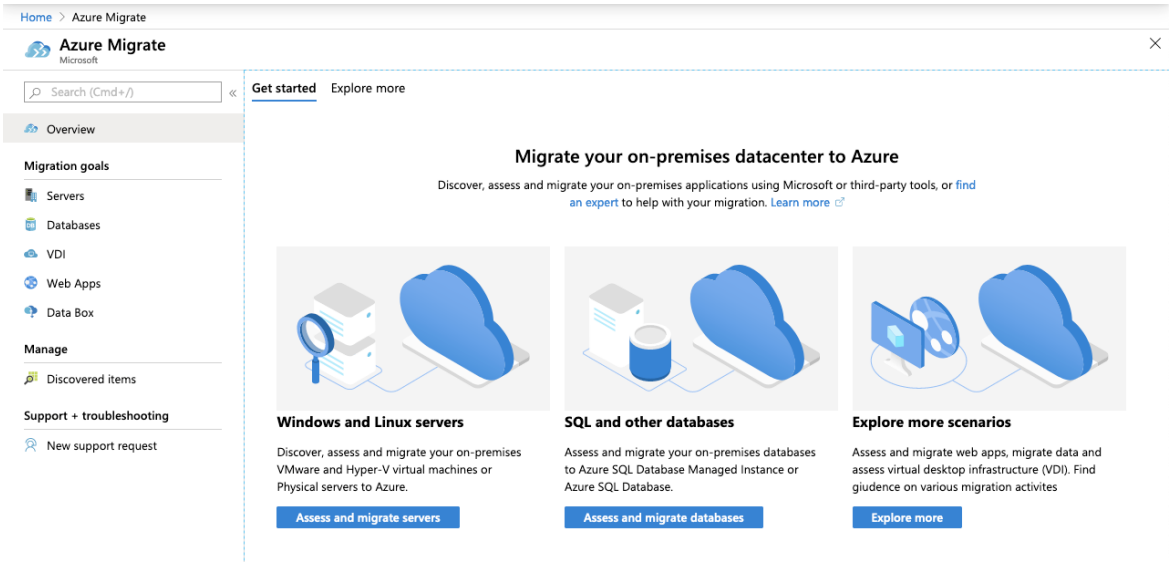


Figure 3: Creating a new Azure Migrate project.

3. Select the option to assess and migrate servers, select **VDI**, and add a tool.

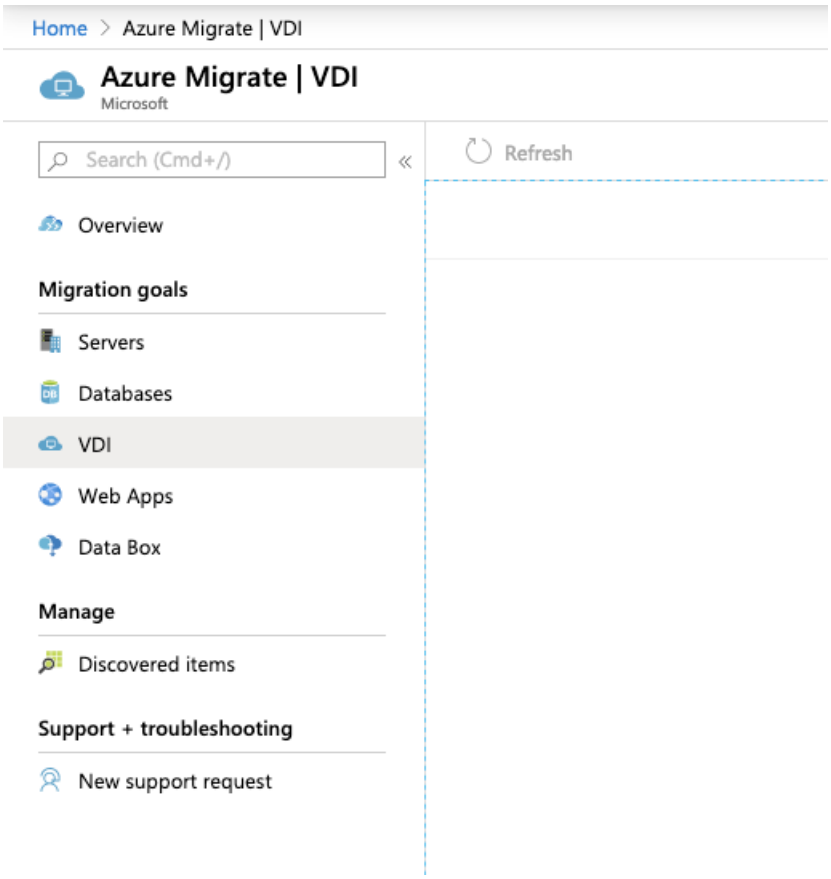


Figure 4: Target Azure Migrate goals.

4. Set the subscription, resource group, project name, and geography for the migrate job data.

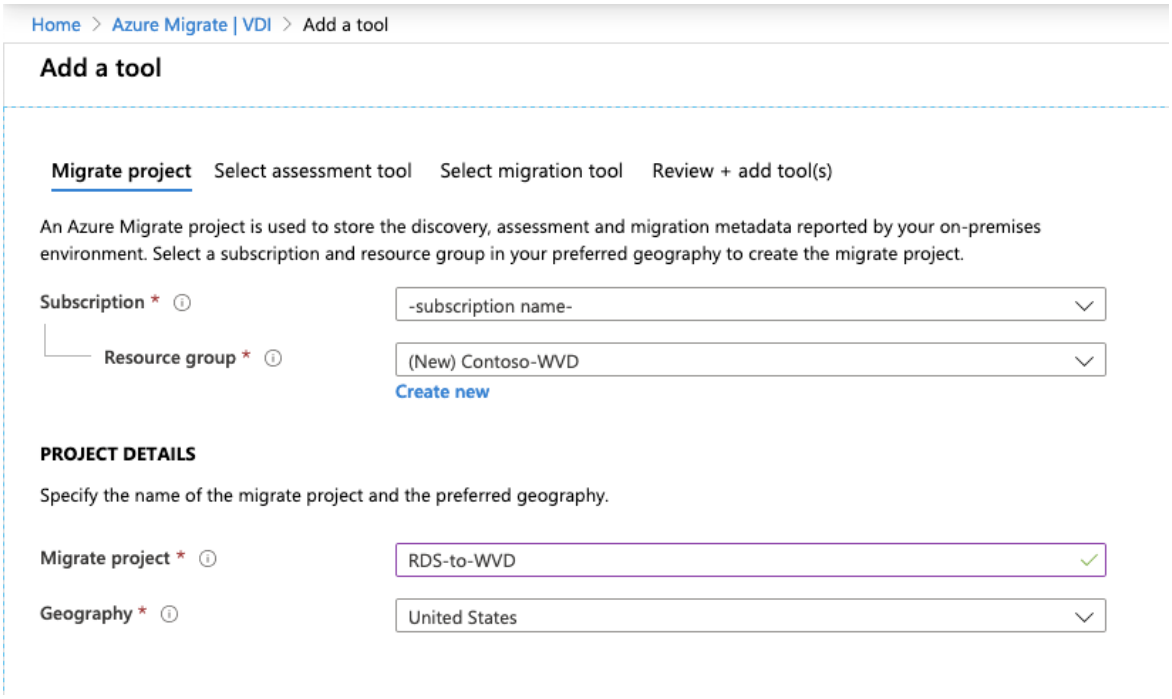


Figure 5: Adding job data to the migration.

> **IMPORTANT**
> This location isn't where the new Azure Virtual Desktop environment will be deployed. Only the data related to the Azure Migrate project will be stored here.

5. Select **Lakeside: SysTrack** as the assessment tool.

6. Select **Azure Migrate: Server Migration** as the migration tool.

7. Add the tools to the migration project.

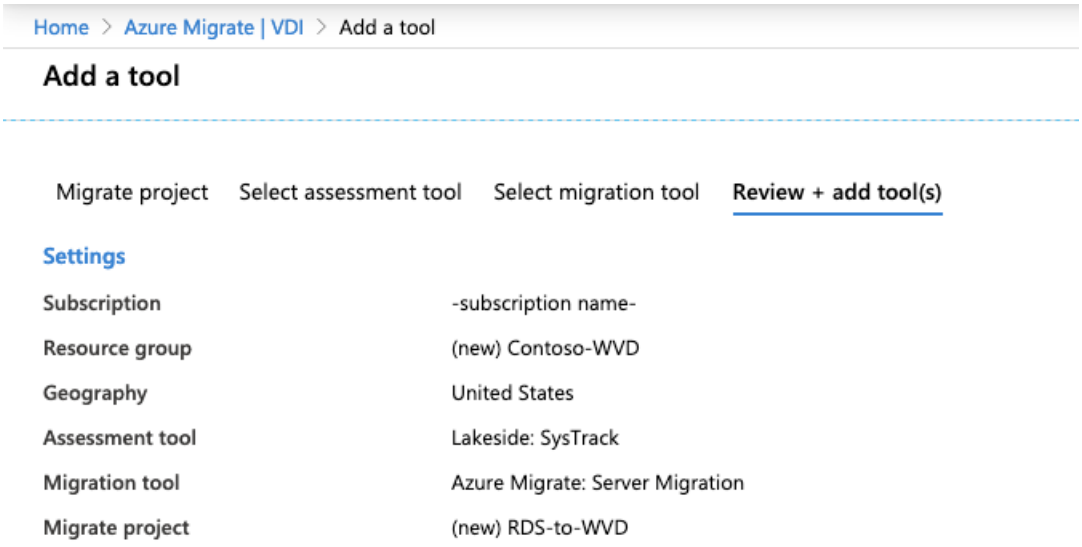


Figure 6: Adding tools to the migration.

8. Start the assessment of the current environment by selecting **Register with Azure Migrate** in the Lakeside tool.

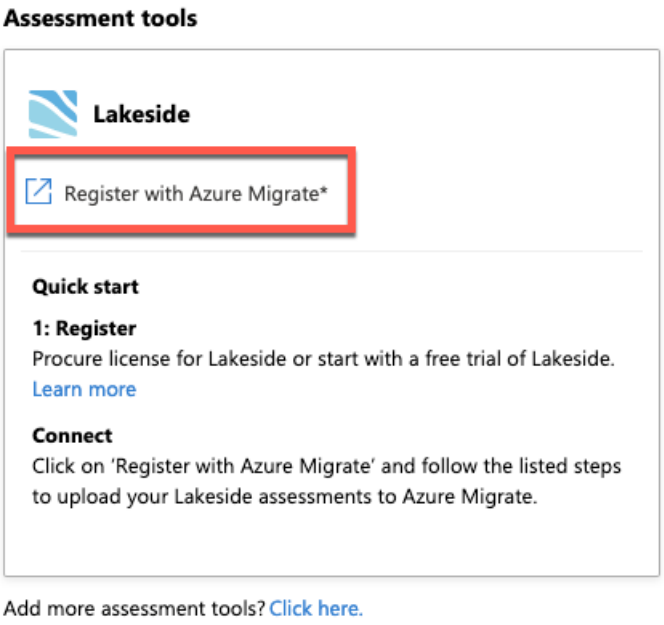


Figure 7: Assessing the current environment.

9. Contoso connects Azure Migrate and Lakeside, and accepts any requested permissions.

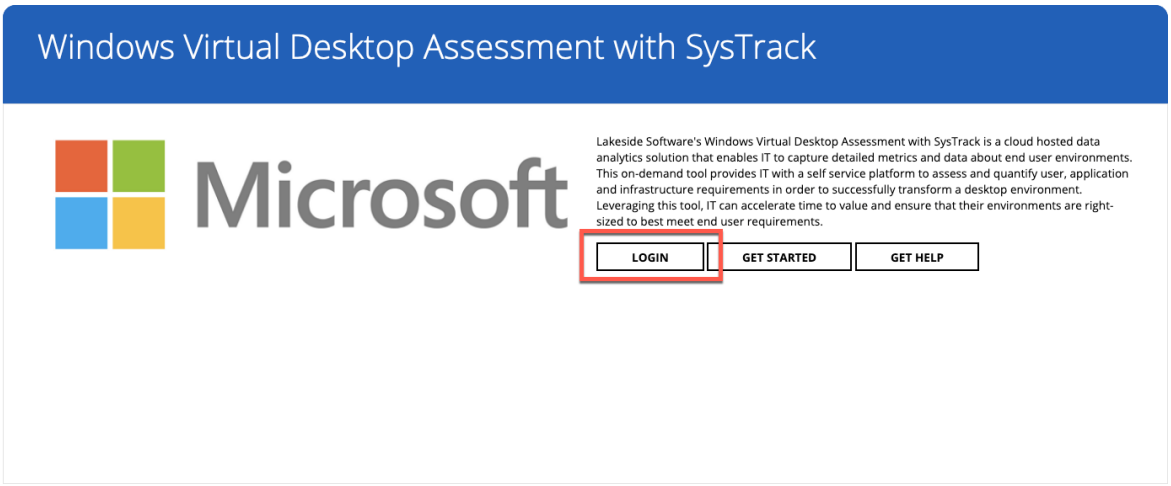


Figure 8: Connecting Azure to Lakeside.

10. Contoso continues with the Lakeside tool to create a new tenant and start assessing the current on-premises RDS environment. From the dashboard, Contoso can access the deployment guide, download the assessment client to deploy to the current environment, and review the data collected from these agents.

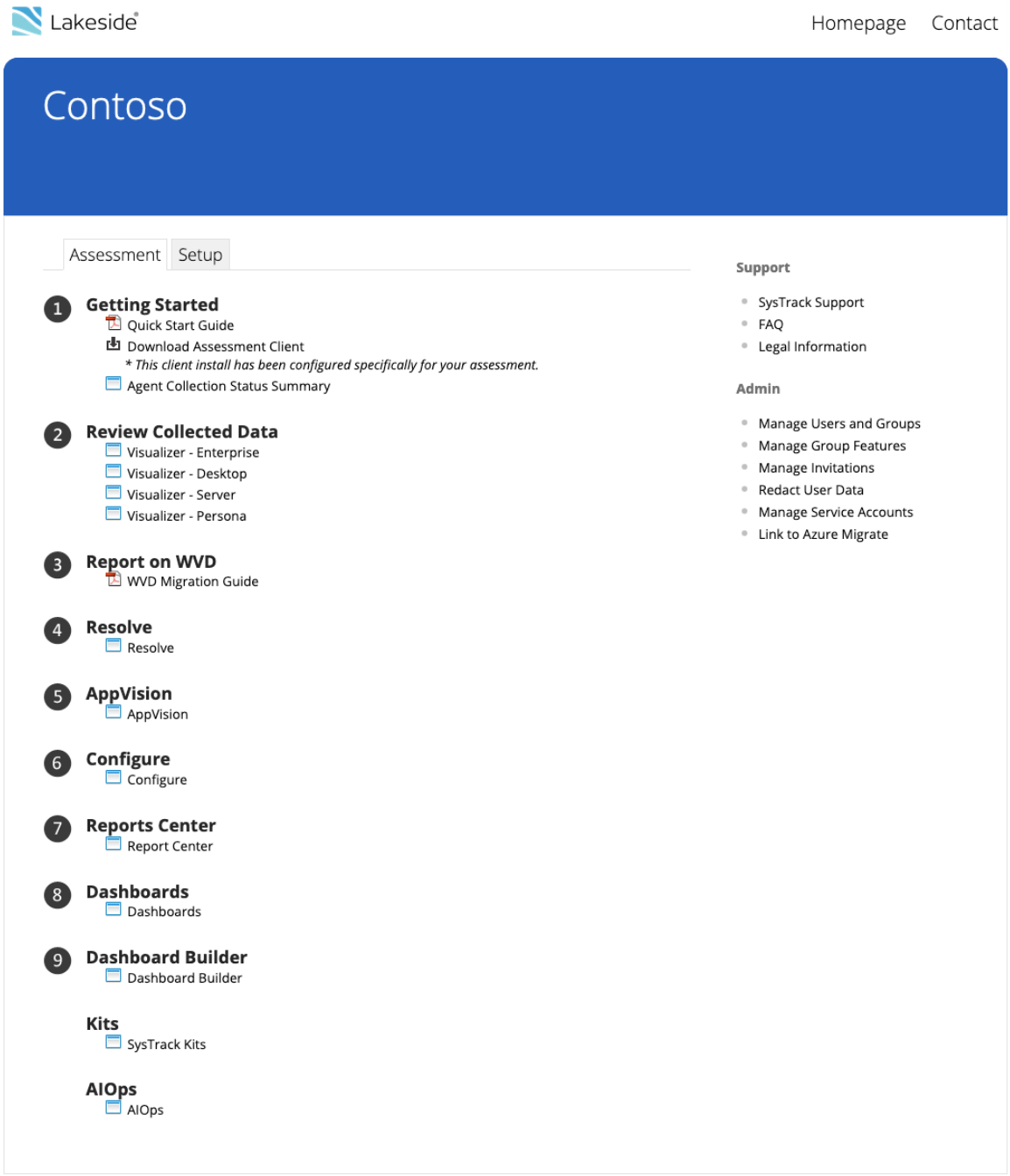


Figure 9: The Lakeside dashboard.

11. After an adequate amount of data is captured, Contoso reviews the assessment data to determine the best migration path. This assessment data includes the raw assessment data from the desktops data and the data broken into different user personas. This information includes the:

- Number of users in each persona.
- Applications in use by users.
- Resource consumption by user.
- Resource utilization averages by user persona.
- VDI server performance data.
- Concurrent user reports.
- Top software packages in use.

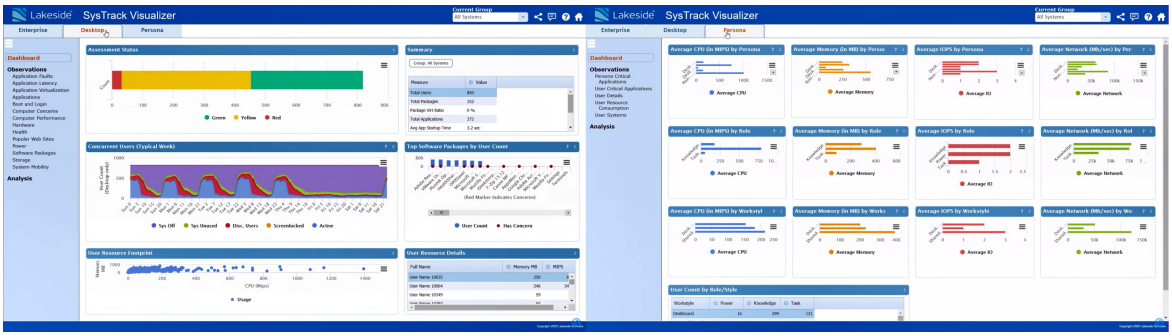


Figure 10: Lakeside dashboard reports.

The data is analyzed by Contoso to determine the most cost-effective use of both pooled Azure Virtual Desktop resources and personal Azure Virtual Desktop resources.

> **NOTE**
>
> Contoso will also need to migrate application servers to Azure to get the company closer to the Azure Virtual Desktop environment and reduce network latency for its users.

Step 2: Create the Azure Virtual Desktop environment for pooled desktops

Using the Azure portal, Contoso will create an Azure Virtual Desktop environment to use for pooled resources. Later, it will go through the migration steps to attach personal desktops to the same environment.

1. Contoso selects the correct subscription, and creates a new Azure Virtual Desktop host pool.

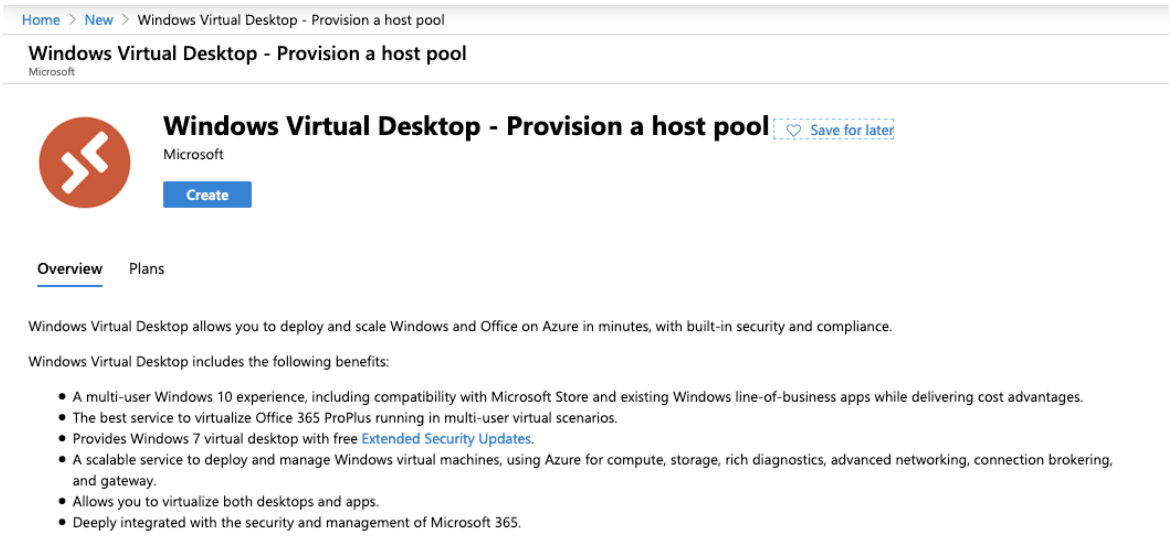


Figure 11: A new Azure Virtual Desktop host pool.

2. Specify the subscription, resource group, and region. Then select the name for the host pool, desktop type, and default desktop users. Desktop type is set to **Pooled** because Contoso is starting with a new shared environment for some of its users. Default desktop users can be left blank. Move on to configure the virtual machines.

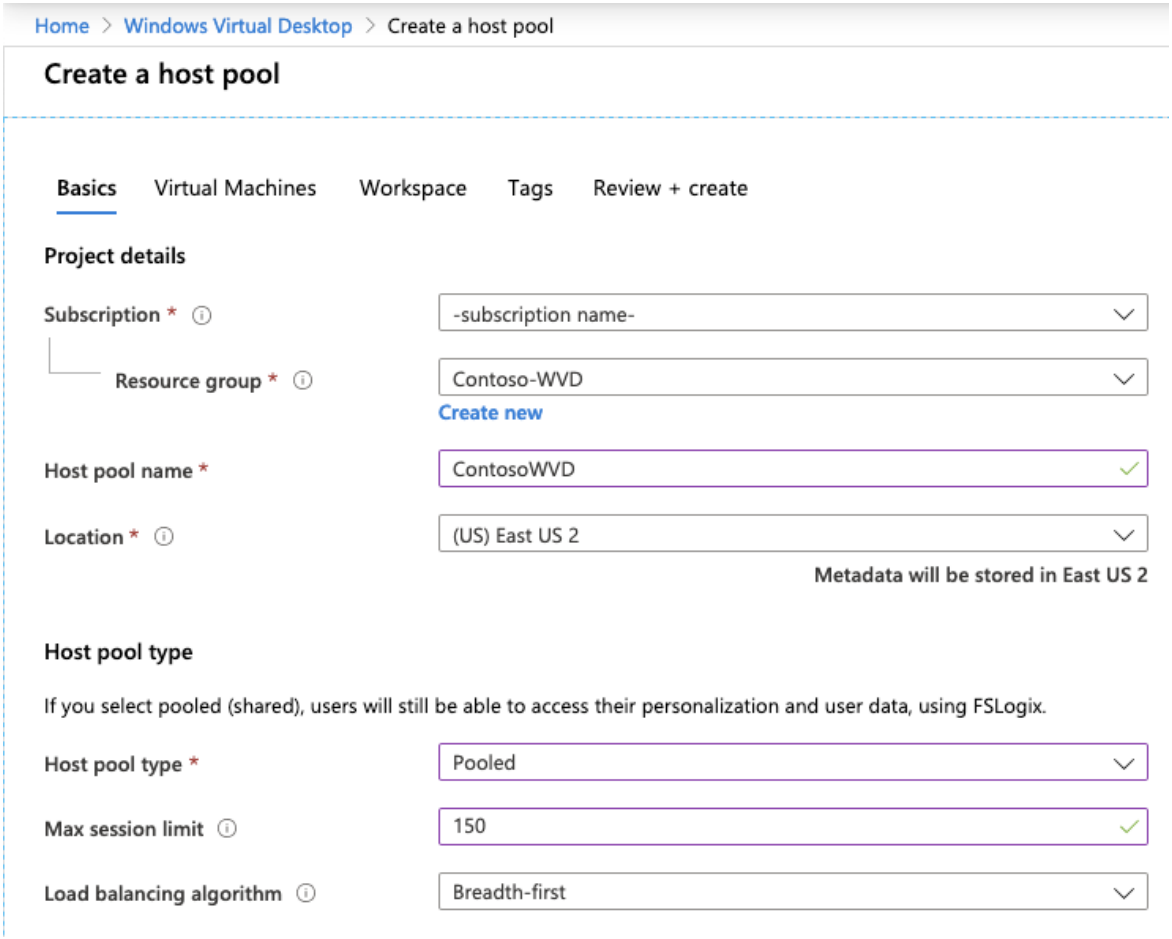
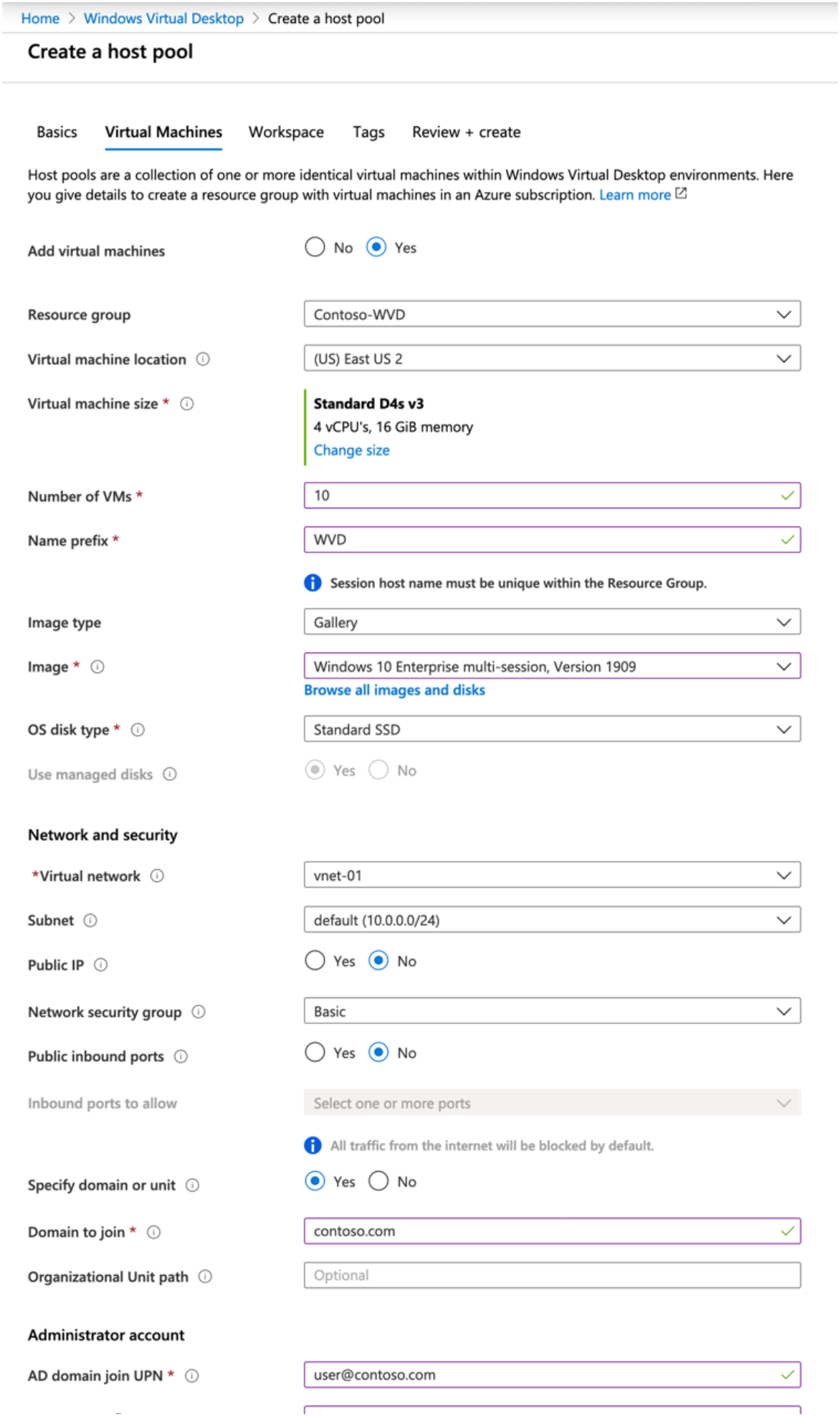


Figure 12: Prerequisites for configuring virtual machines.

- Contoso configures the VM and chooses a custom size by selecting **Change size** or using the default.
- Azure Virtual Desktop is chosen as the VM name prefix for these pooled desktops.
- Because Contoso is creating the pooled servers to use the new Windows 10 Enterprise multi-session functionality for the virtual machine settings, leave the image source set to **Gallery**. This option enables Contoso to select the Windows 10 Enterprise multi-session image for the VMs.
- Based on the personas of the users from the Lakeside assessment, Contoso sets the total users to **150**.

- Other settings include the disk type, an Active Directory domain join UPN field, an admin password, an optional OU path to which machines are added, the virtual network, and a subnet for adding servers.



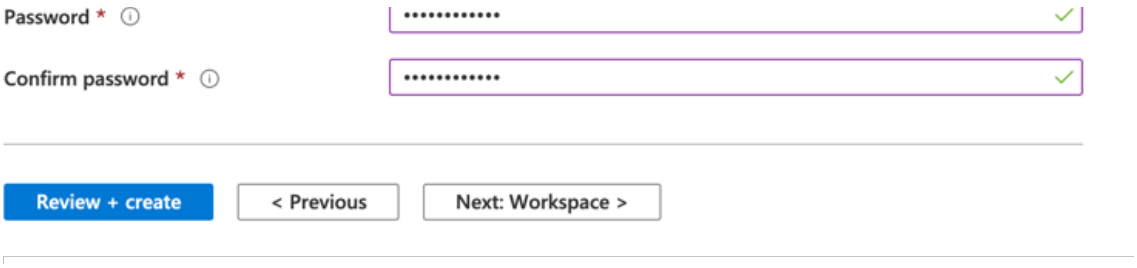


Figure 13: Configuring virtual machines.

> **NOTE**
>
> Contoso can't create a new virtual network at this step. Before reaching this step, Contoso should have already created a virtual network that has access to Active Directory.

> **NOTE**
>
> Contoso can't use a user account that requires multifactor authentication in this step. If Contoso plans to use multifactor authentication for its users, it will need to create a service principal for this purpose.

3. Contoso performs one more validation of the Azure Virtual Desktop settings, and creates the new environment of pooled Azure Virtual Desktop virtual machines.

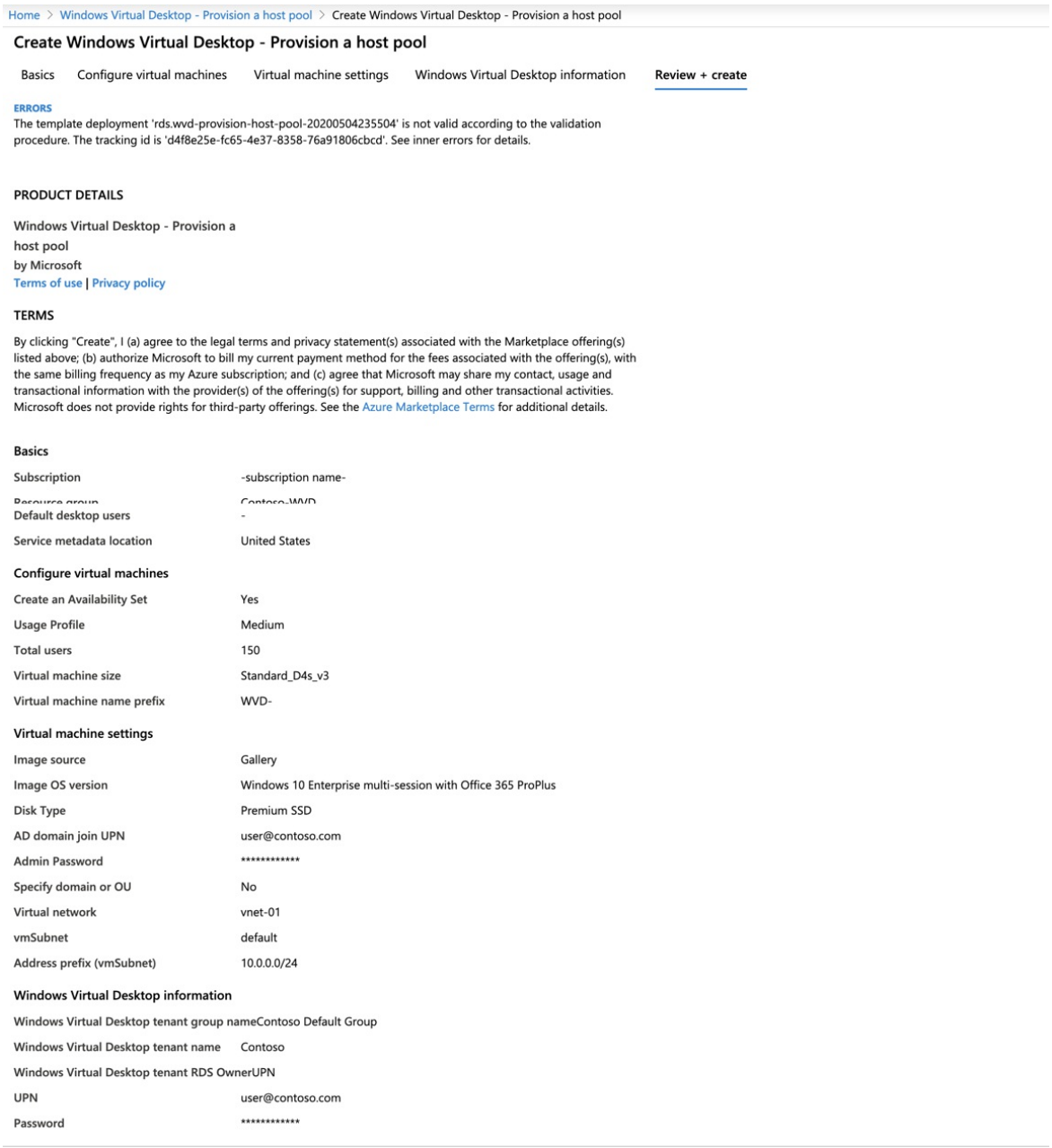


Figure 14: Reviewing and creating virtual machines.

Step 3: Convert the UPDs to FSLogix profile containers

Because Azure Virtual Desktop doesn't support user profile disks (UPDs), Contoso needs to convert all the UPDs to FSLogix via the FSLogixMigration PowerShell module.

After Contoso imports the FSLogixMigration module, it runs the following PowerShell cmdlets to migrate from UPDs to FSLogix.

> **IMPORTANT**
> The PowerShell modules for Hyper-V, Active Directory, and Pester are prerequisites to running the cmdlets to convert UPDs to FSLogix.

A UDP conversion:

```
Convert-RoamingProfile -ParentPath "C:\Users\" -Target "\\Server\FSLogixProfiles$" -MaxVHDSize 20 -
VHDLogicalSectorSize 512
```

A roaming profile conversion:

```
Convert-RoamingProfile -ProfilePath "C:\Users\User1" -Target "\\Server\FSLogixProfiles$" -MaxVHDSize 20 -
VHDLogicalSectorSize 512 -VHD -IncludeRobocopyDetails -LogPath C:\temp\Log.txt
```

At this point, the migration has enabled using pooled resources with Windows 10 Enterprise multi-session. Contoso can begin to deploy the necessary applications to the users who will use Windows 10 Enterprise multi-session.

But now Contoso must migrate the persistent virtual machines to Azure.

Step 4: Replicate and persist VMs to Azure Virtual Desktop

The next step in the migration process for Contoso is to migrate its persistent virtual machines to Azure Virtual Desktop. To do this, Contoso goes back to the Azure Migrate: Server Migration job it created at the beginning of the process.

1. Contoso starts by selecting **Discover** in the Azure Migrate: Server Migration tools.

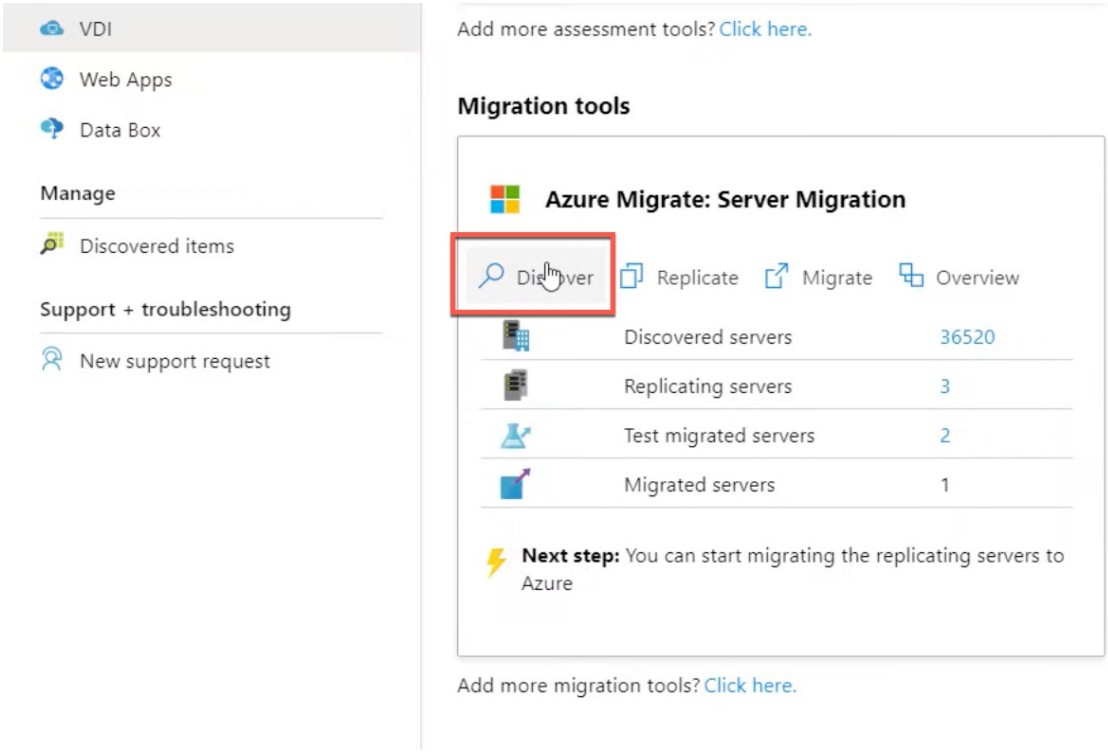


Figure 15: Discovering a server migration.

2. Contoso converts an appliance in its environment that's going to manage the replication of the machines to Azure Virtual Desktop. Ensure that the target region is set to `East US 2`, where the Azure Virtual Desktop environment was created.

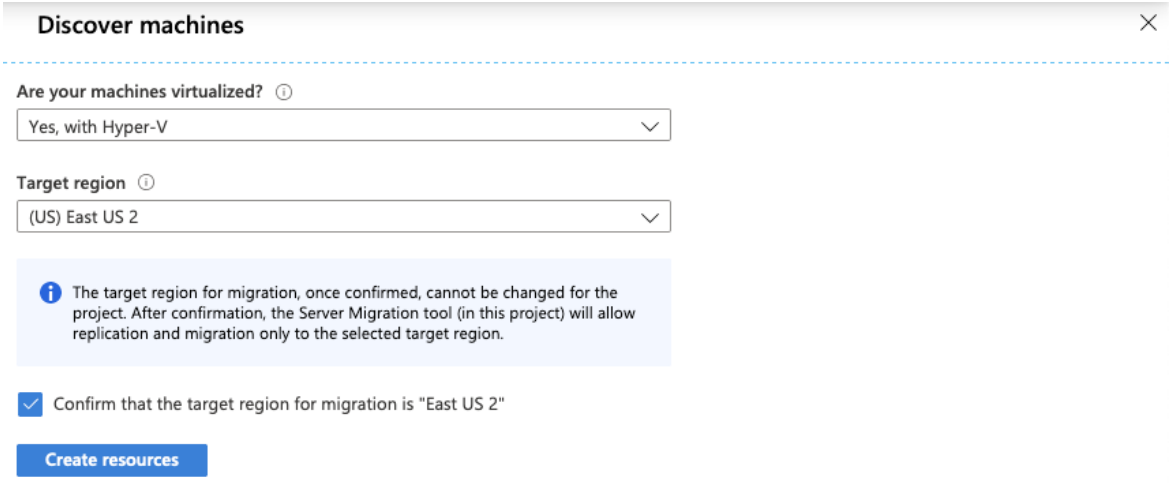


Figure 16: Converting an appliance.

3. The replication provider is downloaded, installed, and registered to the Azure Migrate project to start the replication to Azure.

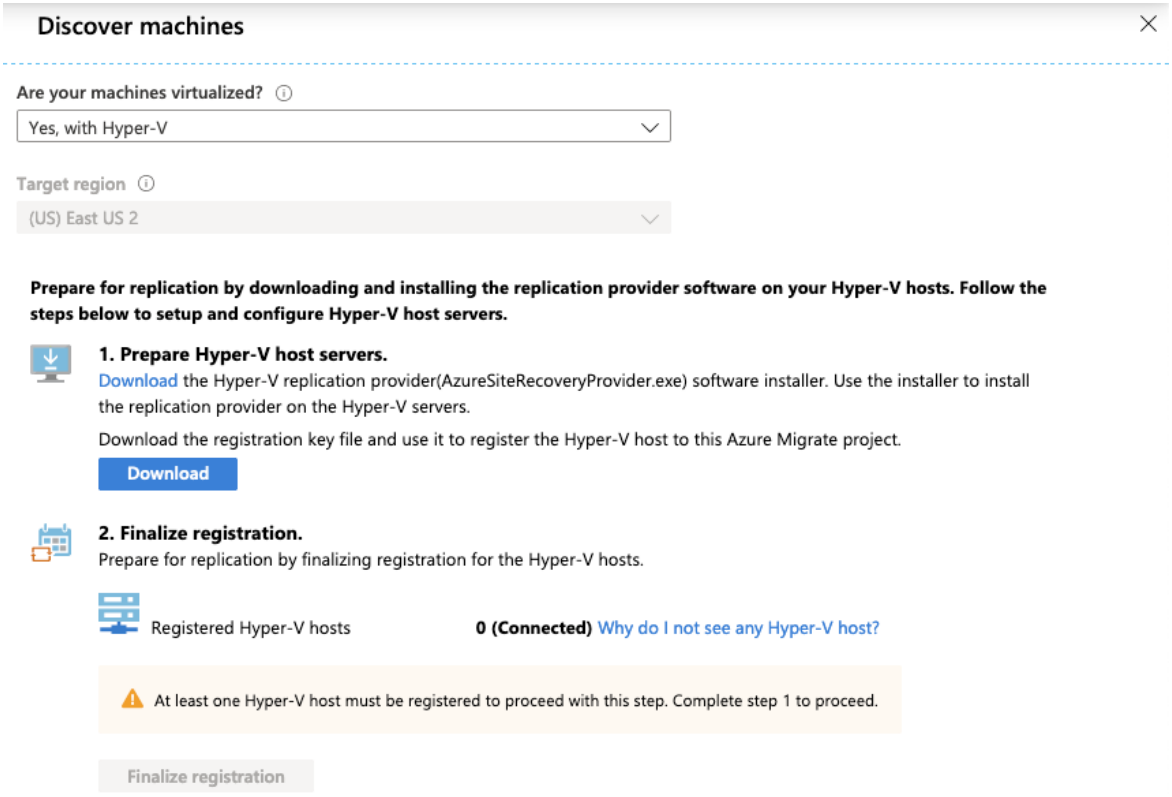


Figure 17: Prerequisites for replicating to Azure.

4. The replication of the hosts into Azure Blob Storage is now started. Contoso can continue to let the replication occur until it's ready to test the VMs and then migrate them into production.
    - As machines start running in Azure, Contoso installs the Azure Virtual Desktop VM agent on each machine.
    - As a part of the installation, enter the registration token for the Azure Virtual Desktop environment to associate the server with the correct environment.
5. The registration token can be obtained by using the following commands:

```
Export-RDSRegistrationInfo -TenantName "Contoso" -HostPoolName "ContosoWVD" | Select-Object -
ExpandProperty Token > .\registration-token.txt
```

> **NOTE**
>
> Contoso can also automate this process by using `msiexec` commands and passing in the registration token as a variable.

6. As the last step before the final migration, Contoso selects the **Users** item in the Azure Virtual Desktop settings to map the servers to their respective users and groups.

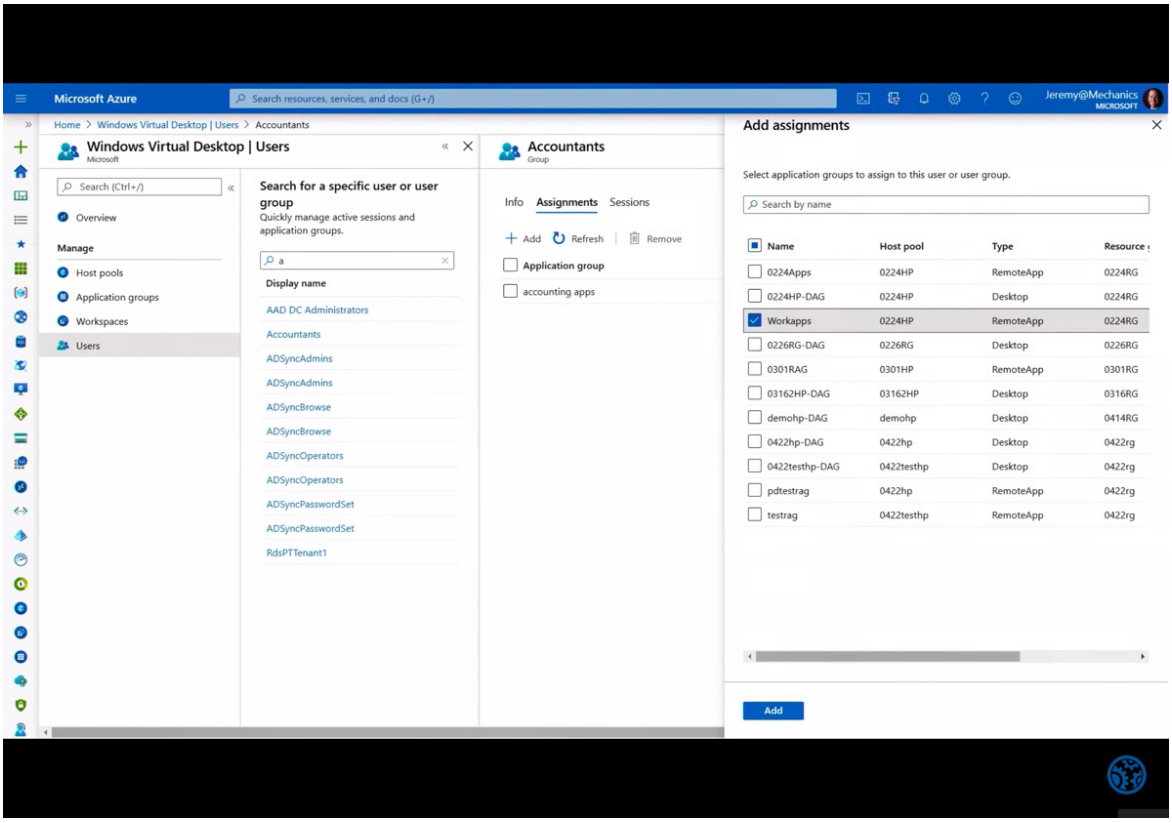


Figure 18: The last step prior to the final migration.

After host pools are assigned to users, Contoso finalizes the migration of those machines and continues to gradually migrate the rest of the on-premises VDI hosts to Azure.

Review the deployment

With the virtual desktops and application servers now running in Azure, Contoso now needs to fully operationalize and secure the deployment.

Security

The Contoso security team reviews the Azure VMs to determine any security issues. To control access, the team reviews the network security groups (NSGs) for the VMs. NSGs are used to ensure that only traffic allowed to the application can reach it. The team also considers securing the data on the disk by using Azure Disk Encryption and Azure Key Vault.

For more information, see Security best practices for IaaS workloads in Azure.

Business continuity and disaster recovery

For business continuity and disaster recovery (BCDR), Contoso backs up the data on the VMs by using Azure Backup to keep data safe. For more information, see An overview of Azure VM backup.

Licensing and cost optimization

- Microsoft 365 licenses are used for the desktop deployments.

- Contoso will enable Azure Cost Management + Billing to help monitor and manage the Azure resources.
- Contoso has existing licensing for its VMs and will take advantage of the Azure Hybrid Benefit for application servers. Contoso will convert the existing Azure VMs to take advantage of this pricing.

Conclusion

In this article, Contoso moved its RDS deployment to Azure Virtual Desktop hosted in Azure.

VMware host migration best practices for Azure

3/6/2021 • 2 minutes to read • Edit Online

Migrating VMware host to Azure can accelerate the standard methodology outlined in the Cloud Adoption Framework, and pictured here.

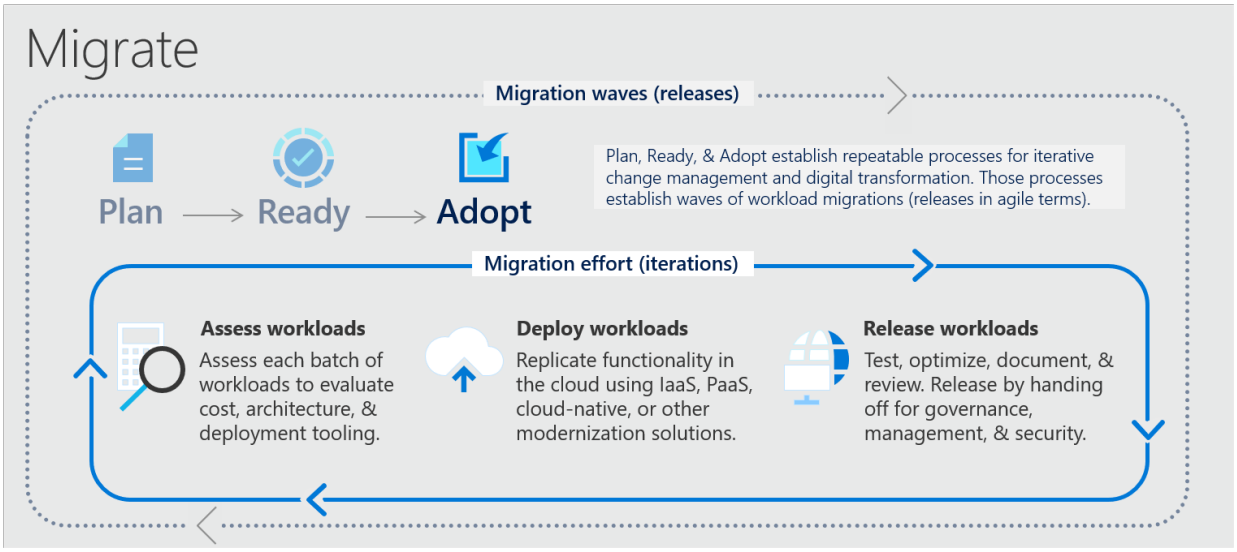


Figure 1

The table of contents on the left outlines best practices across multiple Microsoft web properties. These best practices can guide the execution of your VMware host migration to Azure VMware Solution. Bookmark this page to quickly reference the full list of best practices.

SQL Server migration best practices for Azure

3/6/2021 • 2 minutes to read • Edit Online

Migrating SQL Server to Azure can accelerate the standard methodology outlined in the Cloud Adoption Framework. The process is shown in the following diagram.

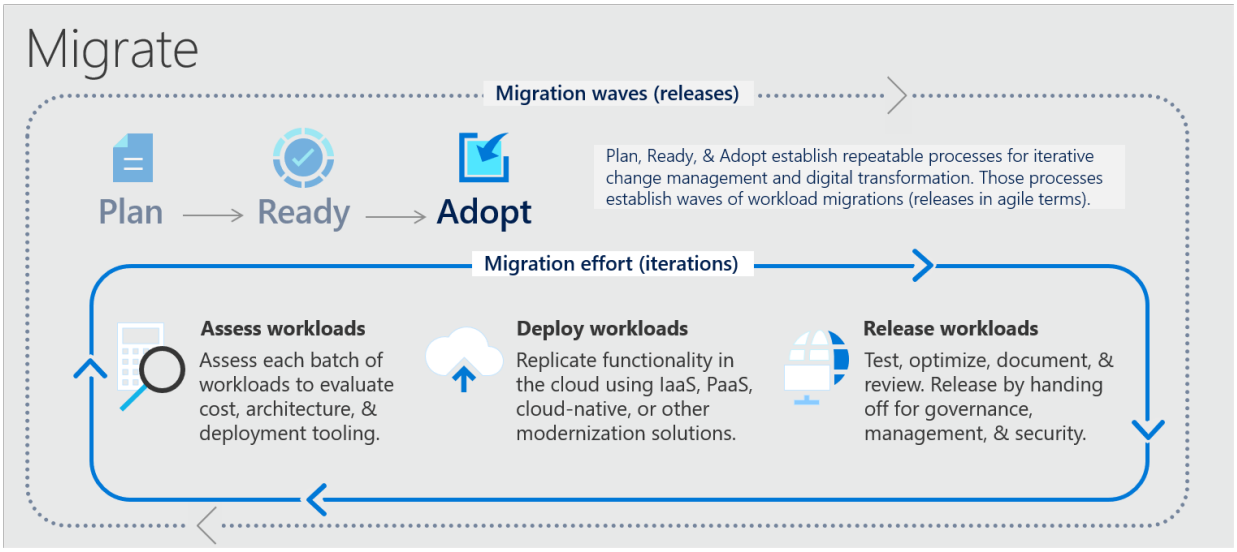


Figure 1

The table of contents on the left outlines best practices that can guide the execution of your SQL Server migration. You can migrate by using Azure Database Migration Guide, Azure Database Migration Service, or other tools. Bookmark this page for quick reference to the full list of best practices.

Azure Stack: A strategic option for running Azure in your datacenter

5/18/2021 • 2 minutes to read • Edit Online

Microsoft takes a cloud-first approach to application and data storage. The priority is to move applications and data to one or more of the hyperscale clouds, including the global Azure option or a sovereign, locale-specific cloud such as Azure Germany or Azure Government.

Azure Stack Hub acts as another instance of a sovereign cloud, whether it's operated by customers in their datacenters or consumed through a cloud service provider. However, Azure Stack Hub is not a hyperscale cloud, and Microsoft doesn't publish or support any service-level agreements for Azure Stack Hub.

Understand your cloud journey

Each organization has a unique journey to the cloud, depending on its history, business specifics, culture and, maybe most importantly, its starting point. Its journey provides many options, features, and functionalities. It also presents opportunities to improve its existing governance and operations, implement new ones, and even redesign its applications to take advantage of the cloud architectures.

Your organization's journey might identify clear benefits to using the cloud as part of your IT and business strategy. However, your journey could identify equally strong motivations to keep the cloud in your existing datacenter, at least for now. If you're facing these two competing drivers, you don't have to choose. At its core, Azure Stack Hub is infrastructure as a service (IaaS). It also provides platform as a service (PaaS) services that allow you to run a subset of Azure services in your own datacenter.

Azure Stack Hub in your strategy

Azure Stack Hub provides an alternative approach to the migration of existing applications that run on either physical servers or existing virtualization platforms. By moving these workloads to an Azure Stack Hub deployment IaaS environment, teams can benefit from smoother operations, self-service deployments, standardized hardware configurations, and Azure consistency. Using Azure Stack Hub for modernization or innovation support enables your teams to prepare your applications and workloads to take full advantage of the cloud.

By following a consistent practice for cloud adoption across Azure and Azure Stack Hub, you can apply the same governance and operations models to assets in the public cloud or your own datacenter. Azure Stack Hub uses the same Azure Resource Manager model as Azure, enabling that single-pane of glass view for your solutions.

Compare Azure with Azure Stack Hub

There are some differences between Azure and Azure Stack Hub. Some are very visible, and others can't be seen until late in the implementation cycle. Be aware of the following differences:

- Azure offers near limitless capacity. Azure Stack Hub is built on physical hardware in your datacenter, which leads to capacity limitations.
- API versions and authentication mechanisms might be slightly different between Azure and Azure Stack Hub.
- Azure Stack Hub differs in *who* operates the cloud, which affects the level of workload operations.
- Consider which part of the Azure Stack Hub service the Azure Stack Hub operator runs on, because that determines whether the customer calls a service PaaS or software as a service (SaaS).

Other differences will be called out in other Azure Stack Hub articles at various points in the cloud adoption lifecycle.

Next steps

For guidance on specific elements of the cloud adoption journey, see:

- Plan for Azure Stack Hub migration
- Environmental readiness
- Assess workloads for Azure Stack Hub
- Deploy workloads to Azure Stack Hub
- Govern Azure Stack Hub
- Manage Azure Stack Hub

Plan your Azure Stack Hub migration

3/6/2021 • 2 minutes to read • Edit Online

This article assumes that you've reviewed how to integrate Azure Stack into your cloud strategy and that your journey aligns with the examples in that article.

Before you move directly into your organization's migration efforts, it's important to set expectations appropriately about Azure and Azure Stack Hub. Doing so can help avoid pitfalls or setbacks later in the project. The key to a successful implementation is a good understanding of when to use Azure and when to use Azure Stack Hub.

Digital estate alignment

Alignment of your organization's digital estate starts with a few simple questions to ask when you complete your digital estate rationalization.

- What is the motivation, such as regulatory requirements, data gravity, or compliance needs, to keep applications and data on-premises?
- Which specific regulatory or compliance requirements affect the decision to stay in the datacenter?
- How will data privacy affect data migration?
- Is migration defined as a modernization journey?
- If so, have you defined the next steps and the goals required after the migration?
- What are the service-level agreement, recovery point objective, recovery time objective, and availability requirements?

For some workloads, your answers to these questions will fuel conversations about the value of Azure versus Azure Stack Hub for that workload.

Assessment best practices

By applying the best practice for assessing a digital estate with Azure Migrate, you can accelerate the assessment and alignment of the workloads and assets in your digital estate. This best practice provides insight into your full IT portfolio. It also helps identify technical requirements for capacity, scale, and configuration to guide your migration.

By using proper assessment data, your cloud adoption team can make wiser choices and establish clearer priorities when they evaluate options for public or private cloud platforms in Azure.

Planning best practices

The following resources can help your team understand the differences between Azure and Azure Stack Hub:

- Azure Stack overview and roadmap
- Azure Stack capacity planning
- Azure Stack Hub datacenter integration walkthrough
- Azure Stack virtual machine features

When you understand the best platform for each workload, you can integrate your decisions into a cloud adoption plan to manage public and private cloud migrations as one aligned effort.

Next steps

For guidance on specific elements of the cloud adoption journey, see:

- Ready your cloud environment for Azure Stack Hub migration
- Assess workloads for Azure Stack Hub
- Deploy workloads to Azure Stack Hub
- Govern Azure Stack Hub
- Manage Azure Stack Hub

Ready your cloud environment for Azure Stack Hub migration

3/6/2021 • 2 minutes to read • Edit Online

This article assumes that you've decided to integrate Azure Stack into your cloud strategy and you've developed a plan for Azure Stack Hub migration.

Assess the infrastructure dependencies that must be addressed first:

- Identity
- Connectivity
- Security
- Encryption

Hybrid environment configuration

In a hybrid environment, some parts of your IT portfolio are in the Azure public cloud and others are in your Azure Stack Hub private cloud. To develop such an environment, you'll first want to configure a few basic elements in the public cloud. To begin establishing landing zones in the public cloud, see Ensure that your environment is prepared for cloud adoption.

Landing zone and cloud platform connections: During the process, ensure that you have a stable network connection between your current datacenter and Azure. After you've established the network connection, test the latency, bandwidth, and reliability of the connection to Azure.

Governance and operations: When you migrate to both clouds, you need to make a few early decisions that will affect the environment. By applying best practices, you build on cloud-native operations and governance tools that run in the public cloud. This approach reduces the cost of running expensive systems in your datacenter or consuming capacity on your Azure Stack Hub deployment. When you migrate to either form of the cloud, you need to either follow best practices or continue using existing systems for operations, governance, and change management.

Private cloud environment

If you choose to only use the private cloud version of Azure, an Azure Stack Hub deployment, you'll need to consider the same decision points:

On-premises governance and operations: The best practice still suggests using the cloud-native operations and governance tools found in the public cloud version of Azure. It's important to evaluate this best practice early and determine whether it's applicable to your scenario.

Landing zone and cloud platform connections: If your workload migrations will reside in your Azure Stack Hub deployment, it will be important to document and test the latency, bandwidth, and reliability of the network routes between end users and your Azure Stack appliance.

Next steps

For guidance on specific elements of the cloud adoption journey, see:

- Assess workloads for Azure Stack Hub
- Deploy workloads to Azure Stack Hub

- Govern Azure Stack Hub
- Manage Azure Stack Hub

Assess workloads for Azure Stack Hub migration

3/6/2021 • 2 minutes to read • Edit Online

This article assumes that you've decided to integrate Azure Stack into your cloud strategy, you've developed a plan for Azure Stack Hub migration, and your environment is ready for migration.

During the rationalization of your organization's digital estate in the Plan methodology, each workload was discovered and inventoried, and initial decisions were made based on quantitative data. Before you deploy each workload, it's important to validate the data and the decisions with qualitative data.

Placement

The first data point to consider is placement. That is, will this workload be migrated to your public cloud, private cloud, or some other cloud platform, such as a sovereign cloud or service provider's Azure environment?

The information in each of the following sections can help validate your decisions about placement. The information will also help surface data that will be useful during the deployment of your workloads.

Stakeholder value

Evaluate the value of migrating this workload with business and IT stakeholders:

- Less friction: short-term focus, limited long-term viability.
- More friction: long-term investment, easier to iterate and continue to modernize.
- A balance of the two.

Governance, risk, and compliance

Assess the impact of regulatory, compliance, and privacy requirements:

- What data can reside on Azure and what data needs to stay on-premises?
- Who can manage the underlying platform?
- Is the data dependent on the location?
- Are there expiration dates for storing the data?

Success metrics

Determine success metrics and availability tolerances:

- Performance
- Availability
- Resiliency
- Deployment or migration approach

Licensing

Assess the impact of licensing and support:

- Are there product licensing restrictions that will limit transformation?
- Is the application or dataset supportable in the new environment?
- Are there third-party software vendors who need to provide support statements?

Operations requirements

- Avoid duplication of effort and optimize service-level agreements (SLAs) by examining the correlation between IT-managed cloud services and application-specific services.
- Consider the automation that's required to orchestrate the provisioning of services during deployment and migration of applications.
- To help meet your operations requirements, consider scalability and availability services such as pay per use, availability sets, virtual machine scale sets, network adapters, and the ability to add and resize VMs and disks.

Monitoring

- Monitor system health and operational status and performance by using well-defined metrics that form the basis of the SLAs that you offer your end users.
- Check security and compliance, evaluating how well the cloud environment meets the regulatory and compliance requirements that are imposed by the application.
- What are the processes for backup/restore and replication/failover?
- Find data-protection services for infrastructure as a service, platform as a service, and software as a service resources.
- Incorporate multiple vendors, technologies, and capabilities to achieve a comprehensive protection strategy.

Next steps

For guidance on specific elements of the cloud adoption journey, see:

- Assess workloads for Azure Stack Hub
- Deploy workloads to Azure Stack Hub
- Govern Azure Stack Hub
- Manage Azure Stack Hub

Deploy workloads to Azure Stack Hub

5/20/2021 • 2 minutes to read • Edit Online

By using Azure Stack, your organization can run its own instance of Azure in its datacenter. Organizations include Azure Stack in their cloud strategy because it helps them handle situations when the public cloud won't work for them. The three most common reasons to use Azure Stack are:

- Poor network connectivity to the public cloud.
- Regulatory or contractual requirements.
- Back-end systems that can't be exposed to the internet.

Infrastructure as a service deployment

Regardless of the reason to deploy infrastructure as a service (IaaS), deployment to Azure Stack Hub is similar to any other IaaS deployment. People often think of IaaS only as virtual machines (VMs), but IaaS is more than that. When you deploy a VM in Azure or Azure Stack, the machine comes with a software-defined network, including Domain Name System, public IPs, firewall rules (also called network security groups), and many other capabilities. The VM deployment also creates disks for your VMs on software-defined storage by using Azure Blob Storage.

For deeper guidance on deploying VMs to Azure Stack, see the Azure Stack compute overview.

Platform as a service deployment

In the cloud, all platform as a service (PaaS) resources run on some form of infrastructure service, such as a VM. However, Azure services obfuscate those backend resources so you don't have to manage them. The obfuscation and coordination of those infrastructure resources is managed by Azure Resource Manager. You may have seen one aspect of Resource Manager when deploying to Azure using an Azure Resource Manager template. Those templates tell Azure which resource provider you want to invoke and how you want your resources to be configured.

When the cloud runs in your datacenter, your stack hub administrators will need to be somewhat familiar with the layers of obfuscation. Before your users or developers can use a PaaS resource, the Azure Stack Hub administrator will need to install the resource provider from the marketplace. Those resource providers allow your instance of Azure Stack Hub to replicate the resource provider functionality of Azure in your stack instance. For more information on deploying Azure Stack Hub resource providers, see the Azure Stack IaaS blog series.

Deploy workloads

After the Azure Stack Hub administrator has properly configured your stack instance, migrations can continue as they would with most other Azure migration efforts. By using Azure Stack, your team can run any of the following types of migration:

- Ethereum blockchain network
- AKS engine
- Azure Cognitive Services
- C# ASP.NET web app
- Linux VM
- Java web app

Additional considerations during migration

The following resources can help your team during migration and modernization:

- Scalability and availability services such as pay per use, availability sets, virtual machine scale sets, network adapters, and the ability to add and resize VMs and disks.
- Storage capacity, including the ability to upload and download and also capture and deploy VM images.
- The Azure Stack quickstart templates GitHub repository.
- The Azure quickstart templates GitHub repository.

Next steps

For guidance on specific elements of the cloud adoption journey, see:

- Govern Azure Stack Hub
- Manage Azure Stack Hub

Govern an Azure instance in your datacenter

5/20/2021 • 2 minutes to read • Edit Online

Governing hybrid solutions across public and private cloud platforms adds complexity. Because your Azure Stack Hub deployment is your own private instance of Azure running in your datacenter, that complexity is inherently reduced.

The business processes, disciplines, and many of the best practices outlined in the Govern methodology of the Cloud Adoption Framework can still be applied to hybrid governance with Azure Stack Hub. Many cloud-native tools used in the public cloud version of Azure can also be used in your Azure Stack Hub deployment.

Azure Stack Hub governance considerations

The following series of blogs shows how your organization can implement cloud governance concepts for Azure Stack Hub:

- Organizational services such as resource groups, Azure role-based access control (Azure RBAC), change auditing, locks, and tags.
- Security services, including default firewalls, restrictions, VM updates and patch management, and malware status.
- DevOps options, including infrastructure as code, a portal with PowerShell and command-line interface, Azure Application Insights, and integration with Azure DevOps and Jenkins.

Governance toolchain for Azure Stack Hub

For guidance on applying cloud-native governance tools to Azure Stack Hub environments, see:

- Azure Resource Manager templates and Desired State Configuration
- PowerShell
- Azure Policy
- Azure role-based access control

Next steps

For guidance on specific elements of the cloud adoption journey, see:

- Manage Azure Stack Hub

Manage workloads that run on Azure Stack Hub

5/4/2021 • 2 minutes to read • Edit Online

Operations and management of hybrid solutions across public and private cloud platforms is complex and could introduce risk to business operations. Because Azure Stack Hub is your organization's private instance of Azure running in your datacenter, the risk of hybrid operations is greatly reduced.

As outlined in the Manage methodology of the Cloud Adoption Framework, suggested operations management activities focus on the following list of core responsibilities. The same responsibilities hold true for the operations management teams that support Azure Stack Hub.

- **Inventory and visibility:** Create an inventory of assets across multiple clouds. Develop visibility into the run state of each asset.
- **Operational compliance:** Establish controls and processes to ensure that each state is properly configured and running in a well-governed environment.
- **Protect and recover:** Ensure that all managed assets are protected and can be recovered by using baseline management tooling.
- **Enhanced baseline options:** Evaluate common additions to the baseline that might meet business needs.
- **Platform operations:** Extend the management baseline with a well-defined service catalog and centrally managed platforms.
- **Workload operations:** Extend the management baseline to include a focus on mission-critical workloads.

Considerations for Azure Stack Hub operations management

Some of the standard operations management activities require slightly different technical considerations. The following articles outline those considerations.

- Self-service support for your customers, including boot diagnostics, screenshots, serial logs, and metrics.
- Guest management, including virtual machine (VM) extensions, the ability to run custom code, software inventory, and change tracking.
- Business continuity through VM backups and disaster recovery options.

Next steps

After your Azure Stack Hub migration reaches an operational state, you can begin the next iteration of migrations by using Azure Stack Hub or other migration scenarios in the Azure public cloud.

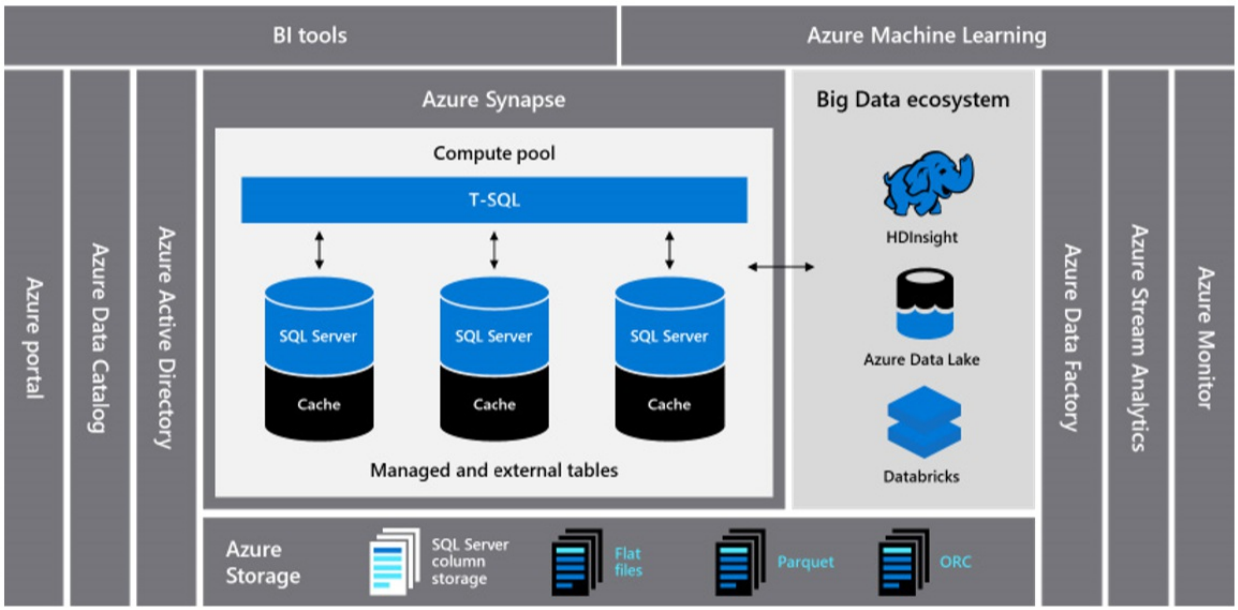
- Plan for Azure Stack Hub migrations
- Environmental readiness
- Assess workloads for Azure Stack Hub
- Deploy workloads to Azure Stack Hub
- Govern Azure Stack Hub
- Manage Azure Stack Hub

Azure Synapse Analytics solutions

3/6/2021 • 2 minutes to read • Edit Online

Current market offerings fall short in meeting an organization's growing needs. Legacy on-premises environments, including Teradata, Netezza, and Oracle Exadata, are expensive, slow to innovate, and inelastic. Organizations that use on-premises systems are now considering taking advantage of innovative cloud, infrastructure as a service, and platform as a service offerings in newer environments like Azure.

Many organizations are ready to take the step of shifting expensive tasks like infrastructure maintenance and platform development to a cloud provider. In Microsoft Azure, an organization has access to a globally available, highly secure, scalable cloud environment that includes Azure Synapse Analytics in an ecosystem of supporting tools and capabilities.



Azure Synapse Analytics provides best-of-class relational database performance through techniques like massively parallel processing and automatic in-memory caching. The results of this approach can be seen in independent benchmarks like the one run recently by GigaOm, which compares Azure Synapse to other popular cloud data warehouse offerings.

Organizations that have already migrated to Azure Synapse Analytics have seen many benefits, including:

- Improved performance and price for performance.
- Increased agility and shorter time to value.
- Faster server deployment and application development.
- Elastic scalability to ensure that you pay only for what you use.
- Improved security and compliance.
- Reduced storage and disaster recovery costs.
- Lower overall TCO and better cost control (operating expenses).

To maximize these benefits, it's necessary to migrate existing data and applications to the Azure Synapse platform. In many organizations, this approach includes migrating an existing data warehouse from a legacy on-premises platform like Teradata, Netezza, or Exadata. Organizations need to modernize their data estate with an analytics offering that is price-performant, rapidly innovative, scalable, and truly elastic. Learn more in the following sections for migration best practices on Teradata, Netezza, and Exadata.

Next steps

- Azure Synapse Analytics solutions for Teradata
- Azure Synapse Analytics solutions for Netezza
- Azure Synapse Analytics solutions for Exadata

Azure Synapse Analytics solutions and migration for Teradata

3/6/2021 • 18 minutes to read • Edit Online

Many organizations are ready to take the step of shifting expensive data warehouse tasks like infrastructure maintenance and platform development to a cloud provider. Organizations are now looking to take advantage of innovative cloud, infrastructure as a service, and platform as a service offerings in newer environments like Azure.

Azure Synapse Analytics is a limitless analytics service that brings together enterprise data warehousing and big data analytics. It gives you the freedom to query data on your terms at scale by using either serverless on-demand or provisioned resources. Learn what to plan for as you migrate a legacy Teradata system to Azure Synapse.

Although Teradata and Azure Synapse are similar in that they're both SQL databases that are designed to use massively parallel processing techniques to achieve high query performance on large data volumes, they have some basic differences:

- Legacy Teradata systems are installed on-premises, and they use proprietary hardware. Azure Synapse is cloud-based and uses Azure compute and storage resources.
- Upgrading a Teradata configuration is a major task that involves extra physical hardware and a potentially lengthy database reconfiguration or dump and reload. In Azure Synapse, compute and storage resources are separate, so you can easily scale up or down independently by using the elastic scalability of Azure.
- Without a physical system to support, you can pause or resize Azure Synapse as needed to reduce resource utilization and cost. In Azure, you have access to a globally available, highly secure, and scalable cloud environment that includes Azure Synapse in an ecosystem of supporting tools and capabilities.

In this article, we look at schema migration, with an objective of obtaining equivalent or increased performance of your migrated Teradata data warehouse and data marts on Azure Synapse. We consider concerns that apply specifically to migrating from an existing Teradata environment.

At a high level, the migration process includes the steps that are listed in the following table:

PREPARATION	MIGRATION	POST-MIGRATION
• Define scope: What do we want to migrate? • Build an inventory of data and processes to migrate. • Define any data model changes. • Identify the best Azure and third-party tools and features to use. • Train staff early on the new platform. • Set up the Azure target platform.	• Start small and simple. • Automate where possible. • Use Azure built-in tools and features to reduce the migration effort. • Migrate metadata for tables and views. • Migrate relevant historical data. • Migrate or refactor stored procedures and business processes. • Migrate or refactor ETL/ELT incremental load processes.	• Monitor and document all stages of the migration process. • Use experience gained to build a template for future migrations. • Reengineer the data model, if necessary, by using the new platform's performance and scalability. • Test applications and query tools. • Benchmark and optimize query performance.

When you migrate from a legacy Teradata environment to Azure Synapse, in addition to the more general subjects that are described in the Teradata documentation, you must consider some specific factors.

Initial migration workload

Legacy Teradata environments typically evolve over time to encompass multiple subject areas and mixed workloads. When you're deciding where to start on an initial migration project, it makes sense to choose an area that:

- Proves the viability of migrating to Azure Synapse by quickly delivering the benefits of the new environment.
- Allows in-house technical staff to gain experience with new processes and tools so that they can use them to migrate other areas.
- Creates a template based on the current tools and processes to use in additional migrations from the source Teradata environment.

A good candidate for an initial migration from a Teradata environment is one that implements a Power BI/analytics workload rather than an OLTP workload. The workload should have a data model that can be migrated with minimal modifications, such as a star or snowflake schema.

For size, it's important that the data volume you migrate in the initial exercise is large enough to demonstrate the capabilities and benefits of the Azure Synapse environment with a short time to demonstrate value. The size that typically meets the requirements is in the range of 1 to 10 terabytes (TB).

An approach for the initial migration project that minimizes risk and implementation time is to confine the scope of the migration to data marts. In Teradata, a good example is the OLAP database part of a Teradata data warehouse. This approach is a good starting point because it limits the scope of the migration, and it often can be achieved on a short timescale.

An initial migration scope of data marts only doesn't address broader concerns like how to migrate ETL and historical data. You must address these areas in later phases and backfill the migrated data mart layer with the data and processes that are required to build them.

Lift-and-shift approach vs. phased approach

Regardless of the drivers and scope you choose for your migration, you can choose from two general types of migrations:

- **Lift-and-shift approach:** In this approach, the existing data model, such as a star schema, is migrated unchanged to the new Azure Synapse platform. The emphasis is on minimizing risk and the time it takes to migrate by reducing the work that's required to achieve the benefits of moving to the Azure cloud environment.

  This approach is a good fit for existing Teradata environments in which a single data mart is to be migrated and if the data is already in a well-designed star or snowflake schema. This approach is a good choice also if you have time and cost pressures to move to a more modern cloud environment.

- **Phased approach that incorporates modifications:** If your legacy warehouse has evolved over time, you might need to reengineer the data warehouse to maintain the required performance or to support new data sources like IoT streams. Migrating to Azure Synapse for the well-known benefits of a scalable cloud environment might be considered part of the reengineering process. This process might include changing the underlying data model, such as moving from an Inmon model to Azure Data Vault.

  The approach we recommend is to initially move the existing data model as-is to Azure. Then, take advantage of the performance and flexibility of Azure services to apply the reengineering changes without affecting the existing source system.

Virtual machine colocation to support migration

An optional approach to migrate from an on-premises Teradata environment takes advantage of inexpensive cloud storage and elastic scalability in Azure. First, you create a Teradata instance on an Azure virtual machine that's colocated with the target Azure Synapse environment. Then, you use a standard Teradata utility like

Teradata Parallel Transporter or a third-party data replication tool like Qlik Replicate (formerly by Attunity) to efficiently move the subset of Teradata tables that you're migrating to the VM instance. All migration tasks take place in the Azure environment.

This approach has several benefits:

- After the initial replication of data, the source system isn't affected by other migration tasks.
- Familiar Teradata interfaces, tools, and utilities are available in the Azure environment.
- After the migration shifts to the Azure environment, you don't have any potential issues with network bandwidth availability between the on-premises source system and the cloud target system.
- Tools like Azure Data Factory efficiently call utilities like Teradata Parallel Transporter to migrate data quickly and easily.
- The migration process is orchestrated and controlled entirely from within the Azure environment.

Metadata migration

It makes sense to automate and orchestrate the migration process by using the capabilities of the Azure environment. This approach minimizes impact on the existing Teradata environment, which might already be running close to full capacity.

Azure Data Factory is a cloud-based data integration service. You can use Data Factory to create data-driven workflows in the cloud to orchestrate and automate data movement and data transformation. Data Factory pipelines can ingest data from disparate datastores. Then, they process and transform the data by using compute services like Azure HDInsight for Apache Hadoop and Apache Spark, Azure Data Lake Analytics, and Azure Machine Learning.

Start by creating metadata that lists the data tables you want to migrate along with their locations. Then, use Data Factory capabilities to manage the migration process.

Design differences between Teradata and Azure Synapse

As you plan your migration from a legacy Teradata environment to Azure Synapse, it's important to consider the design differences between the two platforms.

Multiple databases vs. a single database and schemas

In a Teradata environment, you might have multiple, separate databases for different parts of the overall environment. For example, you might have a separate database for data ingestion and staging tables, a database for core warehouse tables, and another database for data marts (sometimes called a *semantic layer*). Processing separate databases as ETL/ELT pipelines in Azure Synapse might require implementing cross-database joins and moving data between the separate databases.

The Azure Synapse environment has a single database. Schemas are used to separate tables into logically separate groups. We recommend that you use a set of schemas in Azure Synapse to mimic any separate databases you migrate from Teradata.

If you use schemas in the Teradata environment, you might need to use a new naming convention to move the existing Teradata tables and views to the new environment. For example, you might concatenate the existing Teradata schema and table names into the new Azure Synapse table name, and then use schema names in the new environment to maintain the original separate database names.

Another option is to use SQL views over the underlying tables to maintain their logical structures. There are some potential downsides to using SQL views:

- Views in Azure Synapse are read-only, so you must make any updates to the data on the underlying base tables.
- If layers of views already exist, adding another layer of views might affect performance.

Tables

When you migrate tables between different technologies, you physically move only raw data and the metadata that describes it between the two environments. You don't migrate database elements like indexes from the source system because they might not be needed, or they might be implemented differently in the new environment.

However, understanding where performance optimizations like indexes have been used in the source environment can be a helpful indication of where you might optimize performance in the new environment. For example, if a non-unique secondary index was created in the source Teradata environment, you might conclude that it would be advantageous to create a nonclustered index in the migrated Azure Synapse environment, or that using other native performance optimization techniques like table replication might be preferable to creating a like-for-like index.

High availability database

Teradata supports data replication across nodes via the `FALLBACK` option. Table rows that reside physically on a node are replicated to another node within the system. This approach guarantees that data isn't lost if a node fails, and it provides the basis for failover scenarios.

The goal of the high-availability architecture in Azure SQL Database is to guarantee that your database is up and running 99.99% of the time. You don't need to consider how maintenance operations and outages might affect your workload. Azure automatically handles critical servicing tasks like patching, backups, and Windows and SQL upgrades, and unplanned events like underlying hardware, software, or network failures.

Snapshots are a built-in feature of the service that creates restore points in Azure Synapse. Snapshots provide automatic backup for data storage in Azure Synapse. You don't have to enable the capability. Currently, individual users can't delete automatic restore points that the service uses to maintain SLAs for recovery.

Azure Synapse takes snapshots of the data warehouse throughout the day. The restore points that it creates are available for seven days. The retention period can't be changed. Azure Synapse supports an eight-hour recovery point objective. You can restore your data warehouse in the primary region from any of the snapshots taken in the past seven days. Other user-defined options are available if your organization needs more granular backups.

Unsupported Teradata table types

Teradata includes support for special table types for time series and temporal data. Azure Synapse doesn't directly support the syntax and some of the functions for these table types, but you can migrate the data into a standard table that has the required data types and indexing or partitioning on the date/time column.

Teradata implements temporal query functionality by using query rewriting to add filters to a temporal query to limit the applicable date range. If you use temporal queries in the source Teradata environment and you want to migrate it, you must add filters to the relevant temporal queries.

The Azure environment also includes features for complex analytics on time series data at scale through Azure Time Series Insights. Time Series Insights is designed for IoT data analysis applications, and it might be more appropriate for that use case. For more information, see Azure Time Series Insights.

Teradata data type mapping

Some Teradata data types aren't directly supported in Azure Synapse. The following table shows these data types and the recommended approach for handling them. In the table, the Teradata column type is the type that's stored in the system catalog (for example, in `DBC.ColumnsV`).

Use the metadata from the Teradata catalog tables to determine whether any of these data types should be migrated, and then plan for supporting resources in your migration plan. For example, you can use a SQL query like the one in the next section to find any occurrences of unsupported data types that you need to address.

Third-party vendors offer tools and services that can automate migration, including mapping data types

between platforms. If you already use a third-party ETL tool like Informatica or Talend in the Teradata environment, you can use the tool to implement any required data transformations.

SQL Data Manipulation Language (DML) syntax differences

You should be aware of a few differences in SQL Data Manipulation Language (DML) syntax between Teradata SQL and Azure Synapse:

- `QUALIFY`: Teradata supports the `QUALIFY` operator.

  For example:

  ```
  SELECT col1 FROM tab1 WHERE col1='XYZ'
  ```

  Third-party tools and services can automate data-mapping tasks:

  ```
  QUALIFY ROW_NUMBER() OVER (PARTITION by col1 ORDER BY col1) = 1;
  ```

  In Azure Synapse, you can achieve the same result by using the following syntax:

  ```
  SELECT * FROM (SELECT col1, ROW_NUMBER() OVER (PARTITION by col1 ORDER BY col1) rn FROM tab1 WHERE
  c1='XYZ' ) WHERE rn = 1;
  ```

- **Date arithmetic:** Azure Synapse has operators like `DATEADD` and `DATEDIFF`, which you can use on `DATE` or `DATETIME`.

  Teradata supports direct subtraction on dates:

  - `SELECT DATE1 - DATE2 FROM ...`
  - `LIKE ANY` syntax

    Example:

    `SELECT * FROM CUSTOMER WHERE POSTCODE LIKE ANY ('CV1%', 'CV2%', CV3%') ;`.

    The equivalent in Azure Synapse syntax is:

    ```
    SELECT * FROM CUSTOMER WHERE (POSTCODE LIKE 'CV1%') OR (POSTCODE LIKE 'CV2%') OR (POSTCODE LIKE
    'CV3%') ;
    ```

- Depending on system settings, character comparisons in Teradata might not be case-specific by default. In Azure Synapse, these comparisons are always case-specific.

Functions, stored procedures, triggers, and sequences

When you migrate a data warehouse from a mature legacy environment like Teradata, often you need to migrate elements other than simple tables and views to the new target environment. Examples of non-table elements in Teradata that you might need to migrate to Azure Synapse are functions, stored procedures, triggers, and sequences. During the preparation phase of the migration, you should create an inventory of objects to migrate. In the project plan, define the method of handling all objects and allocate the appropriate resources to migrate them.

You might find services in the Azure environment that replace the functionality implemented as functions or stored procedures in the Teradata environment. Usually, it's more efficient to use the built-in Azure capabilities instead of recoding the Teradata functions.

Here's more information about migrating functions, stored procedures, triggers, and sequences:

- **Functions:** Like most database products, Teradata supports system functions and user-defined functions in a SQL implementation. When common system functions are migrated to another database platform

like Azure Synapse, they're usually available in the new environment and can be migrated without change. If system functions have slightly different syntax in the new environment, you usually can automate the required changes.

You might need to recode arbitrary user-defined functions and system functions that have no equivalent in the new environment. Use the languages that are available in the new environment. Azure Synapse uses the popular Transact-SQL language to implement user-defined functions.

- **Stored procedures:** In most modern database products, you can store procedures n the database. A stored procedure typically contains SQL statements and some procedural logic. It might also return data or a status.

  Teradata provides stored procedure language to create stored procedures. Azure Synapse supports stored procedures by using T-SQL. If you migrate stored procedures to Azure Synapse, you must recode them by using T-SQL.

- **Triggers:** You can't create triggers in Azure Synapse, but you can implement triggers in Data Factory.

- **Sequences:** Azure Synapse sequences are handled similarly to how they are handled in Teradata. Use `IDENTITY` columns or SQL code to create the next sequence number in a series.

Metadata and data extraction

Consider the following information when you plan how to extract metadata and data from the Teradata environment:

- **Data Definition Language (DDL) generation:** As described earlier, it's possible to edit existing Teradata `CREATE TABLE` and `CREATE VIEW` scripts to create the equivalent definitions, with modified data types, if necessary. In this scenario, you usually must remove extra Teradata-specific clauses (for example, `FALLBACK`).

  The information that specifies the current table and view definitions is maintained in system catalog tables. System catalog tables are the best source of the information because the tables likely are up to date and complete. User-maintained documentation might not be in sync with current table definitions.

  You can access the information by using views on the catalog, such as `DBC.ColumnsV`. You also can use views to generate the equivalent `CREATE TABLE` Data Definition Language (DDL) statements for the equivalent tables in Azure Synapse.

  Third-party migration and ETL tools also use the catalog information to achieve the same result.

- **Data extraction**

  Migrate the raw data from existing Teradata tables by using standard Teradata utilities like `BTEQ` and `FASTEXPORT`. In a migration exercise, it's generally important to extract the data as efficiently as possible. For recent versions of Teradata, we recommend Teradata Parallel Transporter, a utility that uses multiple parallel `FASTEXPORT` streams to achieve the best throughput.

  You can call Teradata Parallel Transporter directly from Data Factory. We recommend this approach for managing the data migration process, whether the Teradata instance is on-premises or copied to a VM in the Azure environment, as described earlier.

  The data formats we recommend for extracted data are delimited text files (also called *comma-separated values*), optimized row columnar files, or Parquet files.

For detailed information about the process of migrating data and ETL from a Teradata environment, see the Teradata documentation.

Performance-tuning recommendations

The platforms have some differences when it comes to optimization. In the following list of performance-tuning recommendations, lower-level implementation differences between Teradata and Azure Synapse and alternatives for your migration are highlighted:

- **Data distribution options:** In Azure, you can set the data distribution methods for individual tables. The purpose of the functionality is to reduce the amount of data that moves between processing nodes when a query is executed.

  For large table/large table joins, hash distributing in one or both (ideally, both) tables on the join columns helps ensure that join processing can be performed locally because the data rows to be joined are already colocated on the same processing node.

  Azure Synapse provides an additional way to achieve local joins for small table/large table joins (often called a *dimension table/fact table join* in a star schema model). You replicate the smaller table across all nodes, thereby ensuring that any value of the join key for the larger table has a matching dimension row that's locally available. The overhead of replicating the dimension table is relatively low if the tables aren't large. In this case, using the hash distribution approach described earlier is preferable.

- **Data indexing:** Azure Synapse provides various indexing options, but the options are different in operation and usage from indexing options in Teradata. To learn about the indexing options in Azure Synapse, see Design tables in an Azure Synapse pool.

  Existing indexes in the source Teradata environment can provide a useful indication of how data is used and provide an indication of candidate columns for indexing in the Azure Synapse environment.

- **Data partitioning:** In an enterprise data warehouse, fact tables might contain many billions of rows of data. Partitioning is a way to optimize maintenance and querying in these tables. Splitting the tables into separate parts reduces the amount of data processed at one time. Partitioning for a table is defined in the `CREATE TABLE` statement.

  Only one field per table can be used for partitioning. The field that's used for partitioning frequently is a date field because many queries are filtered by date or by a date range. You can change the partitioning of a table after initial load. To change a table's partitioning, re-create the table with a new distribution that uses the `CREATE TABLE AS SELECT` statement. For a detailed description of partitioning in Azure Synapse, see Partition tables in an Azure Synapse SQL pool.

- **Data table statistics:** You can ensure that statistics about data tables are up to date by adding a `COLLECT STATISTICS` step in ETL/ELT jobs or by enabling automatic statistics collection on the table.

- **PolyBase for data loading:** PolyBase is the most efficient method to use to load large amounts of data into a warehouse. You can use PolyBase to load data in parallel streams.

- **Resource classes for workload management:** Azure Synapse uses resource classes to manage workloads. In general, large resource classes provide better individual query performance. Smaller resource classes give you higher levels of concurrency. You can use dynamic management views to monitor utilization to help ensure that the appropriate resources are used efficiently.

Next steps

For more information about implementing a Teradata migration, talk with your Microsoft account representative about on-premises migration offers.

Azure Synapse Analytics solutions and migration for Netezza

5/18/2021 • 17 minutes to read • Edit Online

Many organizations that currently use Netezza data warehouse systems are looking to take advantage of innovative cloud, infrastructure as a service, and platform as a service offerings in newer environments like Azure. Many organizations are ready to take the step of shifting expensive tasks like infrastructure maintenance and platform development to a cloud provider.

Azure Synapse Analytics is a limitless analytics service that brings together enterprise data warehousing and big data analytics. It gives you the freedom to query data on your terms at scale by using either serverless on-demand or provisioned resources. Learn what to plan for as you migrate a legacy Netezza system to Azure Synapse.

Netezza and Azure Synapse are similar in that each is a SQL database that's designed to use massively parallel processing techniques to achieve high query performance on large data volumes. But the two platforms are different in key aspects:

- Legacy Netezza systems are installed on-premises, and they use proprietary hardware. Azure Synapse is cloud-based and uses Azure compute and storage resources.
- Upgrading a Netezza configuration is a major task that involves extra physical hardware and a potentially lengthy database reconfiguration or dump and reload. In Azure Synapse, storage and compute resources are separate. You can use the elastic scalability of Azure to independently scale up or down.
- Without a physical system to support, you can pause or resize Azure Synapse as needed to reduce resource utilization and cost. In Azure, you have access to a globally available, highly secure, and scalable cloud environment that includes Azure Synapse in an ecosystem of supporting tools and capabilities.

In this article, we look at schema migration, with a view to obtaining equivalent or increased performance of your migrated Netezza data warehouse and data marts on Azure Synapse. We consider concerns that apply specifically to migrating from an existing Netezza environment.

At a high level, the migration process includes the steps that are listed in the following table:

PREPARATION	MIGRATION	POST-MIGRATION
• Define scope: What do we want to migrate? • Build an inventory of data and processes to migrate. • Define any data model changes. • Identify the best Azure and third-party tools and features to use. • Train staff early on the new platform. • Set up the Azure target platform.	• Start small and simple. • Automate where possible. • Use Azure built-in tools and features to reduce the migration effort. • Migrate metadata for tables and views. • Migrate relevant historical data. • Migrate or refactor stored procedures and business processes. • Migrate or refactor ETL or ELT incremental load processes.	• Monitor and document all stages of the migration process. • Use experience gained to build a template for future migrations. • Reengineer the data model, if necessary, by using the new platform's performance and scalability. • Test applications and query tools. • Benchmark and optimize query performance.

When you migrate from a legacy Netezza environment to Azure Synapse, you must consider some specific factors, in addition to the more general subjects described in the Netezza documentation.

Initial migration workload

Legacy Netezza environments typically evolve over time to encompass multiple subject areas and mixed workloads. When you are deciding where to start on an initial migration project, it makes sense to choose an area that:

- Proves the viability of migrating to Azure Synapse by quickly delivering the benefits of the new environment.
- Allows in-house technical staff to gain experience with new processes and tools so that they can use them to migrate other areas.
- Creates a template based on the current tools and processes to use in additional migrations from the source Netezza environment.

A good candidate for an initial migration from a Netezza environment that would support these objectives typically is one that implements a Power BI/analytics workload rather than an OLTP workload. The workload should have a data model that can be migrated with minimal modifications, such as a star or snowflake schema.

For size, it's important that the data volume you migrate in the initial exercise is large enough to demonstrate the capabilities and benefits of the Azure Synapse environment with a short time to demonstrate value. The size that typically meets the requirements is in the range of 1 terabyte (TB) to 10 TB.

An approach for the initial migration project that minimizes risk and implementation time is to confine the scope of the migration to data marts. This approach is a good starting point because it clearly limits the scope of the migration and typically can be achieved on a short timescale. An initial migration of data marts only doesn't address broader concerns like how to migrate ETL and historical data. You must address these areas in later phases and backfill the migrated data mart layer with the data and processes that are required to build them.

Lift-and-shift approach vs. phased approach

Regardless of the drivers and scope that you choose for your migration, you can choose from two general types of migration:

- **Lift-and-shift approach:** In this approach, the existing data model, such as a star schema, is migrated unchanged to the new Azure Synapse platform. In this scenario, the emphasis is on minimizing risk and the time it takes to migrate by reducing the work that has to be done to achieve the benefits of moving to the Azure cloud environment.

  This approach is a good fit for existing Teradata environments in which a single data mart is to be migrated, and if the data is already in a well-designed star or snowflake schema. This approach is a good choice also if you have time and cost pressures to move to a more modern cloud environment.

- **Phased approach that incorporates modifications:** When a legacy warehouse has evolved over time, you might need to reengineer the data warehouse to maintain the required performance or to support new data sources like IoT streams. Migrating to Azure Synapse for the well-known benefits of a scalable cloud environment might be considered part of the reengineering process. This process might include changing the underlying data model, such as moving from an Inmon model to Azure Data Vault.

  The approach we recommend is to initially move the existing data model as-is to Azure. Then, take advantage of the performance and flexibility of Azure services to apply the reengineering changes without affecting the existing source system.

Metadata migration

It makes sense to automate and orchestrate the migration process by using the capabilities of the Azure environment. This approach minimizes the effect on the existing Netezza environment, which might already be running close to full capacity.

Azure Data Factory is a cloud-based data integration service. You can use Data Factory to create data-driven workflows in the cloud to orchestrate and automate data movement and data transformation. Data Factory

pipelines can ingest data from disparate datastores, and then process and transform the data by using compute services like Azure HDInsight for Apache Hadoop and Apache Spark, Azure Data Lake Analytics, and Azure Machine Learning. You start by creating metadata to list the data tables you want to migrate, with their locations, and then use Data Factory capabilities to manage the migration process.

Design differences between Netezza and Azure Synapse

As you plan your migration from a legacy Netezza environment to Azure Synapse, it's important to consider the design differences between the two platforms.

Multiple databases vs. a single database and schemas

In a Netezza environment, you might have multiple, separate databases for different parts of the overall environment. For example, you might have a separate database for data ingestion and staging tables, a database for core warehouse tables, and another database for data marts, sometimes called a *semantic layer*. Processing separate databases as ETL/ELT pipelines in Azure Synapse might require implementing cross-database joins and moving data between the separate databases.

The Azure Synapse environment has a single database. Schemas are used to separate tables into logically separate groups. We recommend that you use a set of schemas in the target Azure Synapse to mimic any separate databases that you migrate from Netezza. If you use schemas in the Netezza environment, you might need to use a new naming convention to move the existing Netezza tables and views to the new environment. For example, you might concatenate the existing Netezza schema and table names into the new Azure Synapse table name, and then use schema names in the new environment to maintain the original separate database names.

Another option is to use SQL views over the underlying tables to maintain the logical structures. There are some potential downsides to using SQL views:

- Views in Azure Synapse are read-only, so you must make any updates to the data on the underlying base tables.
- If layers of views already exist, adding another layer of views might affect performance.

Tables

When you migrate tables between different technologies, you physically move only raw data and the metadata that describes it between the two environments. You don't migrate database elements like indexes from the source system because they might not be needed or they might be implemented differently in the new environment.

However, understanding where performance optimizations like indexes have been used in the source environment can be a helpful indication of where you might optimize performance in the new environment. For example, if queries in the source Netezza environment frequently use zone maps, you might conclude that it would be advantageous to create a nonclustered index in the migrated Azure Synapse environment, or that using other native performance optimization techniques like table replication might be preferable to creating a like-for-like index.

Unsupported Netezza database object types

Netezza implements some database objects that aren't directly supported in Azure Synapse. However, Azure Synapse offers methods that you can use to achieve the same functionality in the new environment, as described in the following list:

- **Zone maps:** In Netezza, zone maps are automatically created and maintained for some column types. Zone maps are used at query time on the following column types to restrict the amount of data to be scanned:
  - `INTEGER` columns that are a length of 8 bytes or less

- Temporal columns, including `DATE`, `TIME`, and `TIMESTAMP`
- `CHAR` columns, if they are part of a materialized view and included in the `ORDER BY` clause

You can find out which columns have zone maps by using the nz_zonemap utility. The utility is part of the NZ Toolkit.

Azure Synapse doesn't use zone maps, but you can achieve similar results by using user-defined index types or partitioning.

- **Clustered base tables (CBTs):** In Netezza, the most common CBT is the fact table, which has billions of records. Scanning such a huge table requires a long processing time because a full table scan might be needed to get relevant records. By organizing records in restrictive CBTs, Netezza can group records in the same or nearby extents. The process also creates zone maps that improve performance by reducing the amount of data to scan.

  In Azure Synapse, you can achieve a similar result through partitioning or by using other index types.

- **Materialized views:** Netezza recommends that users create one or more materialized view over large tables that have many columns, and in which only a few columns are regularly used in queries. Materialized views are automatically maintained by the system when data in the base table is updated.

  Currently, Microsoft offers preview support for materialized views, with the same functionality as Netezza, in Azure Synapse.

- **Data type mapping:** Most Netezza data types have a direct equivalent in Azure Synapse. The following table shows the data types and the recommended approaches for mapping the data types.

IBM NETEZZA DATA TYPE	AZURE SYNAPSE SQL V2 DATA TYPE
bigint (int8)	bigint
binary varying(n) (alias varbinary(n))	varbinary(n)
binary(n)	binary(n)
boolean (bool)	bit
bpchar(n)	varchar(n)
byteint (int1)	smallint
character varying (n) (alias varchar(n))	varchar(n)
character(n) (alias char(n))	char(n)
dataslice	*Not Supported*
date	date
datetime	datetime
decimal	decimal(18,0)
decimal(p,s)	decimal(p,s)

IBM NETEZZA DATA TYPE	AZURE SYNAPSE SQL V2 DATA TYPE
double	float(53)
float	float
float(p)	float(p)
float4	float(53)
float8	float(53)
integer (alias int, int4)	int
interval (alias timespan)	*Not Supported*
national bpchar(n)	nvarchar(n)
national character(n) (alias nchar(n))	nchar(n)
number(p,s)	decimal(p,s)
numeric	decimal(18,0)
numeric(p,s)	numeric (p,s)
national character varying(n) (alias nvarchar(n))	nvarchar(n)
real	real
rowid	*Not Supported*
smallint (alias int2)	smallint
st_geometry(n)	*Not Supported*
time	time
time with time zone (alias timetz)	datetimeoffset
timestamp(p)	datetime2(p)
transactionid	*Not Supported*

Note: Some third-party vendors offer tools and services that can automate migration tasks, including data type mapping. If a third-party ETL tool like Informatica or Talend is already used in the Netezza environment, you can use the tool to implement any data transformations that are required.

- **SQL Data Manipulation Language (DML) syntax**: You should be aware of a few differences in SQL DML syntax between Netezza SQL and Azure Synapse.

  Here are some key functions and how they are different:

  - `STRPOS`: In Netezza, the `STRPOS` function returns the position of a substring within a string. The

equivalent in Azure Synapse is the `CHARINDEX` function, and the order of the arguments is reversed.

In Netezza:

```
SELECT STRPOS('abcdef', 'def') ...
```

Is replaced with the following code in Azure Synapse:

```
SELECT CHARINDEX('def', 'abcdef') ...
```

- `AGE`: Netezza supports the `AGE` operator to give the interval between two temporal values (for example, timestamps and dates). For example:

  ```
  SELECT AGE ('23-03-1956', '01-01-2019') FROM ...
  ```

  You can achieve the same result in Azure Synapse by using `DATEDIFF` (note the date representation sequence):

  ```
  SELECT DATEDIFF(day, '1956-03-23', '2019-01-01') FROM ...
  ```

- `NOW()`: Netezza uses `NOW()` to represent `CURRENT_TIMESTAMP` in Azure Synapse.

Functions, stored procedures, and sequences

When you migrate a data warehouse from a mature legacy environment like Netezza, you often need to migrate elements other than simple tables and views to the new target environment. Examples of non-table elements in Netezza that you might need to migrate to Azure Synapse are functions, stored procedures, and sequences. During the preparation phase of the migration, you should create an inventory of objects to migrate. In the project plan, define the method of handling all objects and allocate the appropriate resources for their migration.

You might find services in the Azure environment that replace the functionality implemented as functions or stored procedures in the Netezza environment. Usually, it's more efficient to use the built-in Azure capabilities instead of recoding the Netezza functions.

Also, third-party vendors offer tools and services that can automate the migration of functions, stored procedures, and sequences from Netezza. Examples include Qlik (formerly Attunity) and WhereScape.

Here's some additional information about migrating functions, stored procedures, and sequences:

- **Functions:** Like most database products, Netezza supports system functions and user-defined functions in a SQL implementation. When common system functions are migrated to another database platform like Azure Synapse, they generally are available in the new environment and can be migrated without change. If system functions have slightly different syntax in the new environment, you usually can automate the required changes.

  You might need to recode arbitrary user-defined functions and system functions that have no equivalent in the new environment. Use the languages that are available in the new environment. Netezza user-defined functions are coded by using nzLua or C++. Azure Synapse uses the popular Transact-SQL language to implement user-defined functions.

- **Stored procedures:** In most modern database products, you can store procedures n the database. A stored procedure typically contains SQL statements and some procedural logic. It might also return data or a status.

  Netezza provides the NZPLSQL language, based on PL/pgSQL, for stored procedures. Azure Synapse supports stored procedures by using T-SQL. If you migrate stored procedures to Azure Synapse, you must recode them by using T-SQL.

- **Sequences:** In Netezza, a sequence is a named database object that's created via a `CREATE SEQUENCE` statement. Objects can provide the unique value via the `NEXT()` method. You can use values to generate unique numbers as surrogate key values for primary key values.

  Azure Synapse doesn't support `CREATE SEQUENCE`. In Azure Synapse, sequences are handled by using identity columns or SQL code to create the next sequence number in a series.

Metadata and data extraction

Consider the following information when you plan how to extract metadata and data from the Netezza environment:

- **Data Definition Language (DDL) generation:** It's possible to edit existing Netezza `CREATE TABLE` and `CREATE VIEW` scripts to create the equivalent definitions, with modified data types if necessary, as described earlier. This task usually involves removing or modifying any clauses that are specific to Netezza, like `ORGANIZE ON`.

  In Netezza, the information that specifies the current table and view definitions is maintained in system catalog tables. System catalog tables are the best source of the information because the tables likely are up to date and complete. User-maintained documentation might not be in sync with current table definitions.

You can access system catalog tables in Netezza by using a utility like nz_ddl_table. You can use the tables to generate `CREATE TABLE` DDL statements, which you can then edit for the equivalent tables in Azure Synapse. Third-party migration and ETL tools also use the catalog information to achieve the same results.

- **Data extraction:** You can extract raw data to migrate from an existing Netezza table into a flat, delimited file by using standard Netezza utilities like nzsql and nzunload, and by using external tables. Compress the files by using Gzip, and then use AzCopy or an Azure data transport service like Azure Data Box to upload the files to Azure Blob Storage.

  During a migration exercise, it's important to extract data as efficiently as possible. The recommended approach for Netezza is to use external tables, which is also the fastest method. You can complete multiple extracts in parallel to maximize the throughput for data extraction.

Here's a simple example of an external table extract:

```
CREATE EXTERNAL TABLE '/tmp/export_tab1.CSV' USING (DELIM ',') AS SELECT * from <TABLE-NAME>;
```

If you have sufficient network bandwidth, you can extract data directly from an on-premises Netezza system into Azure Synapse tables or into Azure data storage by using Data Factory processes or third-party data migration or ETL products.

Recommended data formats for extracted data are delimited text files (also called *comma-separated values*), optimized row columnar files, or Parquet files.

For more detailed information about the process of migrating data and ETL from a Netezza environment, see the Netezza documentation about data migration ETL and load.

Performance-tuning recommendations

When you move to Azure Synapse from a Netezza environment, many of the performance-tuning concepts you use will be familiar.

For example, these concepts are the same for both environments:

- Data distribution colocates data to be joined onto the same processing node.

- Using the smallest data type for a specific column saves storage space and accelerates query processing.
- Ensuring that data types of columns to be joined are identical optimizes join processing by reducing the need to transform data for matching.
- Ensuring that statistics are up to date helps the optimizer produce the best execution plan.

There are some differences between platforms when it comes to optimization. In the following list of performance-tuning recommendations, lower-level implementation differences between Netezza and Azure Synapse, and alternatives for your migration, are highlighted:

- **Data distribution options:** In both Netezza and Azure Synapse, you can use a `CREATE TABLE` statement to specify a distribution definition. Use `DISTRIBUTE ON` for Netezza and `DISTRIBUTION =` for Azure Synapse.

  Azure Synapse provides an additional way to achieve local joins for small table/large table joins, often called a *dimension table/fact table join* in a star schema model. The approach is to replicate the smaller dimension table across all nodes, thereby ensuring that any value of the join key for the larger table will have a matching dimension row that's locally available. The overhead of replicating the dimension table is relatively low if the tables are not large. In this case, using the hash distribution approach described earlier is preferable.

- **Data indexing:** Azure Synapse provides various user-definable indexing options, but the options are different in operation and usage than system-managed zone maps in Netezza. To learn about the indexing options in Azure Synapse, see Index tables in an Azure Synapse SQL pool.

  Existing system-managed zone maps in the source Netezza environment can provide a useful indication of how data is used and provide an indication of candidate columns for indexing in the Azure Synapse environment.

- **Data partitioning:** In an enterprise data warehouse, fact tables might contain many billions of rows of data. Partitioning is a way to optimize maintenance and querying in these tables. Splitting the tables into separate parts reduces the amount of data processed at one time. Partitioning for a table is defined in the `CREATE TABLE` statement.

  Only one field per table can be used for partitioning. The field that's used for partitioning frequently is a date field because many queries are filtered by date or by a date range. You can change the partitioning of a table after initial load. To change a table's partitioning, re-create the table with a new distribution that uses the `CREATE TABLE AS SELECT` statement. For a detailed description of partitioning in Azure Synapse, see Partition tables in an Azure Synapse SQL pool.

- **PolyBase for data loading:** PolyBase is the most efficient method to use to load large amounts of data into a warehouse. You can use PolyBase to load data in parallel streams.

- **Resource classes for workload management:** Azure Synapse uses resource classes to manage workloads. In general, large resource classes provide better individual query performance. Smaller resource classes give you higher levels of concurrency. You can use dynamic management views to monitor utilization to help ensure that the appropriate resources are used efficiently.

Next steps

For more information about implementing a Netezza migration, talk with your Microsoft account representative about on-premises migration offers.

Azure Synapse Analytics solutions and migration for an Oracle data warehouse

3/6/2021 • 2 minutes to read • Edit Online

An Oracle data warehouse schema is different from Azure Synapse Analytics in several ways. The differences include databases, data types, and a range of Oracle Database object types that aren't supported in Azure Synapse.

Like other database management systems, when you migrate an Oracle data warehouse to Azure Synapse, you'll find that Oracle has multiple, separate databases and Azure Synapse has only one database. You might need to use a new naming convention, such as concatenating Oracle schema and table names, to move tables and views in your Oracle data warehouse staging database, production database, and data mart databases to Azure Synapse.

Several Oracle Database objects aren't supported in Azure Synapse. Database objects that aren't supported in Azure Synapse include Oracle bit-mapped indexes, function-based indexes, domain indexes, Oracle clustered tables, row-level triggers, user-defined data types, and PL/SQL stored procedures. You can identify these objects by querying various Oracle system catalog tables and views. In some cases, you can use workarounds. For example, you can use partitioning or other index types in Azure Synapse to work around the unsupported index types in Oracle. You might be able to use materialized views instead of Oracle clustered tables, and migration tools like SQL Server Migration Assistant (SSMA) for Oracle can translate at least some PL/SQL.

When you migrate an Oracle data warehouse schema, you also must take into account data type differences on columns. To find the columns in your Oracle data warehouse and data mart schemas that have data types that don't map to data types in Azure Synapse, query the Oracle catalog. You can use workarounds for several of these instances.

To maintain or improve performance of your schema after migration, consider performance mechanisms, like Oracle indexing, that you currently have in place. For example, bit-mapped indexes that Oracle queries frequently use might indicate that creating a nonclustered index in the migrated schema on Azure Synapse would be advantageous.

A good practice in Azure Synapse includes using data distribution to colocate data to be joined onto the same processing node. Another good practice in Azure Synapse is ensuring that data types of columns to be joined are identical. Using identical joined columns optimizes join processing by reducing the need to transform data for matching. In Azure Synapse, often it isn't necessary to migrate every Oracle index because other features provide high performance. You can instead use parallel query processing, in-memory data, and result set caching and data distribution options that reduce I/O.

SSMA for Oracle can help you migrate an Oracle data warehouse or data mart to Azure Synapse. SSMA is designed to automate the process of migrating tables, views, and data from an existing Oracle environment. Among other features, SSMA recommends index types and data distributions for target Azure Synapse tables, and it applies data type mappings during migration. Although SSMA isn't the most efficient approach for very high volumes of data, it's useful for smaller tables.

Mainframe migration overview

3/6/2021 • 4 minutes to read • Edit Online

Many companies and organizations benefit from moving some or all their mainframe workloads, applications, and databases to the cloud. Azure provides mainframe-like features at cloud scale without many of the drawbacks associated with mainframes.

The term *mainframe* generally refers to a large computer system, but the vast majority currently of mainframes deployed are IBM System z servers or IBM plug-compatible systems running MVS, DOS, VSE, OS/390, or z/OS. Mainframe systems continue to be used in many industries to run vital information systems, and they have a place in highly specific scenarios, such as large, high-volume, transaction-intensive IT environments.

Migrating to the cloud enables companies to modernize their infrastructure. With cloud services you can make mainframe applications, and the value that they provide, available as a workload whenever your organization needs it. Many workloads can be transferred to Azure with only minor code changes, such as updating the names of databases. You can migrate more complex workloads using a phased approach.

Most Fortune 500 companies are already running Azure for their critical workloads. Azure's significant bottom-line incentives motivate many migration projects. Companies typically move development and test workloads to Azure first, followed by DevOps, email, and disaster recovery.

Intended audience

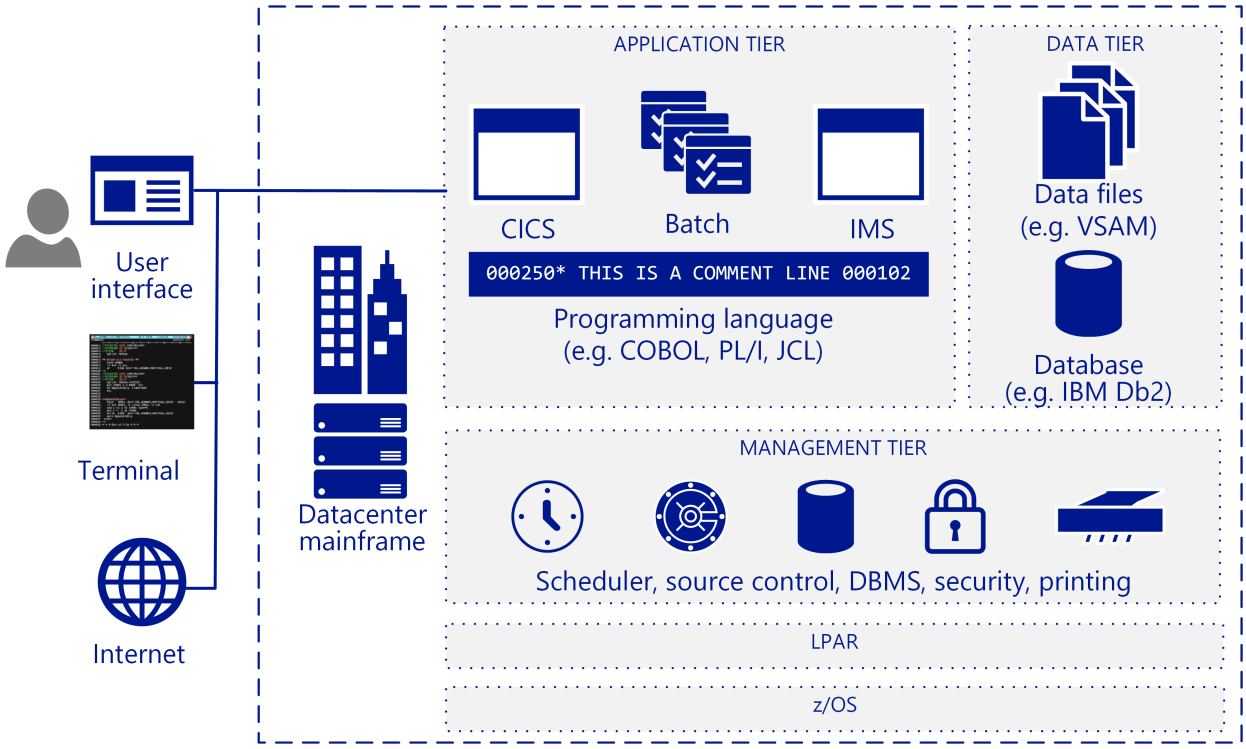
If you're considering a migration or the addition of cloud services as an option for your IT environment, this guide is for you.

This guidance helps IT organizations start the migration conversation. You may be more familiar with Azure and cloud-based infrastructures than you are with mainframes, so this guide starts with an overview of how mainframes work, and continues with various strategies for determining what and how to migrate.

Mainframe architecture

In the late 1950s, mainframes were designed as scale-up servers to run high-volume online transactions and batch processing. Because of this, mainframes have software for online transaction forms (sometimes called green screens) and high-performance I/O systems for processing batch runs.

Mainframes are known for high reliability and availability as well as their ability to run huge online transactions and batch jobs. A transaction results from a piece of processing initiated by a single request, typically from a user at a terminal. Transactions can also come from multiple other sources, including web pages, remote workstations, and applications from other information systems. A transaction can also be triggered automatically at a predefined time as the following figure shows.



A typical IBM mainframe architecture includes these common components:

- **Front-end systems:** Users can initiate transactions from terminals, web pages, or remote workstations. Mainframe applications often have custom user interfaces that can be preserved after migration to Azure. Terminal emulators (also called green-screen terminals) are still used to access mainframe applications.
- **Application tier:** Mainframes typically include a Customer Information Control System (CICS), a leading transaction management suite for the IBM z/OS mainframe that is often used with IBM Information Management System (IMS), a message-based transaction manager. Batch systems handle high-throughput data updates for large volumes of account records.
- **Code:** Programming languages used by mainframes include COBOL, Fortran, PL/I, and Natural. Job control language (JCL) is used to work with z/OS.
- **Database tier:** A common relational database management system (DBMS) for z/OS is IBM DB2. It manages data structures called *dbspaces* that contain one or more tables and are assigned to storage pools of physical data sets called *dbextents*. Two important database components are the directory that identifies data locations in the storage pools, and the log that contains a record of operations performed on the database. Various flat-file data formats are supported. DB2 for z/OS typically uses Virtual Storage Access Method (VSAM) datasets to store the data.
- **Management tier:** IBM mainframes include scheduling software such as TWS-OPC, tools for print and output management such as CA-SAR and SPOOL, and a source control system for code. Secure access control for z/OS is handled by resource access control facility (RACF). A database manager provides access to data in the database and runs in its own partition in a z/OS environment.
- **LPAR:** Logical partitions, or LPARs, are used to divide compute resources. A physical mainframe is partitioned into multiple LPARs.
- **z/OS:** A 64-bit operating system that is most commonly used for IBM mainframes.

IBM systems use a transaction monitor such as CICS to track and manage all aspects of a business transaction. CICS manages the sharing of resources, the integrity of data, and prioritization of execution. CICS authorizes users, allocates resources, and passes database requests by the application to a database manager, such as IBM DB2.

For more precise tuning, CICS is commonly used with IMS/TM, formerly IMS/Data Communications (IMS/DC).

IMS was designed to reduce data redundancy by maintaining a single copy of the data. It complements CICS as a transaction monitor by maintaining state throughout the process and recording business functions in a data store.

Mainframe operations

The following are typical mainframe operations:

- **Online:** Workloads include transaction processing, database management, and connections. They are often implemented using IBM DB2, CICS, and z/OS connectors.
- **Batch:** Jobs run without user interaction, typically on a regular schedule such as every weekday morning. Batch jobs can be run on systems based on Windows or Linux by using a JCL emulator such as Micro Focus Server Enterprise Edition or BMC Control-M software.
- **Job control language (JCL):** Specify resources needed to process batch jobs. JCL conveys this information to z/OS through a set of job control statements. Basic JCL contains six types of statements: JOB, ASSGN, DLBL, EXTENT, LIBDEF, and EXEC. A job can contain several EXEC statements (steps), and each step could have several LIBDEF, ASSGN, DLBL, and EXTENT statements.
- **Initial program load (IPL):** Refers to loading a copy of the operating system from disk into a processor's real storage and running it. IPLs are used to recover from downtime. An IPL is like booting the operating system on Windows or Linux VMs.

Next steps

Myths and facts

Mainframe myths and facts

3/6/2021 • 2 minutes to read • Edit Online

Mainframes figure prominently in the history of computing and remain viable for highly specific workloads. Most agree that mainframes are a proven platform with long-established operating procedures that make them reliable, robust environments. Software runs based on usage, measured in million instructions per second (MIPS), and extensive usage reports are available for chargebacks.

The reliability, availability, and processing power of mainframes have taken on almost mythical proportions. To evaluate the mainframe workloads that are most suitable for Azure, you first want to distinguish the myths from the reality.

Myth: Mainframes never go down and have a minimum of five 9s of availability

Mainframe hardware and operating systems are viewed as reliable and stable. But the reality is that downtime must be scheduled for maintenance and reboots, referred to as initial program loads (IPLs). When these tasks are considered, a mainframe solution often has closer to two or three 9s of availability, which is equivalent to that of high-end, Intel-based servers.

Mainframes also remain as vulnerable to disasters as any other servers do, and require uninterruptible power supply (UPS) systems to handle these types of failures.

Myth: Mainframes have limitless scalability

A mainframe's scalability depends on the capacity of its system software, such as the Customer Information Control System (CICS), and the capacity of new instances of mainframe engines and storage. Some large companies that use mainframes have customized their CICS for performance, and have otherwise outgrown the capability of the largest available mainframes.

Myth: Intel-based servers are not as powerful as mainframes

The new core-dense, Intel-based systems have as much compute capacity as mainframes.

Myth: The cloud can't accommodate mission-critical applications for large companies such as financial institutions

Although there may be some isolated instances where cloud solutions fall short, it's usually because the application algorithms cannot be distributed. These few examples are the exceptions, not the rule.

Summary

By comparison, Azure offers an alternative platform that is capable of delivering equivalent mainframe functionality and features, and at a much lower cost. In addition, the total cost of ownership (TCO) of the cloud's subscription-based, usage-driven cost model is far less expensive than mainframe computers.

Next steps

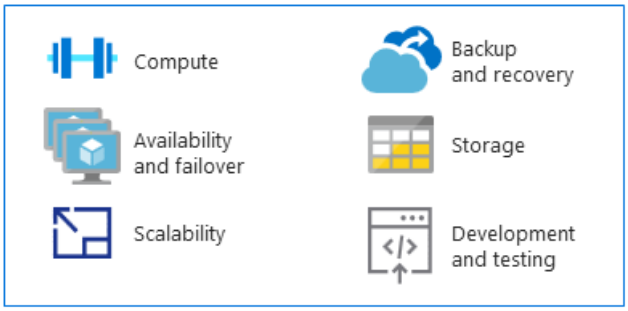
Make the switch from mainframes to Azure

Make the switch from mainframes to Azure

3/6/2021 • 4 minutes to read • Edit Online

As an alternative platform for running traditional mainframe applications, Azure offers hyperscale compute and storage in a high availability environment. You get the value and agility of a modern, cloud-based platform without the costs associated with a mainframe environment.

This section provides technical guidance for making the switch from a mainframe platform to Azure.



MIPS vs. vCPUs

There is no universal mapping formula that exists for determining the number of virtual central processing units (vCPUs) needed to run mainframe workloads. However, the metric of a million instructions per second (MIPS) is often mapped to vCPUs on Azure. MIPS measures the overall compute power of a mainframe by providing a constant value of the number of cycles per second for a given machine.

A small organization might require less than 500 MIPS, while a large organization typically uses more than 5,000 MIPS. At $1,000 per single MIPS, a large organization spends approximately $5 million annually to deploy a 5,000-MIPS infrastructure. The annual cost estimate for a typical Azure deployment of this scale is approximately one-tenth the cost of a MIPS infrastructure. For details, see Table 4 in the Demystifying Mainframe-to-Azure Migration white paper.

An accurate calculation of MIPS to vCPUs with Azure depends on the type of vCPU and the exact workload you are running. However, benchmark studies provide a good basis for estimating the number and type of vCPUs you will need. A recent HPE zRef benchmark provides the following estimates:

- 288 MIPS per Intel-based core running on HPE ProLiant servers for online (CICS) jobs.
- 170 MIPS per Intel core for COBOL batch jobs.

This guide estimates 200 MIPS per vCPU for online processing and 100 MIPS per vCPU for batch processing.

> **NOTE**
> These estimates are subject to change as new virtual machine (VM) series become available in Azure.

High availability and failover

Mainframe systems often offer five 9s availability (99.999 percent) when mainframe coupling and Parallel Sysplex are used. Yet system operators still need to schedule downtime for maintenance and initial program loads (IPLs). The actual availability approaches two or three 9s, comparable to high end, Intel-based servers.

By comparison, Azure offers commitment-based service-level agreements (SLAs), where multiple 9s availability

is the default, optimized with local or geo-based replication of services.

Azure provides additional availability by replicating data from multiple storage devices, either locally or in other geographic regions. In the event of an Azure-based failure, compute resources can access the replicated data on either the local or regional level.

When you use Azure platform as a service (PaaS) resources, such as Azure SQL Database and Azure Cosmos DB, Azure can automatically handle failovers. When you use Azure infrastructure as a service (IaaS), failover relies on specific system functionality, such as SQL Server Always On features, failover clustering instances, and availability groups.

Scalability

Mainframes typically scale up, while cloud environments scale out. Mainframes can scale out with the use of a coupling facility (CF), but the high cost of hardware and storage makes mainframes expensive to scale out.

A CF also offers tightly coupled compute, whereas the scale-out features of Azure are loosely coupled. The cloud can scale up or down to match exact user specifications, with compute power, storage, and services scaling on demand under a usage-based billing model.

Backup and recovery

Mainframe customers typically maintain disaster recovery sites or make use or an independent mainframe provider for disaster contingencies. Synchronization with a disaster recovery site is usually done through offline copies of data. Both options incur high costs.

Automated geo-redundancy is also available through the mainframe coupling facility. This approach is expensive and is typically reserved for mission-critical systems. In contrast, Azure has easy-to-implement and cost-effective options for backup, recovery, and redundancy at local or regional levels, or via geo-redundancy.

Storage

Part of understanding how mainframes work involves decoding various overlapping terms. For example, central storage, real memory, real storage, and main storage all generally refer to storage attached directly to the mainframe processor.

Mainframe hardware includes processors and many other devices, such as direct-access storage devices (DASDs), magnetic tape drives, and several types of user consoles. Tapes and DASDs are used for system functions and by user programs.

Types of physical storage for mainframes include:

- **Central storage:** Located directly on the mainframe processor, this is also known as processor or real storage.
- **Auxiliary storage:** Located separately from the mainframe, this type includes storage on DASDs and is also known as paging storage.

The cloud offers a range of flexible, scalable options, and you will pay only for those options that you need. Azure Storage offers a massively scalable object store for data objects, a file system service for the cloud, a reliable messaging store, and a NoSQL store. For VMs, managed and unmanaged disks provide persistent, secure disk storage.

Mainframe development and testing

A major driver in mainframe migration projects is the changing face of application development. Organizations want their development environment to be more agile and responsive to business needs.

Mainframes typically have separate logical partitions (LPARs) for development and testing, such as QA and staging LPARs. Mainframe development solutions include compilers (COBOL, PL/I, Assembler) and editors. The most common is the Interactive System Productivity Facility (ISPF) for the z/OS operating system that runs on IBM mainframes. Others include ROSCOE Programming Facility (RPF) and Computer Associates tools, such as CA Librarian and CA-Panvalet.

Emulation environments and compilers are available on x86 platforms, so development and testing can typically be among the first workloads to migrate from a mainframe to Azure. The availability and widespread use of DevOps tools in Azure is accelerating the migration of development and testing environments.

When solutions are developed and tested on Azure and are ready for deployment to the mainframe, you will need to copy the code to the mainframe and compile it there.

Next steps

Mainframe application migration

Mainframe application migration

6/1/2021 • 10 minutes to read • Edit Online

When migrating applications from mainframe environments to Azure, most teams follow a pragmatic approach: reuse wherever and whenever possible, and then start a phased deployment where applications are rewritten or replaced.

Application migration typically involves one or more of the following strategies:

- **Rehost:** You can move existing code, programs, and applications from the mainframe, and then recompile the code to run in a mainframe emulator hosted in a cloud instance. This approach typically starts with moving applications to a cloud-based emulator, and then migrating the database to a cloud-based database. Some engineering and refactoring are required along with data and file conversions.

  Alternatively, you can rehost using a traditional hosting provider. One of the principal benefits of the cloud is outsourcing infrastructure management. You can find a datacenter provider that will host your mainframe workloads for you. This model may buy time, reduce vendor lock in, and produce interim cost savings.

- **Retire:** All applications that are no longer needed should be retired before migration.

- **Rebuild:** Some organizations choose to completely rewrite programs using modern techniques. Given the added cost and complexity of this approach, it's not as common as a lift-and-shift approach. Often after this type of migration, it makes sense to begin replacing modules and code using code transformation engines.

- **Replace:** This approach replaces mainframe functionality with equivalent features in the cloud. Software as a service (SaaS) is one option, which is using a solution created specifically for an enterprise concern, such as finance, human resources, manufacturing, or enterprise resource planning. In addition, many industry-specific applications are now available to solve problems that custom mainframe solutions used to previously solve.

You should consider starting by planning those workloads that you want to initially migrate, and then determine those requirements for moving associated applications, legacy code bases, and databases.

Mainframe emulation in Azure

Azure services can emulate traditional mainframe environments, enabling you to reuse existing mainframe code and applications. Common server components that you can emulate include online transaction processing (OLTP), batch, and data ingestion systems.

OLTP systems

Many mainframes have OLTP systems that process thousands or millions of updates for huge numbers of users. These applications often use transaction processing and screen-form handling software, such as Customer Information Control System (CICS), Information Management System (IMS), and Terminal Interface Processor (TIP).

When moving OLTP applications to Azure, emulators for mainframe transaction processing (TP) monitors are available to run as infrastructure as a service (IaaS) using virtual machines (VMs) on Azure. The screen handling and form functionality can also be implemented by web servers. This approach can be combined with database APIs, such as ActiveX Data Objects (ADO), Open Database Connectivity (ODBC), and Java Database Connectivity (JDBC) for data access and transactions.

Time-constrained batch updates

Many mainframe systems perform monthly or annual updates of millions of account records, such as those used in banking, insurance, and government. Mainframes handle these types of workloads by offering high-throughput data handling systems. Mainframes batch jobs are typically serial in nature and depend on the input/output operations per second (IOPS) provided by the mainframe backbone for performance.

Cloud-based batch environments use parallel compute and high-speed networks for performance. If you need to optimize batch performance, Azure provides various compute, storage, and networking options.

Data ingestion systems

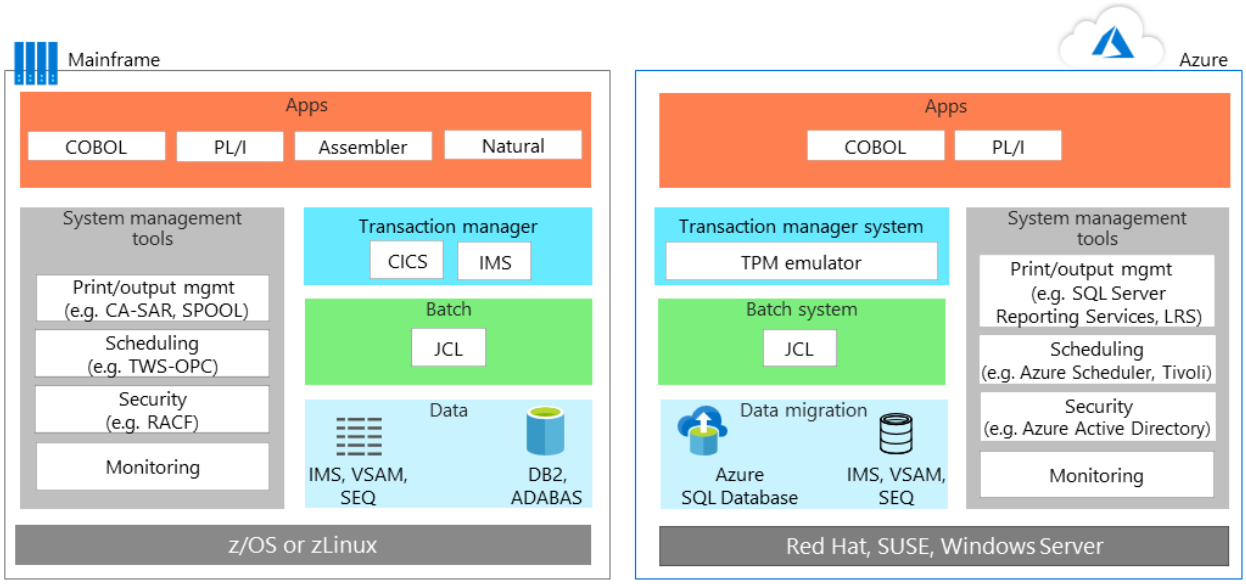
Mainframes ingest large batches of data from retail, financial services, manufacturing, and other solutions for processing. With Azure, you can use simple command-line utilities such as AzCopy for copying data to and from storage location. You can also use the Azure Data Factory service, enabling you to ingest data from disparate data stores to create and schedule data-driven workflows.

In addition to emulation environments, Azure provides platform as a service (PaaS) and analytics services that can enhance existing mainframe environments.

Migrate OLTP workloads to Azure

The lift-and-shift approach is the no-code option for quickly migrating existing applications to Azure. Each application is migrated as is, which provides the benefits of the cloud without the risks or costs of making code changes. Using an emulator for mainframe transaction processing (TP) monitors on Azure supports this approach.

TP monitors are available from various vendors and run on virtual machines, an infrastructure as a service (IaaS) option on Azure. The following before and after diagrams show a migration of an online application backed by IBM DB2, a relational database management system (DBMS), on an IBM z/OS mainframe. DB2 for z/OS uses Virtual Storage Access Method (VSAM) files to store the data and Indexed Sequential Access Method (ISAM) for flat files. This architecture also uses CICS for transaction monitoring.



On Azure, emulation environments are used to run the TP manager and the batch jobs that use JCL. In the data tier, DB2 is replaced by Azure SQL Database, although Microsoft SQL Server, DB2 LUW, or Oracle Database can also be used. An emulator supports IMS, VSAM, and SEQ. The mainframe's system management tools are replaced by Azure services, and software from other vendors, that run in VMs.

The screen handling and form entry functionality is commonly implemented using web servers, which can be combined with database APIs, such as ADO, ODBC, and JDBC for data access and transactions. The exact line-up of Azure IaaS components to use depends on the operating system you prefer. For example:

- **Windows-based VMs:** Internet Information Server (IIS) along with ASP.NET for the screen handling and business logic. Use ADO.NET for data access and transactions.
- **Linux-based VMs:** The Java-based application servers that are available, such as Apache Tomcat for screen handling and Java-based business functionality. Use JDBC for data access and transactions.

Migrate batch workloads to Azure

Batch operations in Azure differ from the typical batch environment on mainframes. Mainframe batch jobs are typically serial in nature and depend on the IOPS provided by the mainframe backbone for performance. Cloud-based batch environments use parallel computing and high-speed networks for performance.

To optimize batch performance using Azure, consider the compute, storage, networking, and monitoring options as follows.

Compute

Use:

- VMs with the highest clock speed. Mainframe applications are often single-threaded and mainframe CPUs have a very high clock speed.
- VMs with large memory capacity to allow caching of data and application work areas.
- VMs with higher density vCPUs to take advantage of multithreaded processing if the application supports multiple threads.
- Parallel processing, as Azure easily scales out for parallel processing, delivering more compute power for a batch run.

Storage

Use:

- Azure premium SSDs or Azure ultra disks for maximum available IOPS.
- Striping with multiple disks for more IOPS per storage size.
- Partitioning for storage to spread I/O over multiple Azure Storage devices.

Networking

- Use Azure accelerated networking to minimize latency.

Monitoring

- Use monitoring tools, Azure Monitor, Application Insights, and Azure logs enable administrators to monitor any over performance of batch runs and help eliminate bottlenecks.

Migrate development environments

The cloud's distributed architectures rely on a different set of development tools that provide the advantage of modern practices and programming languages. To ease this transition, you can use a development environment with other tools that are designed to emulate IBM z/OS environments. The following list shows options from Microsoft and other vendors:

COMPONENT	AZURE OPTIONS
z/OS	Windows, Linux, or Unix

COMPONENT	AZURE OPTIONS
CICS	Azure services offered by Micro Focus, Oracle, GT Software (Fujitsu), TmaxSoft, Raincode, and NTT Data, or rewrite using Kubernetes
IMS	Azure services offered by Micro Focus and Oracle
Assembler	Azure services from Raincode and TmaxSoft; or COBOL, C, or Java, or map to operating system functions
JCL	JCL, PowerShell, or other scripting tools
COBOL	COBOL, C, or Java
Natural	Natural, COBOL, C, or Java
Fortran and PL/I	Fortran, PL/I, COBOL, C, or Java
REXX and PL/I	REXX, PowerShell, or other scripting tools

Migrate databases and data

Application migration usually involves rehosting the data tier. You can migrate SQL Server, open-source, and other relational databases to fully managed solutions on Azure, such as Azure SQL Managed Instance, Azure Database for PostgreSQL, and Azure Database for MySQL with Azure Database Migration Service.

For example, you can migrate if the mainframe data tier uses:

- IBM DB2 or an IMS database, use Azure SQL Database, SQL Server, DB2 LUW, or Oracle Database on Azure.
- VSAM and other flat files, use Indexed Sequential Access Method (ISAM) flat files for Azure SQL Database, SQL Server, DB2 LUW, or Oracle.
- Generation date groups (GDGs), migrate to files on Azure that use a naming convention and filename extensions that provide similar functionality to GDGs.

The IBM data tier includes several key components that you must also migrate. For example, when you migrate a database, you also migrate a collection of data contained in pools, each containing dbextents, which are z/OS VSAM data sets. Your migration must include the directory that identifies data locations in the storage pools. Also, your migration plan must consider the database log, which contains a record of operations performed on the database. A database can have one, two (dual or alternate), or four (dual and alternate) logs.

Database migration also includes these components:

- **Database manager:** Provides access to data in the database. The database manager runs in its own partition in a z/OS environment.
- **Application requester:** Accepts requests from applications before passing them to an application server.
- **Online resource adapter:** Includes application requester components for use in CICS transactions.
- **Batch resource adapter:** Implements application requester components for z/OS batch applications.
- **Interactive SQL (ISQL):** Runs as a CICS application and interface enabling users to enter SQL statements or operator commands.
- **CICS application:** Runs under the control of CICS, using available resources and data sources in CICS.
- **Batch application:** Runs process logic without interactive communication with users to, for example,

produce bulk data updates or generate reports from a database.

Optimize scale and throughput for Azure

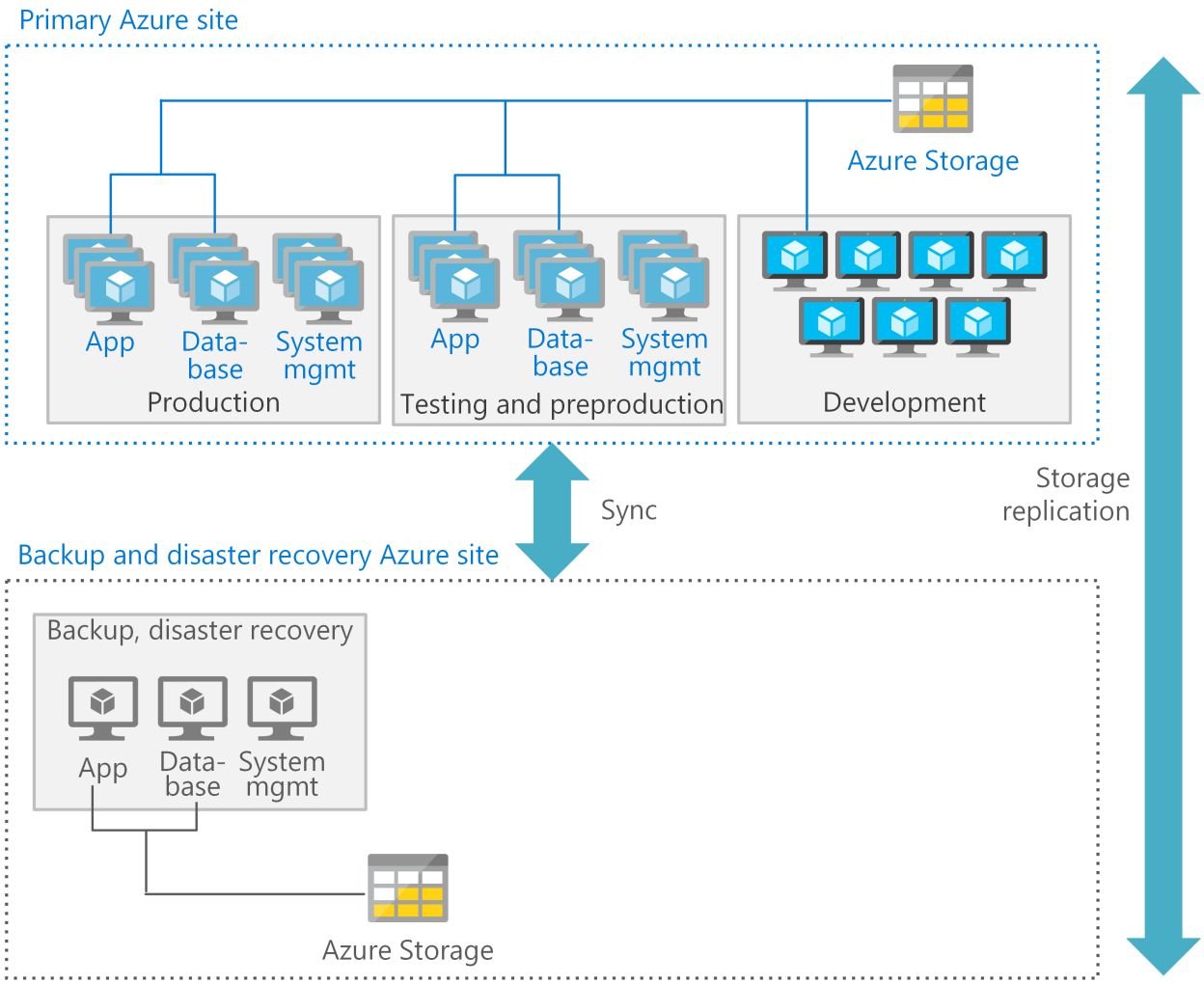
Generally speaking, mainframes scale up, while the cloud scales out. To optimize scale and throughput of mainframe-style applications running on Azure, it's important that you understand at how mainframes can separate and isolate applications. A z/OS mainframe uses a feature called logical partitions (LPARs) to isolate and manage the resources for a specific application on a single instance.

For example, a mainframe might use one LPAR for a CICS region with associated COBOL programs, and a separate LPAR for DB2. Additional LPARs are often used for the development, testing, and staging environments.

On Azure, it's more common to use separate VMs to serve this purpose. Azure architectures typically deploy VMs for the application tier, a separate set of VMs for the data tier, another set for development, and so on. Each tier of processing can be optimized using the most suitable type of VMs and features for that environment.

In addition, each tier can also provide appropriate disaster recovery services. For example, production and database VMs might require a hot or warm recovery, while the development and testing VMs support a cold recovery.

The following figure shows a possible Azure deployment using a primary and a secondary site. In the primary site, the production, staging, and testing VMs are deployed with high availability. The secondary site is for backup and disaster recovery.



Perform a staged migration to Azure

Moving solutions from a mainframe to Azure may involve a *staged* migration, whereby some applications are moved first, and others remain on the mainframe temporarily or permanently. This approach typically requires

systems that allow applications and databases to interoperate between the mainframe and Azure.

A common scenario is to move an application to Azure while keeping the data used by the application on the mainframe. Specific software is used to enable the applications on Azure to access data from the mainframe. Fortunately, a wide range of solutions provide integration between Azure and existing mainframe environments, support for hybrid scenarios, and migration over time. Microsoft partners, independent software vendors, and system integrators can help you on your journey.

One option is Microsoft Host Integration Server, a solution that provides the distributed relational database architecture (DRDA) required for applications in Azure to access data in DB2 that remains on the mainframe. Other options for mainframe-to-Azure integration include solutions from IBM, Attunity, Codit, other vendors, and open-source options.

Partner solutions

If you are considering a mainframe migration, the partner ecosystem is available to assist you.

Azure provides a proven, highly available, and scalable infrastructure for systems that currently run on mainframes. Some workloads can be migrated with relative ease. Other workloads that depend on legacy system software, such as CICS and IMS, can be rehosted using partner solutions and migrated to Azure over time. Regardless of the choice you make, Microsoft and our partners are available to assist you in optimizing for Azure while maintaining mainframe system software functionality.

Learn more

For more information, see the following resources:

- Get started with Azure
- Deploy IBM DB2 pureScale on Azure
- Host Integration Server documentation

Best practices to secure and manage workloads migrated to Azure

5/18/2021 • 29 minutes to read • Edit Online

As you plan and design for migration, in addition to thinking about the migration itself, you need to consider your security and management model in Azure after migration. This article describes planning and best practices for securing your Azure deployment after migrating. It also covers ongoing tasks to keep your deployment running at an optimal level.

> **IMPORTANT**
> The best practices and opinions described in this article are based on the Azure platform and service features available at the time of writing. Features and capabilities change over time.

Secure migrated workloads

After migration, the most critical task is to secure migrated workloads from internal and external threats. These best practices help you to do that:

- Learn how to work with the monitoring, assessments, and recommendations provided by Azure Security Center.
- Get best practices for encrypting your data in Azure.
- Protect your VMs from malware and malicious attacks.
- Keep sensitive information secure in migrated web apps.
- Verify who can access your Azure subscriptions and resources after migration.
- Review your Azure auditing and security logs on a regular basis.
- Understand and evaluate advanced security features that Azure offers.

These best practices are described in more detail in the sections that follow.

Best practice: Follow Azure Security Center recommendations

Azure tenant admins need to enable security features that protect workloads from attacks. Security Center provides unified security management. From Security Center, you can apply security policies across workloads, limit threat exposure, and detect and respond to attacks. Security Center analyzes resources and configurations across Azure tenants, and makes security recommendations, including:

- **Centralized policy management:** Ensure compliance with company or regulatory security requirements by centrally managing security policies across all your hybrid cloud workloads.
- **Continuous security assessment:** Monitor the security posture of machines, networks, storage and data services, and applications to discover potential security issues.
- **Actionable recommendations:** Remediate security vulnerabilities before they can be exploited by attackers, with prioritized and actionable security recommendations.
- **Prioritized alerts and incidents:** Focus on the most critical threats first, with prioritized security alerts and incidents.

In addition to assessments and recommendations, Security Center provides other security features that you can enable for specific resources.

- **Just-in-time (JIT) access.** Reduce your network attack surface with JIT, controlled access to management ports on Azure VMs.
  - Having VM RDP port 3389 open on the internet exposes VMs to continual activity from bad actors. Azure IP addresses are well-known, and hackers continually probe them for attacks on open 3389 ports.
  - JIT uses network security groups (NSGs) and incoming rules that limit the amount of time that a specific port is open.
  - With JIT access enabled, Security Center checks that a user has Azure role-based access control (Azure RBAC) write access permissions for a VM. In addition, you can specify rules for how users can connect to VMs. If permissions are OK, an access request is approved, and Security Center configures NSGs to allow inbound traffic to the selected ports for the amount of time you specify. NSGs return to their previous state when the time expires.
- **Adaptive application controls.** Keep software and malware off VMs by controlling which applications run on them, by using dynamic allow lists.
  - Adaptive application controls allow you to approve applications, and prevent rogue users or administrators from installing unapproved or vetting software applications on your VMs.
    - You can block or alert attempts to run malicious applications, avoid unwanted or malicious applications, and ensure compliance with your organization's application security policy.
- **File Integrity Monitoring.** Ensure the integrity of files running on VMs.
  - You don't need to install software to cause VM issues. Changing a system file can also cause VM failure or performance degradation. File Integrity Monitoring examines system files and registry settings for changes, and notifies you if something is updated.
  - Security Center recommends which files you should monitor.

Learn more:

- Learn more about Azure Security Center.
- Learn more about just-in-time VM access.
- Learn about applying adaptive application controls.
- Get started with File Integrity Monitoring.

Best practice: Encrypt data

Encryption is an important part of Azure security practices. Ensuring that encryption is enabled at all levels helps prevent unauthorized parties from gaining access to sensitive data, including data in transit and at rest.

Encryption for infrastructure as a service

- **Virtual machines:** For VMs, you can use Azure Disk Encryption to encrypt your Windows and Linux infrastructure as a service (IaaS) VM disks.
  - Azure Disk Encryption uses BitLocker for Windows, and dm-crypt for Linux, to provide volume encryption for the operating system and data disks.
  - You can use an encryption key created by Azure, or you can supply your own encryption keys, safeguarded in Azure Key Vault.
  - With Azure Disk Encryption, IaaS VM data is secured at rest (on the disk) and during VM boot.
    - Security Center alerts you if you have VMs that aren't encrypted.
- **Storage:** Protect at-rest data stored in Azure Storage.
  - Data stored in Azure Storage accounts can be encrypted by using Microsoft-generated AES keys that are FIPS 140-2 compliant, or you can use your own keys.
  - Azure Storage encryption is enabled for all new and existing storage accounts, and it can't be disabled.

Encryption for platform as a service

Unlike IaaS, in which you manage your own VMs and infrastructure, in a platform as a service (PaaS) model platform and infrastructure is managed by the provider. You can focus on core application logic and capabilities. With so many different types of PaaS services, each service is evaluated individually for security purposes. As an example, let's see how you might enable encryption for Azure SQL Database.

- **Always Encrypted:** Use the Always Encrypted wizard in SQL Server Management Studio to protect data at rest.
  - You create an Always Encrypted key to encrypt individual column data.
  - Always Encrypted keys can be stored as encrypted in database metadata, or stored in trusted key stores such as Azure Key Vault.
  - Most likely, you'll need to make application changes to use this feature.
- **Transparent data encryption (TDE):** Protect the Azure SQL Database with real-time encryption and decryption of the database, associated backups, and transaction log files at rest.
  - TDE allows encryption activities to take place without changes at the application layer.
  - TDE can use encryption keys provided by Microsoft, or you can bring your own key.

Learn more:

- Learn about Azure Disk Encryption for virtual machines and virtual machine scale sets.
- Enable Azure Disk Encryption for Windows VMs.
- Learn about Azure Storage encryption for data at rest.
- Read the Always Encrypted overview.
- Read about TDE for SQL Database and Azure Synapse.
- Learn about Azure SQL Database TDE with customer-managed key.

Best practice: Protect VMs with antimalware

In particular, older Azure-migrated VMs might not have the appropriate level of antimalware installed. Azure provides a free endpoint solution that helps protect VMs from viruses, spyware, and other malware.

- Microsoft Antimalware for Azure Cloud Services and Virtual Machines generates alerts when known malicious or unwanted software tries to install itself.
- It's a single agent solution that runs in the background without human intervention.
- In Security Center, you can identify VMs that don't have endpoint protection running and install Microsoft antimalware as needed.

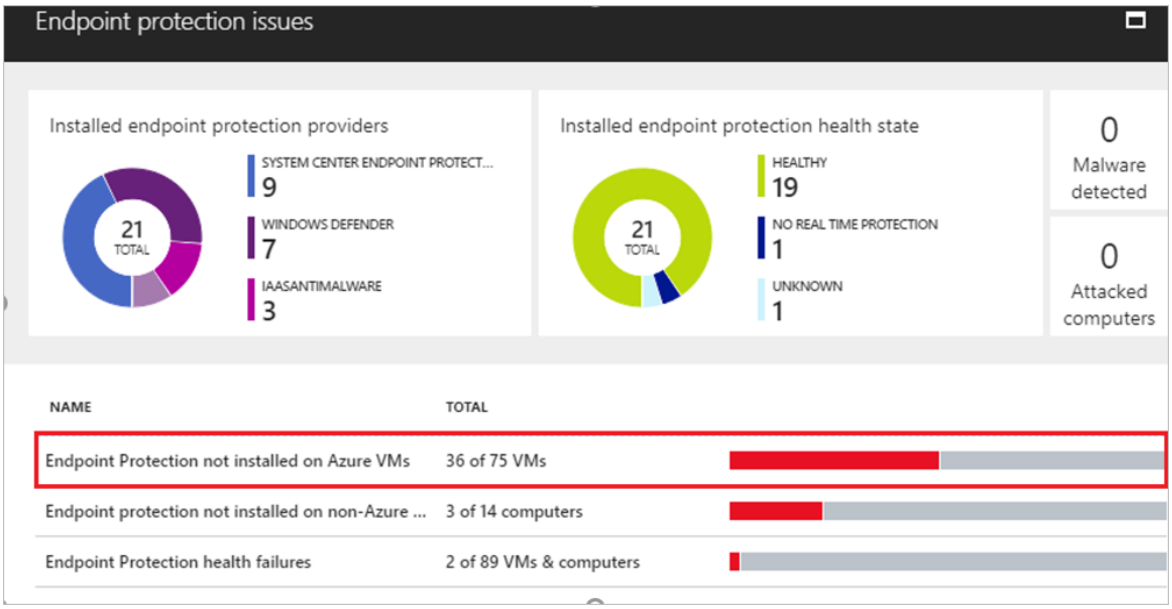


Figure 1: Antimalware for VMs.

Learn more:

- Learn about Microsoft Antimalware for Azure.

Best practice: Secure web apps

Migrated web apps face a couple of issues:

- Most legacy web applications tend to have sensitive information inside configuration files. Files containing such information can present security issues when applications are backed up, or when application code is checked into or out of source control.
- When you migrate web apps residing in a VM, you're likely moving that machine from an on-premises network and firewall-protected environment, to an environment facing the internet. Make sure that you set up a solution that does the same work as your on-premises protection resources.

Azure provides the following solutions:

- **Azure Key Vault:** Today, web app developers are taking steps to ensure that sensitive information isn't leaked from these files. One method to secure information is to extract it from files and put it into an Azure Key Vault.
  - You can use Key Vault to centralize storage of application secrets, and control their distribution. It avoids the need to store security information in application files.
  - Applications can securely access information in the vault by using URIs, without needing custom code.
  - Azure Key Vault allows you to lock down access via Azure security controls, and to seamlessly implement rolling keys. Microsoft doesn't see or extract your data.
- **App Service Environment for Power Apps:** If an application that you migrate needs extra protection, consider adding App Service Environment and Web Application Firewall to protect the application resources.
  - App Service Environment provides a fully isolated and dedicated environment for running applications, such as Windows and Linux web apps, Docker containers, mobile apps, and function apps.
  - It's useful for applications that are very high scale, require isolation and secure network access, or have high memory utilization.
- **Web Application Firewall:** This is a feature of Azure Application Gateway that provides centralized protection for web apps.
  - It protects web apps without requiring back-end code modifications.
  - It protects multiple web apps at the same time, behind Application Gateway.
  - You can monitor Web Application Firewall by using Azure Monitor. Web Application Firewall is integrated into Security Center.

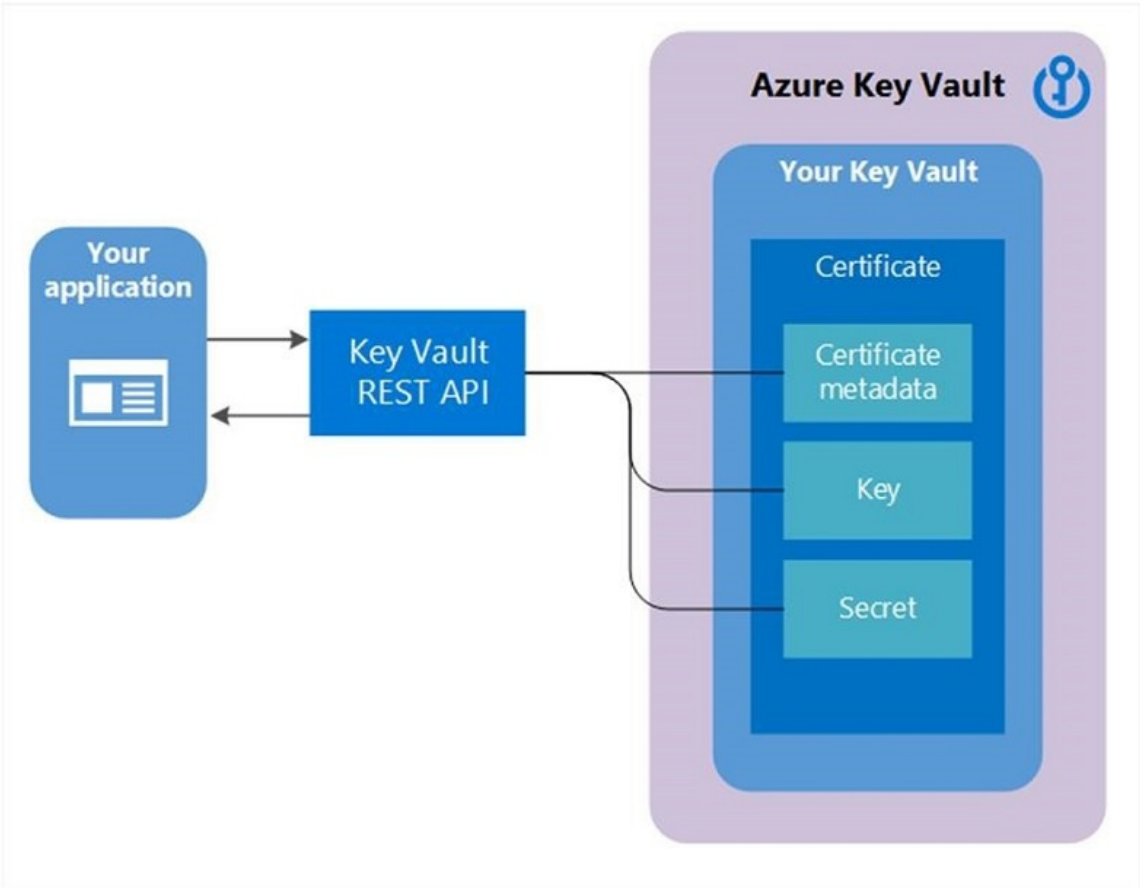


Figure 2: Azure Key Vault.

Learn more:

- Read the Azure Key Vault overview.
- Learn about Web Application Firewall.
- Read an introduction to App Service environments.
- Learn how to configure a web app to read secrets from Key Vault.

Best practice: Review subscriptions and resource permissions

As you migrate your workloads and run them in Azure, staff with workload access move around. Your security team should review access to your Azure tenant and resource groups on a regular basis. Azure has offerings for identity management and access control security, including Azure role-based access control (Azure RBAC) to authorize permissions to access Azure resources.

- Azure RBAC assigns access permissions for security principals. Security principals represent users, groups (a set of users), service principals (identity used by applications and services), and managed identities (an Azure Active Directory identity automatically managed by Azure).
- Azure RBAC can assign roles to security principals (such as Owner, Contributor, and Reader) and role definitions (a collection of permissions) that define the operations that the roles can perform.
- Azure RBAC can also set scopes that set the boundary for a role. The scope can be set at several levels, including a management group, subscription, resource group, or resource.
- Ensure that admins with Azure access can access only resources that you want to allow. If the predefined roles in Azure aren't granular enough, you can create custom roles to separate and limit access permissions.

Ensure that admins with Azure access can access only resources that you want to allow. If the predefined roles in Azure aren't granular enough, you can create custom roles to separate and limit access permissions.

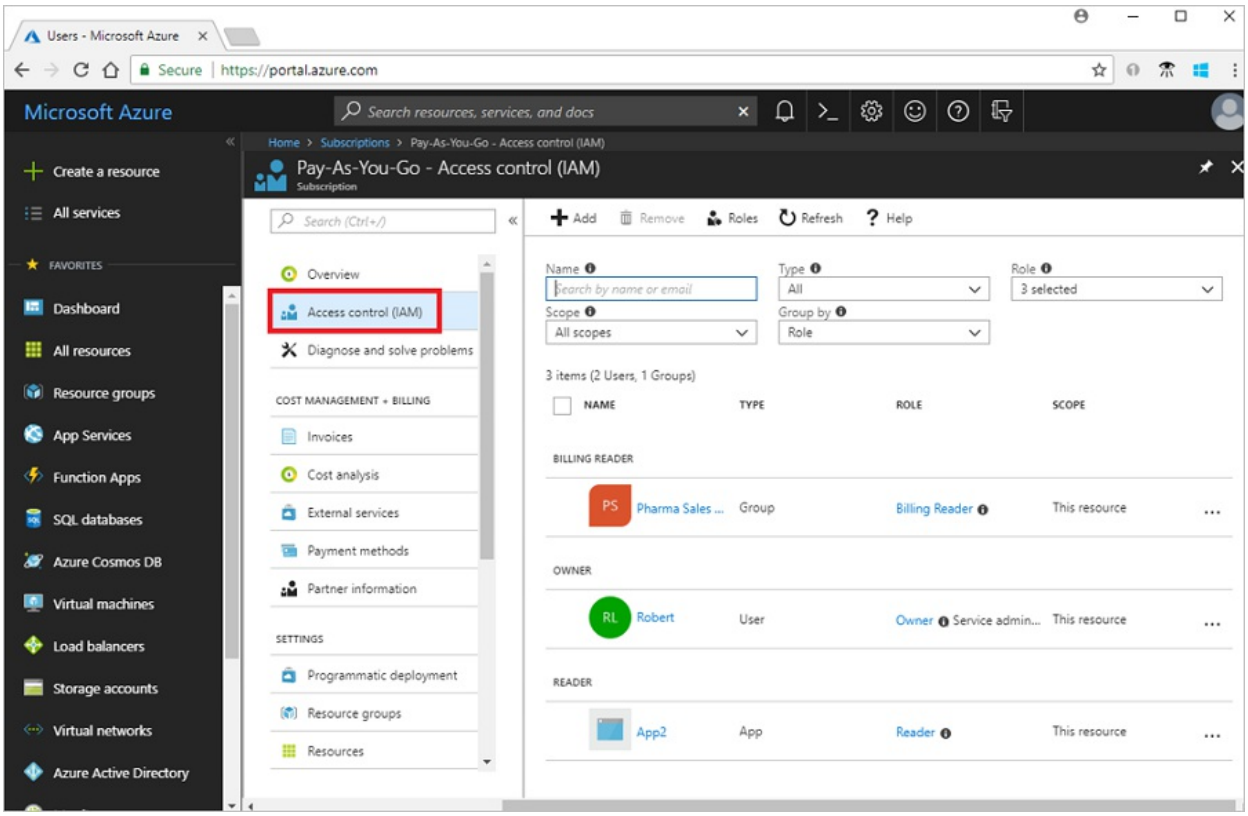


Figure 3: Access control.

Learn more:

- Learn about Azure RBAC.
- Learn to manage access via Azure RBAC and the Azure portal.
- Learn about custom roles.

Best practice: Review audit and security logs

Azure AD provides activity logs that appear in Azure Monitor. The logs capture the operations performed in Azure tenancy, when they occurred, and who performed them.

- Audit logs show the history of tasks in the tenant. Sign-in activity logs show who carried out the tasks.
- Access to security reports depends on your Azure AD license. With the free and basic licenses, you get a list of risky users and sign-ins. With the premium licenses, you get underlying event information.
- You can route activity logs to various endpoints for long-term retention and data insights.
- Make it a common practice to review the logs, or integrate your security information and event management (SIEM) tools to automatically review abnormalities. If you're not using a premium license, you'll need to do a lot of analysis yourself, or by using your SIEM system. Analysis includes looking for risky sign-ins and events, and other user attack patterns.

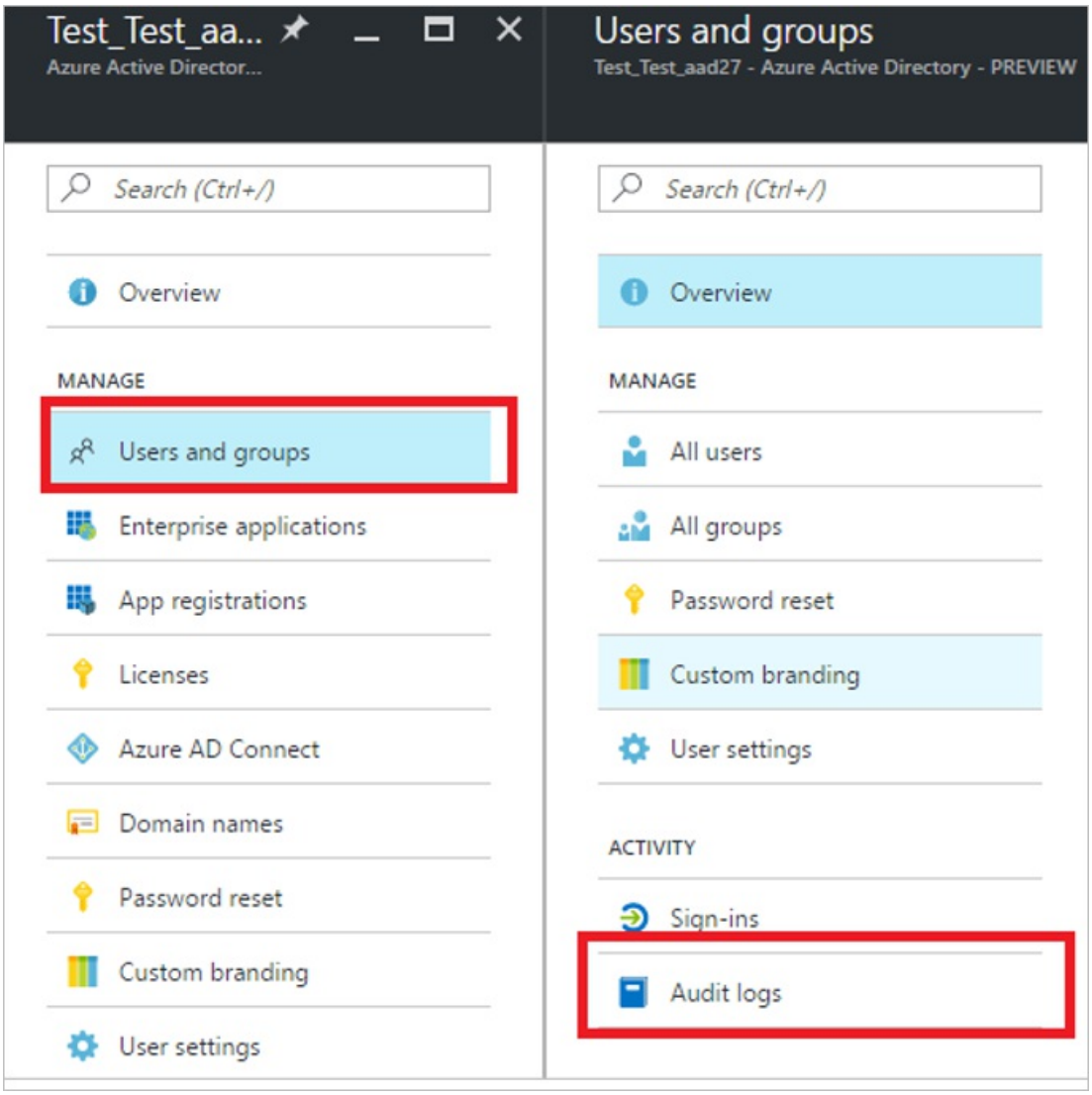


Figure 4: Azure AD users and groups.

Learn more:

- Learn about Azure AD activity logs in Azure Monitor.
- Learn how to audit activity reports in the Azure AD portal.

Best practice: Evaluate other security features

Azure provides other security features that provide advanced security options. Note that some of the following best practices require add-on licenses and premium options.

- **Implement Azure AD administrative units (AU).** Delegating administrative duties to support staff can be tricky with just basic Azure access control. Giving support staff access to administer all the groups in Azure AD might not be the ideal approach for organizational security. Using AU allows you to segregate Azure resources into containers in a similar way to on-premises organizational units (OUs). To use AUs, the AU admin must have a premium Azure AD license. For more information, see Administrative units management in Azure AD.
- **Use multi-factor authentication.** If you have a premium Azure AD license, you can enable and enforce multi-factor authentication on your admin accounts. Phishing is the most common way that accounts credentials are compromised. When a bad actor has admin account credentials, there's no stopping them from far-reaching actions, such as deleting all of your resource groups. You can establish multi-factor authentication in several ways, including with email, an authenticator app, and phone text messages. As an administrator, you can select the least intrusive option. Multi-factor authentication integrates with threat analytics and conditional access policies to randomly require a multi-factor authentication challenge

response. Learn more about security guidance, and how to set up multi-factor authentication.

- **Implement conditional access.** In most small and medium-sized organizations, Azure admins and the support team are probably located in a single geography. In this case, most sign-ins come from the same areas. If the IP addresses of these locations are fairly static, it makes sense that you shouldn't see administrator sign-ins from outside these areas. Even if a remote bad actor compromises an administrator's credentials, you can implement security features like conditional access, combined with multi-factor authentication, to prevent signing in from remote locations. This can also prevent spoofed locations from random IP addresses. Learn more about conditional access and review best practices for conditional access in Azure AD.
- **Review enterprise application permissions.** Over time, admins select Microsoft and third-party links without knowing their affect on the organization. Links can present consent screens that assign permissions to Azure apps. This might allow access to read Azure AD data, or even full access to manage your entire Azure subscription. You should regularly review the applications to which your admins and users have allowed access to Azure resources. Ensure that these applications have only the permissions that are necessary. Additionally, quarterly or semi-annually you can email users with a link to application pages, so that they're aware of the applications to which they've allowed access to their organizational data. For more information, see Unexpected application in my applications list, and how to control application assignments in Azure AD.

Managed migrated workloads

In the following sections, we'll recommend some best practices for Azure management, including:

- Best practices for Azure resource groups and resources, including smart naming, preventing accidental deletion, managing resource permissions, and effective resource tagging.
- Get a quick overview on using blueprints for building and managing your deployment environments.
- Review sample Azure architectures to learn from as you build your post-migration deployments.
- If you have multiple subscriptions, you can gather them into management groups, and apply governance settings to those groups.
- Apply compliance policies to your Azure resources.
- Put together a business continuity and disaster recovery (BCDR) strategy to keep data safe, your environment resilient, and resources up and running when outages occur.
- Group VMs into availability groups for resilience and high availability. Use managed disks for ease of VM disk and storage management.
- Enable diagnostic logging for Azure resources, build alerts and playbooks for proactive troubleshooting, and use the Azure dashboard for a unified view of your deployment health and status.
- Understand your Azure Support plan and how to implement it, get best practices for keeping VMs up-to-date, and put processes in place for change management.

Best practice: Name resource groups

Ensure that your resource groups have meaningful names that admins and support team members can easily recognize and scan. This can drastically improve productivity and efficiency.

If you're synchronizing your on-premises Active Directory to Azure AD by using Azure AD Connect, consider matching the names of security groups on-premises to the names of resource groups in Azure.

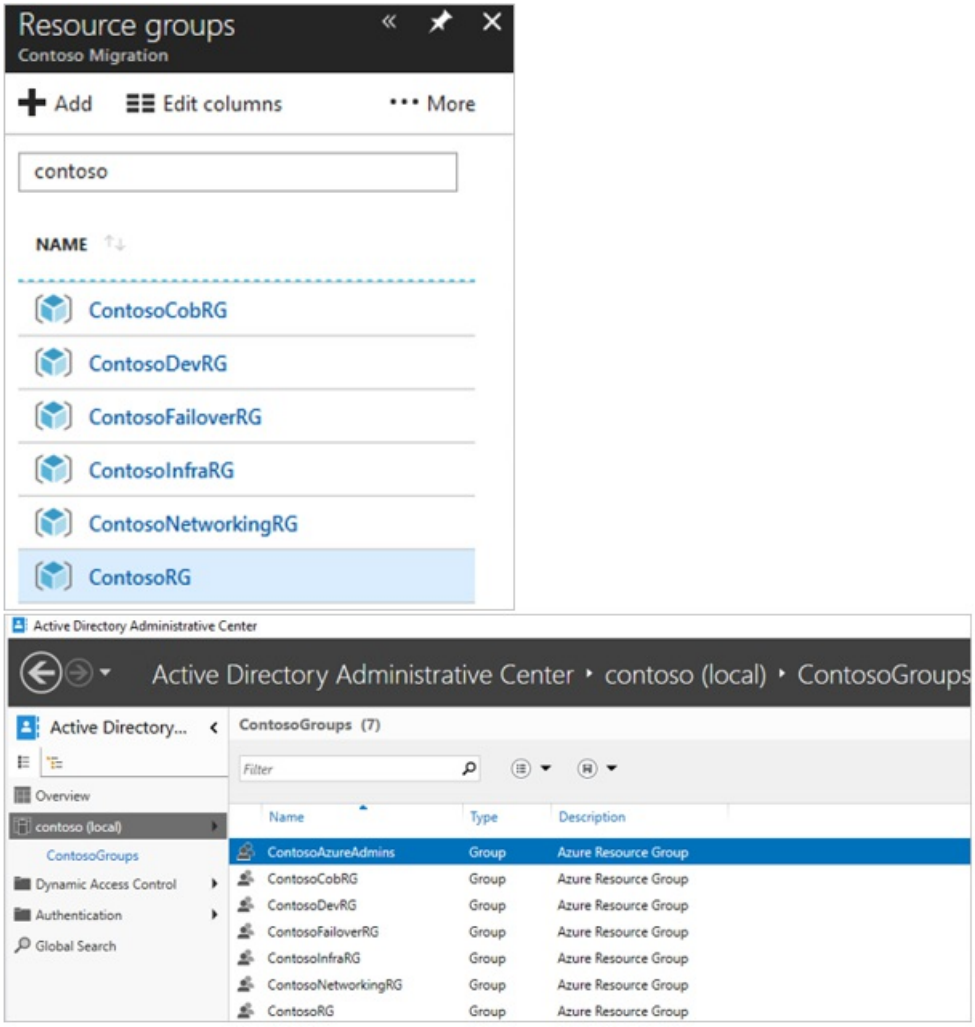


Figure 5: Resource group naming.

Learn more:

- Learn about recommended naming conventions.

Best practice: Implement delete locks for resource groups

The last thing you need is for a resource group to disappear because it was deleted accidentally. We recommend that you implement delete locks, so that this doesn't happen.

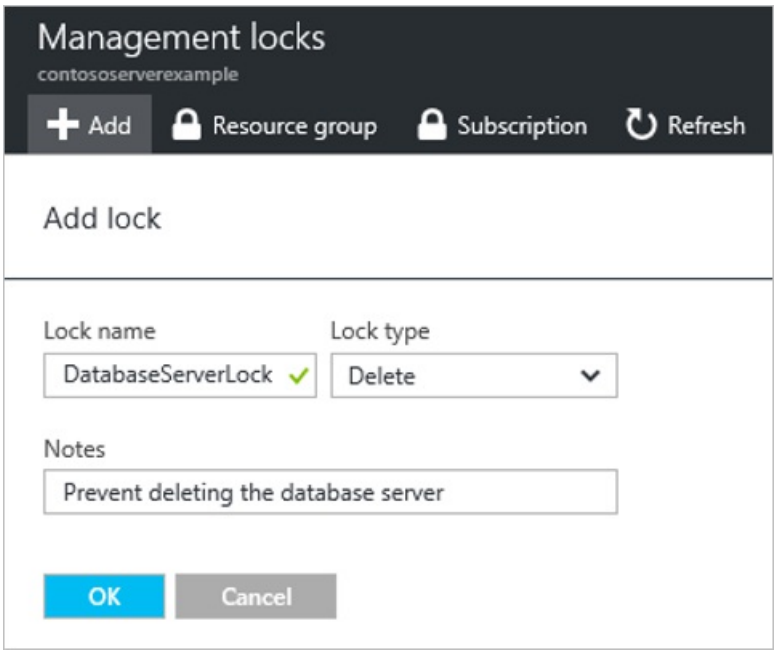


Figure 6: Delete locks.

Learn more:

- Learn about locking resources to prevent unexpected changes.

Best practice: Understand resource access permissions

A subscription owner has access to all the resource groups and resources in your subscription.

- Add people sparingly to this valuable assignment. Understanding the ramifications of these types of permissions is important in keeping your environment secure and stable.
- Make sure you place resources in appropriate resource groups:
  - Match resources with a similar lifecycle together. Ideally, you shouldn't need to move a resource when you need to delete an entire resource group.
  - Resources that support a function or workload should be placed together for simplified management.

Learn more:

- Learn about organizing subscriptions and resource groups.

Best practice: Tag resources effectively

Often, using only a resource group name related to resources won't provide enough metadata for effective implementation of mechanisms, such as internal billing or management within a subscription.

- As a best practice, use Azure tags to add useful metadata that can be queried and reported on.
- Tags provide a way to logically organize resources with properties that you define. Tags can be applied to resource groups or resources directly.
- Tags can be applied on a resource group or on individual resources. Resource group tags aren't inherited by the resources in the group.
- You can automate tagging by using PowerShell or Azure Automation, or tag individual groups and resources.
- If you have a request and change management system in place, then you can easily use the information in the request to populate your company-specific resource tags.

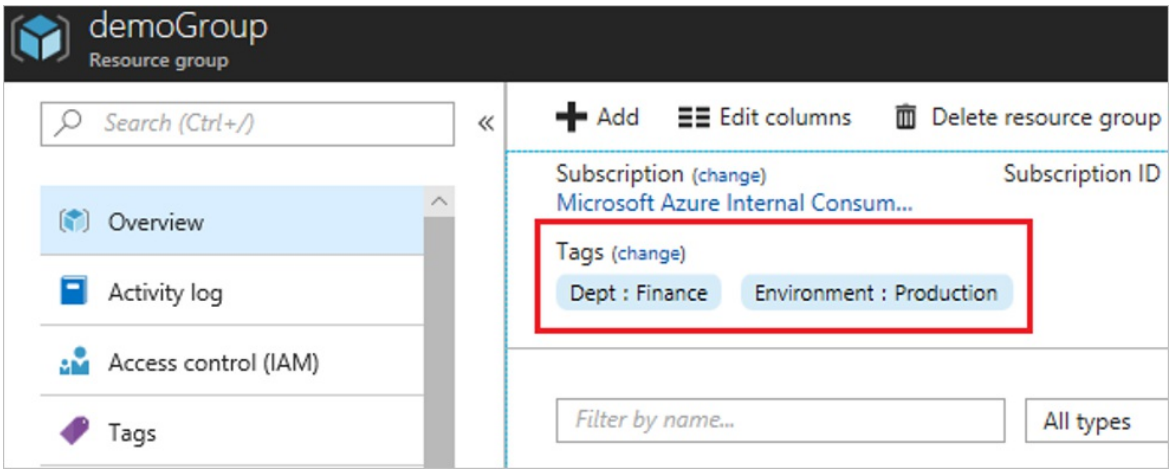


Figure 7: Tagging.

Learn more:

- Learn about tagging and tag limitations.
- Review PowerShell and CLI examples to set up tagging, and to apply tags from a resource group to its resources.

- Read Azure tagging best practices.

Best practice: Implement blueprints

Just as a blueprint allows engineers and architects to sketch a project's design parameters, the Azure Blueprints service enables cloud architects and central IT groups to define a repeatable set of Azure resources. This helps them to implement and adhere to an organization's standards, patterns, and requirements. Using Azure Blueprints, development teams can rapidly build and create new environments that meet organizational compliance requirements. These new environments have a set of built-in components, such as networking, to speed up development and delivery.

- Use blueprints to orchestrate the deployment of resource groups, Azure Resource Manager templates, and policy and role assignments.
- Store blueprints in a globally distributed service, Azure Cosmos DB. Blueprint objects are replicated to multiple Azure regions. Replication provides low latency, high availability, and consistent access to a blueprint, regardless of the region to which a blueprint deploys resources.

Learn more:

- Read about blueprints.
- Review an example blueprint for accelerating AI in healthcare.

Best practice: Review Azure reference architectures

Building secure, scalable, and manageable workloads in Azure can be daunting. With continual changes, it can be difficult to keep up with different features for an optimal environment. Having a reference to learn from can be helpful when designing and migrating your workloads. Azure and Azure partners have built several sample reference architectures for various types of environments. These samples are designed to provide ideas that you can learn from and build on.

Reference architectures are arranged by scenario. They contain best practices and advice on management, availability, scalability, and security. App Service Environment provides a fully isolated and dedicated environment for running applications, such as Windows and Linux web apps, Docker containers, mobile apps, and functions. App Service adds the power of Azure to your application, with security, load balancing, autoscaling, and automated management. You can also take advantage of its DevOps capabilities, such as continuous deployment from Azure DevOps and GitHub, package management, staging environments, custom domain, and SSL certificates. App Service is useful for applications that need isolation and secure network access, and those that use high amounts of memory and other resources that need to scale.

Learn more:

- Learn about Azure reference architectures.
- Review Azure example scenarios.

Best practice: Manage resources with Azure management groups

If your organization has multiple subscriptions, you need to manage access, policies, and compliance for them. Azure management groups provide a level of scope above subscriptions. Here are some tips:

- You organize subscriptions into containers called management groups, and apply governance conditions to them.
- All subscriptions in a management group automatically inherit the management group conditions.
- Management groups provide large-scale, enterprise-grade management, no matter what type of subscriptions you have.
- For example, you can apply a management group policy that limits the regions in which VMs can be created.

This policy is then applied to all management groups, subscriptions, and resources under that management group.

- You can build a flexible structure of management groups and subscriptions, to organize your resources into a hierarchy for unified policy and access management.

The following diagram shows an example of creating a hierarchy for governance by using management groups.

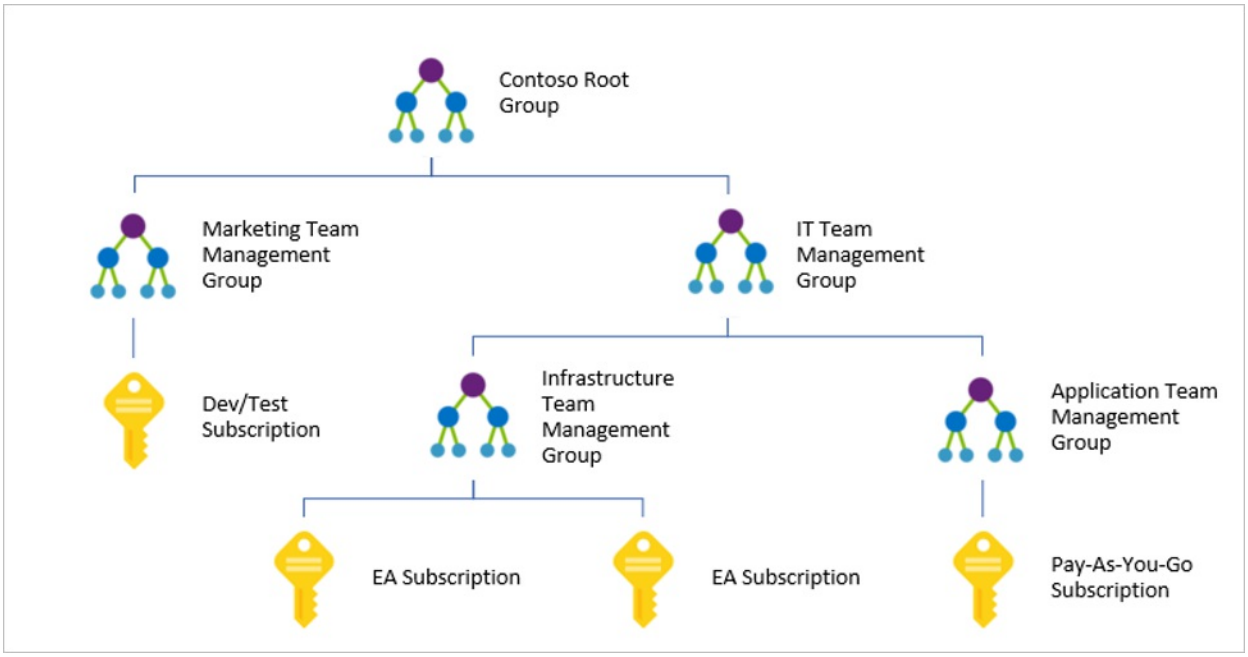


Figure 8: Management groups.

Learn more:

- Learn more about organizing resources into management groups.

Best practice: Deploy Azure Policy

Azure Policy is a service that you use to create, assign, and manage policies. Policies enforce different rules and effects over your resources, so those resources stay compliant with your corporate standards and service-level agreements.

Azure Policy evaluates your resources, scanning for those not compliant with your policies. For example, you can create a policy that allows only a specific SKU size for VMs in your environment. Azure Policy will evaluate this setting when you create and update resources, and when scanning existing resources. Note that Azure provides some built-in policies that you can assign, or you can create your own.

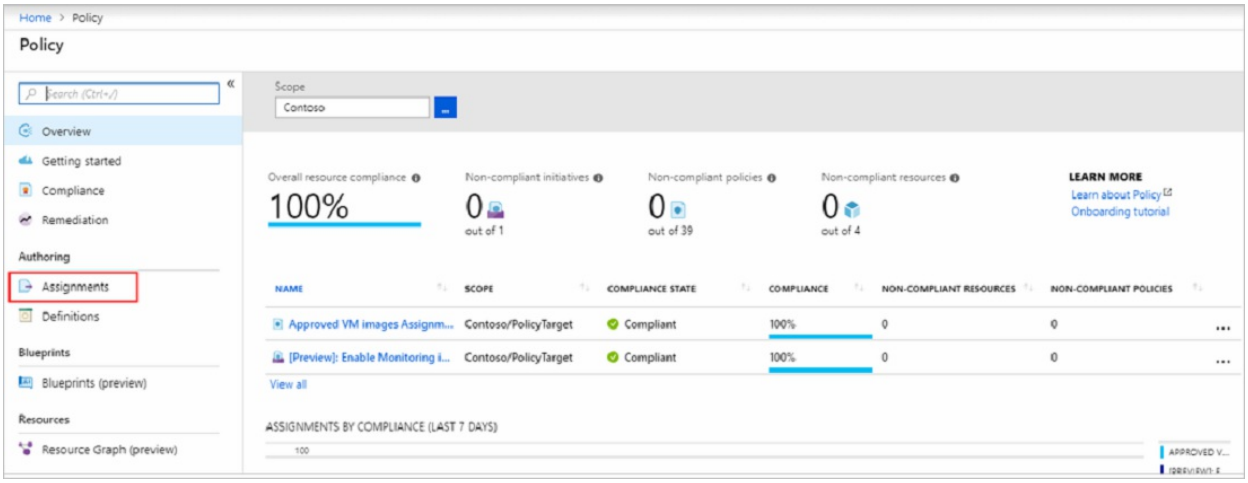


Figure 9: Azure Policy.

Learn more:

- Read the Azure Policy overview.
- Learn about creating and managing policies to enforce compliance.

Best practice: Implement a BCDR strategy

Planning for BCDR is a critical exercise that you should complete as part of your Azure migration planning process. In legal terms, your contracts might include a *force majeure* clause that excuses obligations due to a greater force, such as hurricanes or earthquakes. But you must ensure that essential services continue to run and recover when disaster strikes. Your ability to do this can make or break your company's future.

Broadly, your BCDR strategy must consider:

- **Data backup:** How to keep your data safe so that you can recover it easily if outages occur.
- **Disaster recovery:** How to keep your applications resilient and available if outages occur.

Set up BCDR

When migrating to Azure, understand that although the Azure platform provides some built-in resiliency capabilities, you need to design your Azure deployment to take advantage of them.

- Your BCDR solution will depend on your company objectives, and is influenced by your Azure deployment strategy. Infrastructure as a service (IaaS) and platform as a service (PaaS) deployments present different challenges for BCDR.
- After they are in place, your BCDR solutions should be tested regularly to check that your strategy remains viable.

Back up an IaaS deployment

In most cases, an on-premises workload is retired after migration, and your on-premises strategy for backing up data must be extended or replaced. If you migrate your entire datacenter to Azure, you'll need to design and implement a full backup solution by using Azure technologies, or third-party integrated solutions.

For workloads running on Azure IaaS VMs, consider these backup solutions:

- **Azure Backup:** Provides application-consistent backups for Azure Windows and Linux VMs.
- **Storage snapshots:** Takes snapshots of Blob Storage.

Azure Backup

Azure Backup creates data recovery points that are stored in Azure Storage. Azure Backup can back up Azure VM disks, and Azure Files (preview). Azure Files provide file shares in the cloud, accessible via Server Message Block (SMB).

You can use Azure Backup to back up VMs in the following ways:

- **Direct backup from VM settings.** You can back up VMs with Azure Backup directly from the VM options in the Azure portal. You can back up the VM once per day, and you can restore the VM disk as needed. Azure Backup takes application-aware data snapshots, and no agent is installed on the VM.
- **Direct backup in a Recovery Services vault.** You can back up your IaaS VMs by deploying an Azure Backup Recovery Services vault. This provides a single location to track and manage backups, as well as granular backup and restore options. Backup is up to three times a day, at the file and folder levels. It isn't application-aware, and Linux isn't supported. Install the Microsoft Azure recovery services (MARS) agent on each VM that you want to back up by using this method.
- **Protect the VM to Azure Backup server.** Azure Backup server is provided free with Azure Backup. The VM is backed up to local Azure Backup server storage. You then back up the Azure Backup server to Azure in a vault. Backup is application-aware, with full granularity over backup frequency and retention. You can back up at the application level, for example by backing up SQL Server or SharePoint.

For security, Azure Backup encrypts data in-flight by using AES-256. It sends it over HTTPS to Azure. Backed-up data-at-rest in Azure is encrypted by using Azure Storage encryption.

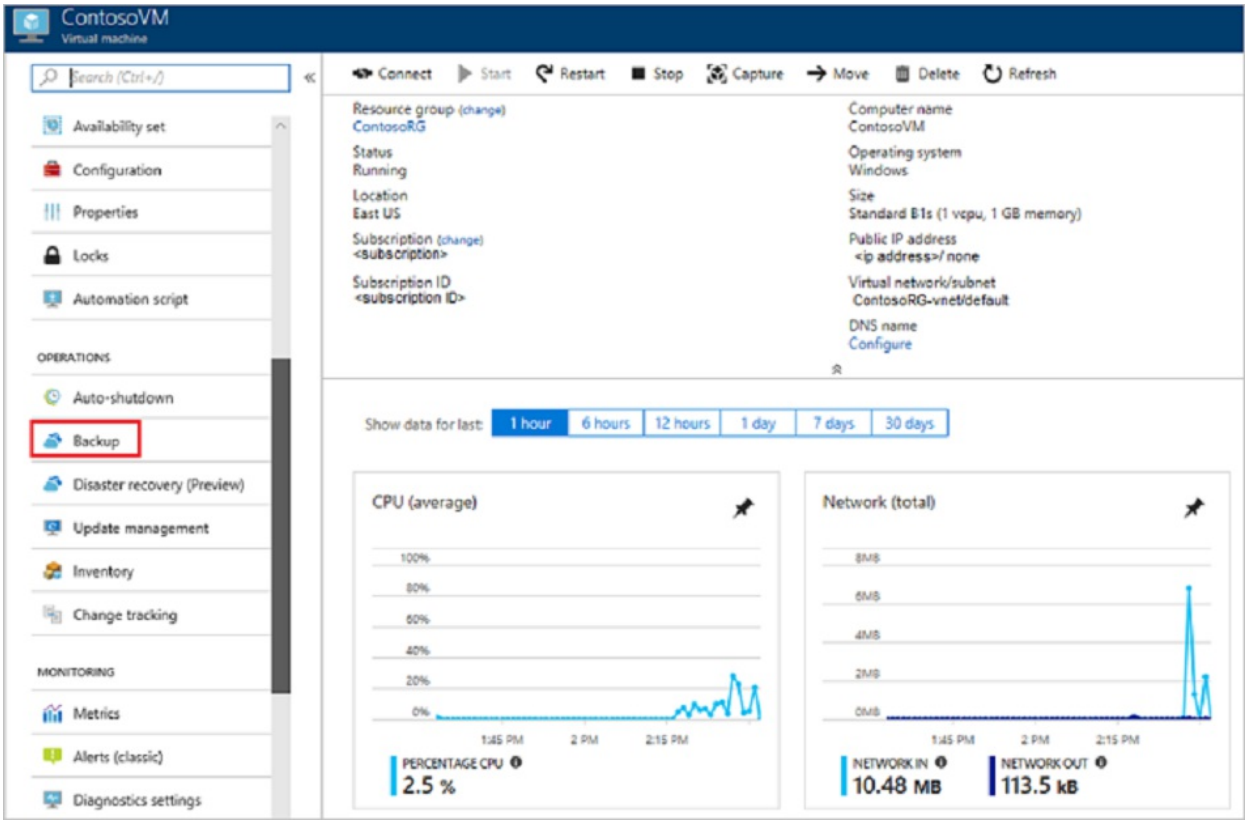


Figure 10: Azure Backup.

Learn more:

- Learn about Azure Backup.
- Plan a backup infrastructure for Azure VMs.

Storage snapshots

Azure VMs are stored as page blobs in Azure Storage. Snapshots capture the blob state at a specific point in time. As an alternative backup method for Azure VM disks, you can take a snapshot of storage blobs and copy them to another storage account.

You can copy an entire blob, or use an incremental snapshot copy to copy only delta changes and reduce storage space. As an extra precaution, you can enable soft delete for Blob Storage accounts. With this feature enabled, a blob that's deleted is marked for deletion, but not immediately purged. During the interim period, you can restore the blob.

Learn more:

- Learn about Azure Blob Storage.
- Learn how to create a blob snapshot.
- Review a sample scenario for Blob Storage backup.
- Read about soft delete for blobs.
- Disaster recovery and forced failover (preview) in Azure Storage

Third-party backup

In addition, you can use third-party solutions to back up Azure VMs and storage containers to local storage or other cloud providers. For more information, see Backup solutions in Azure Marketplace.

Set up disaster recovery for IaaS applications

In addition to protecting data, BCDR planning must consider how to keep applications and workloads available if a disaster occurs. For workloads that run on Azure IaaS VMs and Azure Storage, consider the solutions in the

following sections.

Azure Site Recovery

Azure Site Recovery is the primary Azure service for ensuring that Azure VMs can be brought online, and VM applications made available, when outages occur.

Site Recovery replicates VMs from a primary to a secondary Azure region. If disaster strikes, you fail VMs over from the primary region, and continue accessing them as normal in the secondary region. When operations return to normal, you can fail back VMs to the primary region.

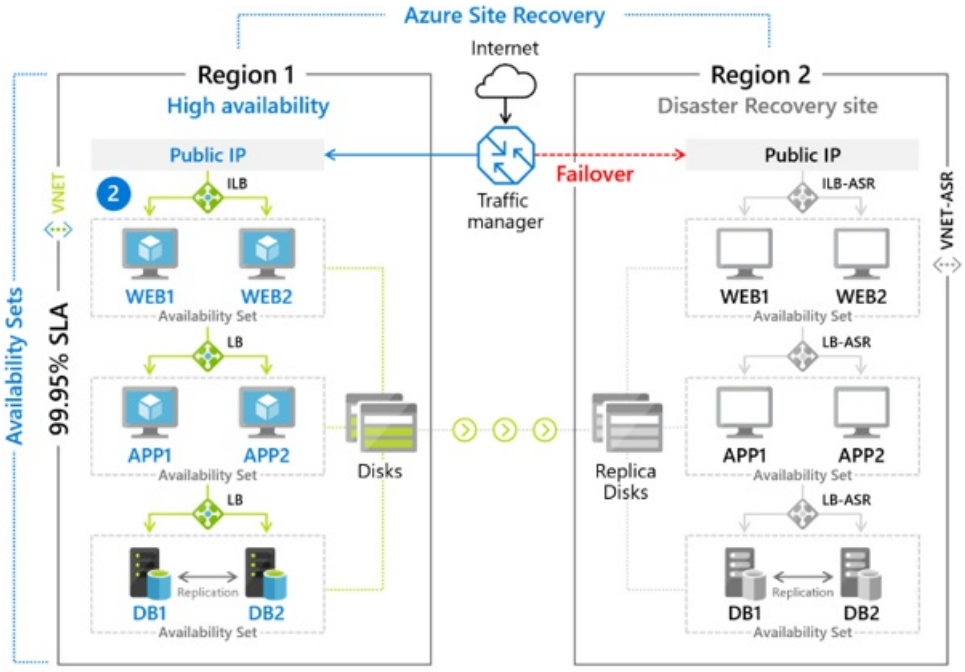


Figure 11: Site Recovery.

Learn more:

- Review disaster recovery scenarios for Azure VMs.
- Learn how to set up disaster recovery for an Azure VM after migration.

Best practice: Use managed disks and availability sets

Azure uses availability sets to logically group VMs together, and to isolate VMs in a set from other resources. VMs in an availability set are spread across multiple fault domains with separate subsystems, which protects against local failures. The VMs are also spread across multiple update domains, preventing a simultaneous reboot of all VMs in the set.

Azure managed disks simplify disk management for Azure Virtual Machines by managing the storage accounts associated with the VM disks.

- Use managed disks wherever possible. You only have to specify the type of storage you want to use and the size of disk you need, and Azure creates and manages the disk for you.
- You can convert existing disks to managed disks.
- You should create VMs in availability sets for high resilience and availability. When planned or unplanned outages occur, availability sets ensure that at least one VM in the set remains available.

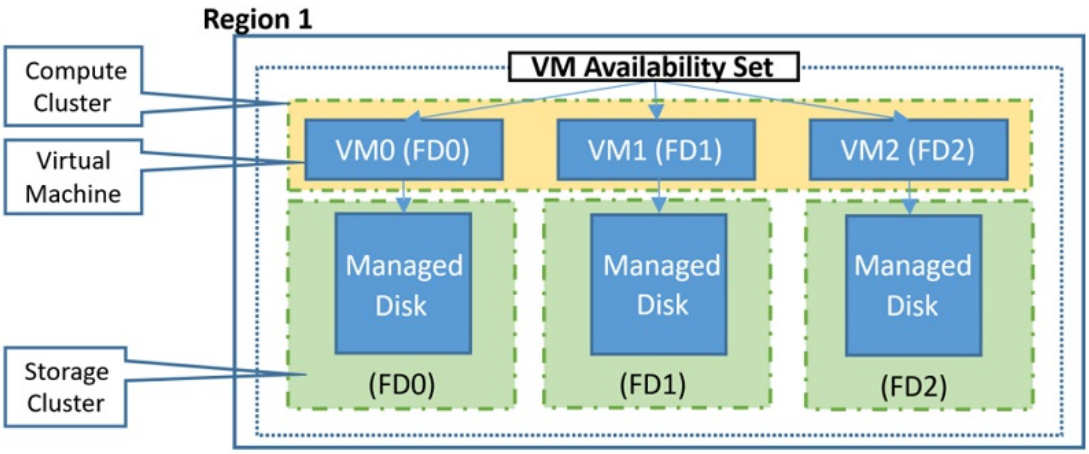


Figure 12: Managed disks.

Learn more:

- Read the managed disks overview.
- Learn about converting disks to managed.
- Learn how to manage the availability of Windows VMs in Azure.

Best practice: Monitor resource usage and performance

You might have moved your workloads to Azure for its immense scaling capabilities. But moving your workload doesn't mean that Azure will automatically implement scaling without your input. Here are two examples:

- If your marketing organization pushes a new television advertisement that drives 300 percent more traffic, this might cause site availability issues. Your newly migrated workload might hit assigned limits, and crash.
- If there's a distributed denial-of-service (DDoS) attack on your migrated workload, in this case you don't want to scale. You want to prevent the source of the attacks from reaching your resources.

These two cases have different resolutions, but for both you need insight into what's happening with usage and performance monitoring.

- Azure Monitor can help surface these metrics, and provide response with alerts, autoscaling, Event Hubs, and Logic Apps.
- You can also integrate your third-party SIEM application to monitor the Azure logs for auditing and performance events.

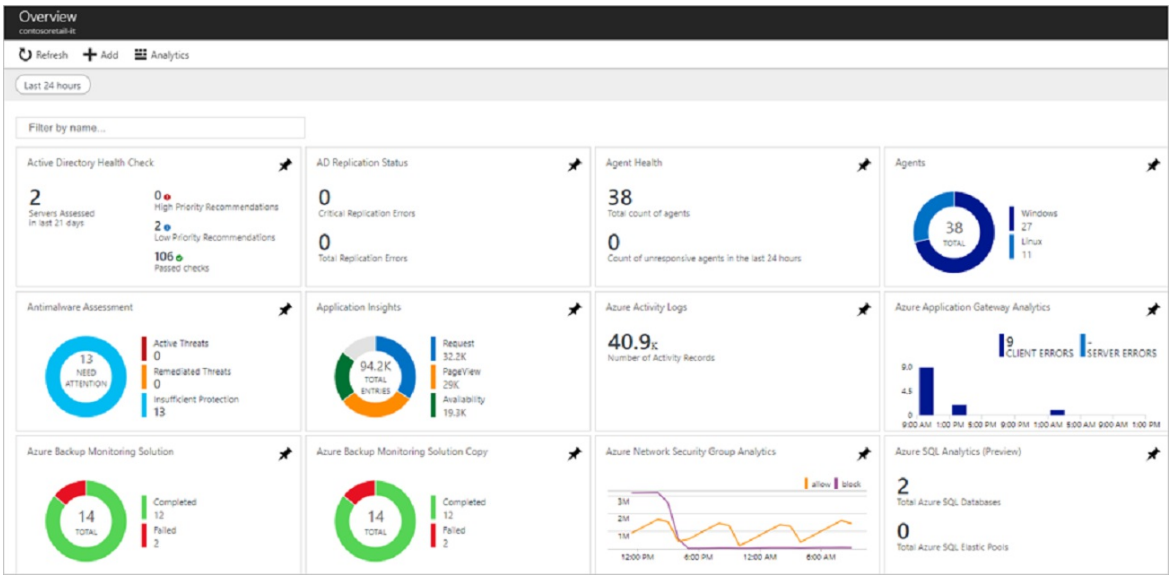


Figure 13: Azure Monitor.

Learn more:

- Learn about Azure Monitor.
- Get best practices for monitoring and diagnostics.
- Learn about autoscaling.
- Learn how to route Azure data to a SIEM tool.

Best practice: Enable diagnostic logging

Azure resources generate a fair number of logging metrics and telemetry data. By default, most resource types don't have diagnostic logging enabled. By enabling diagnostic logging across your resources, you can query logging data, and build alerts and playbooks based on it.

When you enable diagnostic logging, each resource will have a specific set of categories. You select one or more logging categories, and a location for the log data. Logs can be sent to a storage account, event hub, or to Azure Monitor Logs.

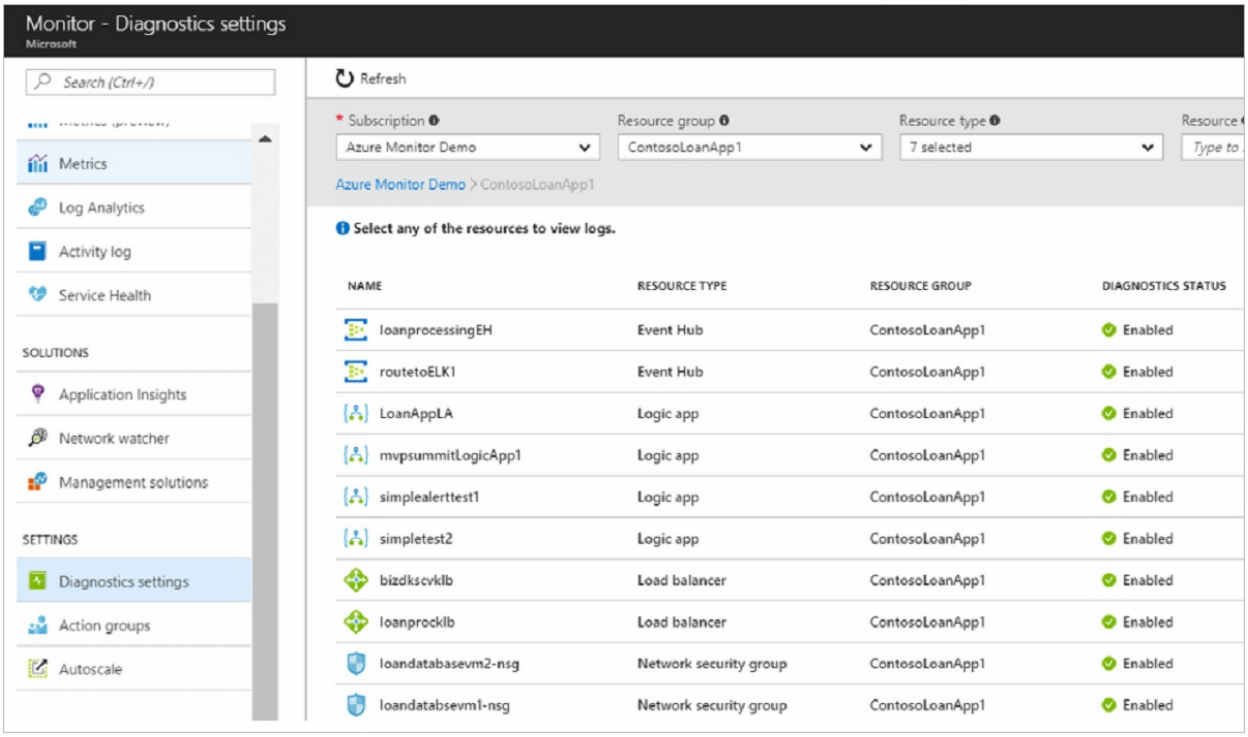


Figure 14: Diagnostic logging.

Learn more:

- Learn about collecting and consuming log data.
- Learn what's supported for diagnostic logging.

Best practice: Set up alerts and playbooks

With diagnostic logging enabled for Azure resources, you can start to use logging data to create custom alerts.

- Alerts proactively notify you when conditions are found in your monitoring data. You can then address issues before system users notice them. You can alert on metric values, log search queries, activity log events, platform health, and website availability.
- When alerts are triggered, you can run a logic app playbook. A playbook helps you to automate and orchestrate a response to a specific alert. Playbooks are based on Azure Logic Apps. You can use logic app templates to create playbooks, or create your own.
- As a simple example, you can create an alert that triggers when a port scan happens against an NSG. You can set up a playbook that runs and locks down the IP address of the scan origin.
- Another example is an application with a memory leak. When the memory usage gets to a certain point, a playbook can recycle the process.

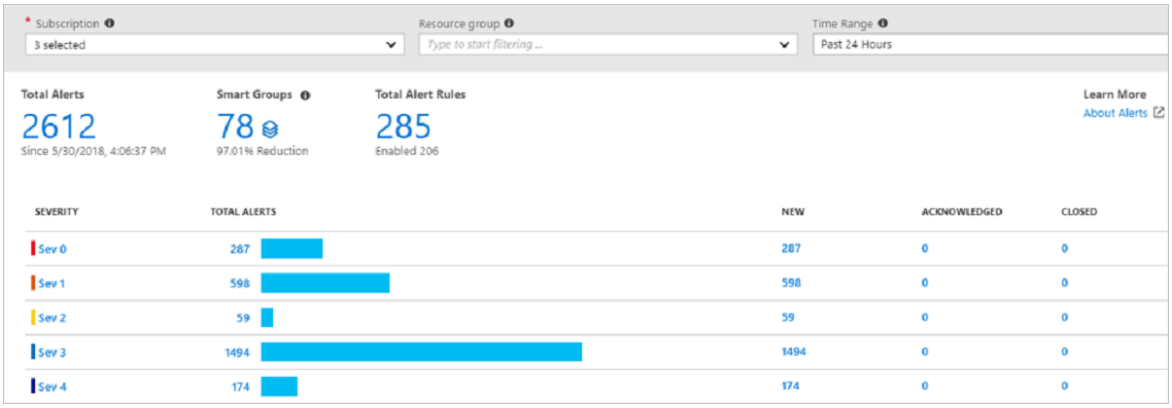


Figure 15: Alerts.

Learn more:

- Learn about alerts.
- Learn about security playbooks that respond to Security Center alerts.

Best practice: Use the Azure dashboard

The Azure portal is a web-based unified console that allows you to build, manage, and monitor everything from simple web apps to complex cloud applications. It includes a customizable dashboard and accessibility options.

- You can create multiple dashboards and share them with others who have access to your Azure subscriptions.
- With this shared model, your team has visibility into the Azure environment, which helps them be proactive when managing systems in the cloud.

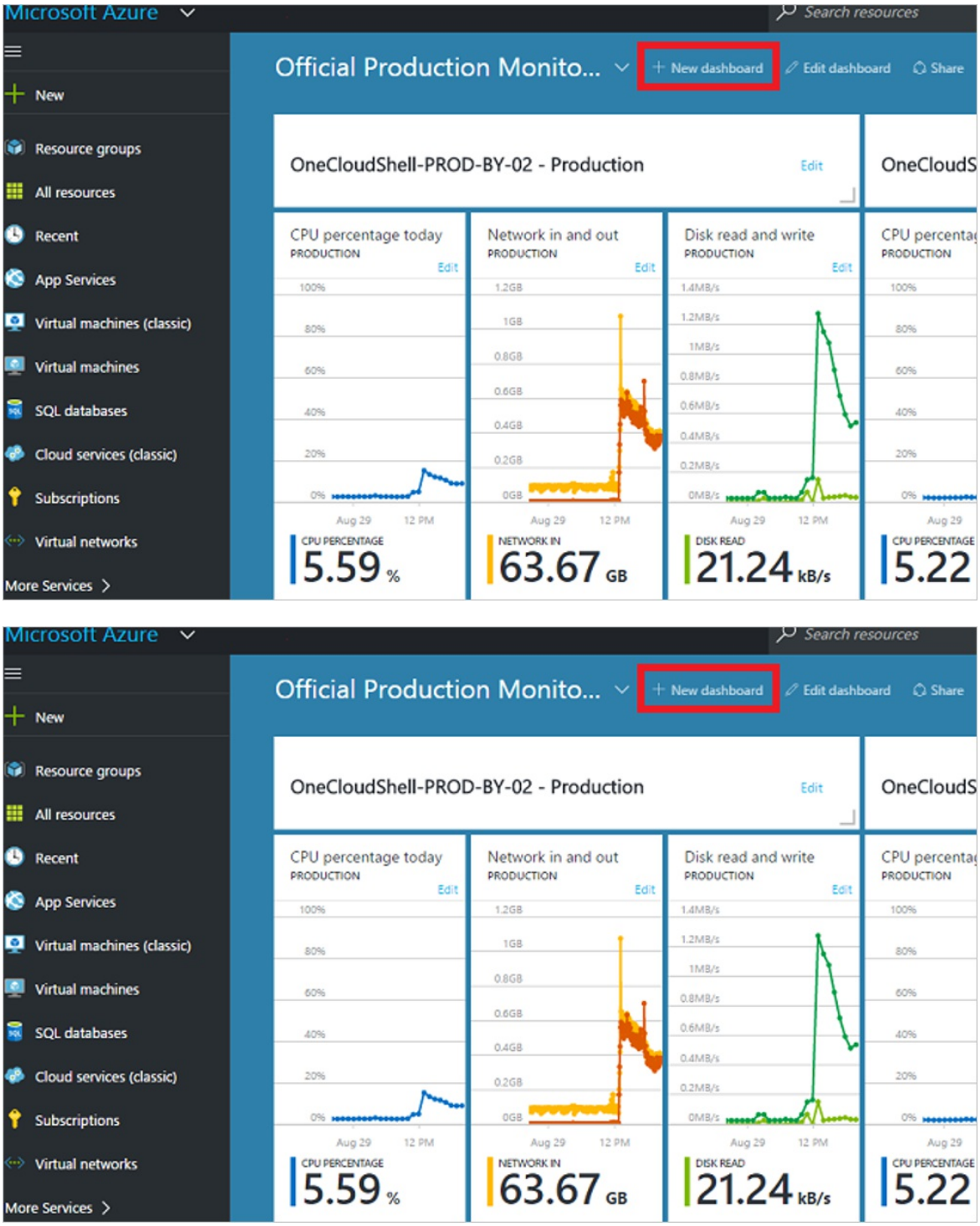


Figure 16: Azure dashboard.

Learn more:

- Learn how to create a dashboard.
- Learn about dashboard structure.

Best practice: Understand support plans

At some point, you will need to collaborate with your support staff or Microsoft support staff. Having a set of policies and procedures for support during scenarios such as disaster recovery is vital. In addition, your admins and support staff should be trained on implementing those policies.

- In the unlikely event that an Azure service issue affects your workload, admins should know how to submit a support ticket to Microsoft in the most appropriate and efficient way.
- Familiarize yourself with the various support plans offered for Azure. They range from response times

dedicated to developer instances, to premier support with a response time of less than 15 minutes.



	BASIC	DEVELOPER	STANDARD	PROFESSIONAL DIRECT	PREMIER
		Purchase support	Purchase support	Purchase support	Contact Premier
Case Severity/Response Times		Minimal business impact (Sev C): <8 business hours[1]	Minimal business impact (Sev C): <8 business hours[1] Moderate business impact (Sev B): <4 hours Critical business impact (Sev A): <1 hour	Minimal business impact (Sev C): <4 business hours[1] Moderate business impact (Sev B): <2 hours Critical business impact (Sev A): <1 hour	Minimal business impact (Sev C): <4 business hours[1] Moderate business impact (Sev B): <2 hours Critical business impact (Sev A): <1 hour <15 minutes (with Azure Rapid Response or Azure Event Management)

Figure 17: Support plans.

Learn more:

- Read an overview of Azure Support plans.
- Learn about service-level agreements (SLAs).

Best practice: Manage updates

Keeping Azure VMs updated with the latest operating system and software updates is a massive chore. The ability to surface all VMs, figure out which updates they need, and automatically push those updates is extremely valuable.

- You can use Update Management in Azure Automation to manage operating system updates. This applies to machines that run Windows and Linux computers that are deployed in Azure, on-premises, and in other cloud providers.
- Use Update Management to quickly assess the status of available updates on all agent computers, and manage update installation.
- You can enable Update Management for VMs directly from an Azure Automation account. You can also update a single VM from the VM page in the Azure portal.
- In addition, you can register Azure VMs with System Center Configuration Manager. You can then migrate the Configuration Manager workload to Azure and do reporting and software updates from a single web interface.

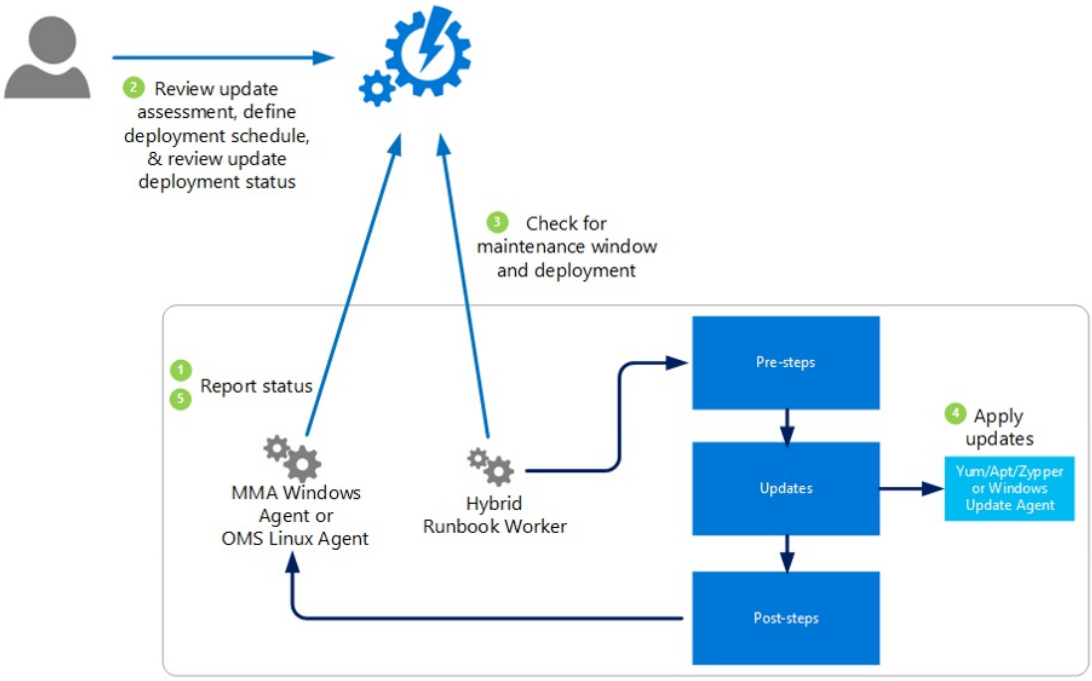


Figure 18: VM updates.

Learn more:

- Learn about Update Management in Azure.
- Learn how to integrate Configuration Manager with Update Management.
- Frequently asked questions about Configuration Manager in Azure.

Implement a change management process

As with any production system, making any type of change can affect your environment. A change management process that requires requests to be submitted in order to make changes to production systems is a valuable addition in your migrated environment.

- You can build best practice frameworks for change management to raise awareness in administrators and support staff.
- You can use Azure Automation to help with configuration management and change tracking for your migrated workflows.
- When enforcing change management process, you can use audit logs to link Azure change logs to existing change requests. Then, if you see a change made without a corresponding change request, you can investigate what went wrong in the process.

Azure has a change-tracking solution in Azure Automation:

- The solution tracks changes to Windows and Linux software and files, Windows registry keys, Windows services, and Linux daemons.
- Changes on monitored servers are sent to Azure Monitor for processing.
- Logic is applied to the received data, and the cloud service records the data.
- On the change tracking dashboard, you can easily see the changes that were made in your server infrastructure.

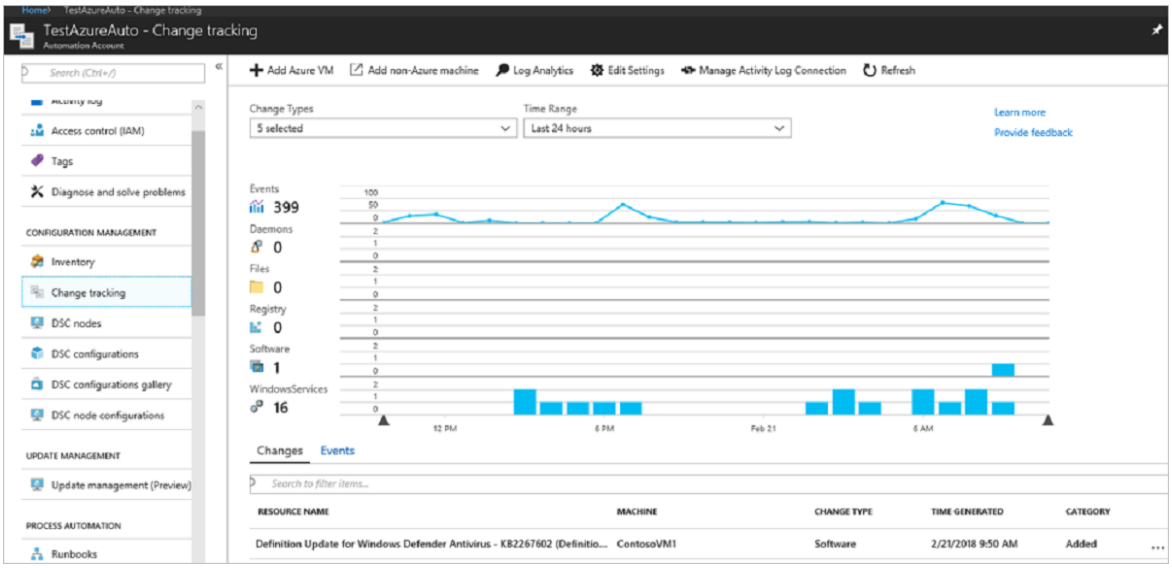


Figure 19: A change management chart.

Learn more:

- Learn about change tracking.
- Learn about Azure Automation capabilities.

Next steps

Review other best practices:

- Best practices for networking after migration.
- Best practices for cost management after migration.

Azure cloud migration best practices checklist

5/18/2021 • 2 minutes to read • Edit Online

Start with the Azure migration guide in the Cloud Adoption Framework if you're interested in migrating to Azure. That guide walks you through a set of tools and basic approaches to migrating virtual machines to the cloud.

The following checklists provide Azure cloud migration best practices that go beyond the basic cloud-native tools. These checklists outline the common areas of complexity that might require the scope of the migration to expand beyond the Azure migration guide.

Migration best practices for business-driven scope expansion

- Support global markets: The business operates in multiple geographic regions with disparate data sovereignty requirements. To meet those requirements, you should factor in additional considerations to the prerequisite review and distribution of assets during migration.

Migration best practices for technology-driven scope expansion

- VMware migration: Migrating VMware hosts can accelerate the overall migration process. Each migrated VMware host can move multiple workloads to the cloud. After migration, those VMs and workloads can stay in VMware, or be migrated to modern cloud capabilities.
- SQL Server migration: Migrating instances of SQL Server can accelerate the overall migration process. Each migrated instance can move multiple databases and services, potentially accelerating multiple workloads.
- Multiple datacenters: Migrating multiple datacenters adds significant complexity. During each process of the move (assess, migrate, optimize, and manage), other considerations are discussed to prepare for more complex environments.
- Data requirements exceed network capacity: Companies frequently choose to migrate to the cloud because the capacity, speed, or stability of an existing datacenter is no longer satisfactory. Unfortunately, those same constraints add complexity to the migration process, requiring additional planning during the assessment and migration processes.
- Governance or compliance strategy: When governance and compliance are vital to the success of a migration, IT governance teams and the cloud adoption team must ensure additional alignment with one another.

Additional migration best practices

- Set up networking for workloads migrated to Azure
- Deploy a migration infrastructure
- Cost and size workloads migrated to Azure
- Scale a migration to Azure

Next steps

The following step is a good starting point for reviewing Azure migration best practices.

Multiple datacenters

Multiple datacenters

3/6/2021 • 2 minutes to read • Edit Online

Often the scope of a migration involves the transition of multiple datacenters. The following guidance expands the scope of the Azure migration guide to address multiple datacenters.

General scope expansion

Most of the effort required in this scope expansion occurs during the prerequisites, assess, and optimize processes of a migration.

Suggested prerequisites

Before starting the migration, you should create epics within the project management tool for each datacenter that's to be migrated. Each epic represents a datacenter. It's important to understand the business outcomes and motivations for this migration. Use those motivations to prioritize the list of epics (or datacenters). For instance, if migration is driven by a desire to exit datacenters before leases must be renewed, then each epic would be prioritized based on lease renewal date.

Within each epic, the workloads to be assessed and migrated are managed as features. Each asset within that workload is managed as a user story. The work required to assess, migrate, optimize, promote, secure, and manage each asset is represented as tasks for each asset.

Sprints or iterations then consist of a series of tasks required to migrate the assets and user stories committed to by the cloud adoption team. Releases then consist of one or more workloads or features to be promoted to production.

Assess process changes

When you're expanding the scope to address multiple datacenters, the biggest change to the assess process is related to the accurate recording and prioritization of workloads and dependencies across datacenters.

Suggested action during the assess process

Evaluate cross-datacenter dependencies: The dependency visualization tools in Azure Migrate can help pinpoint dependencies. Using this toolset before migration is generally a best practice. But when dealing with global complexity, it becomes a necessary step in the assessment process. Through dependency grouping, the visualization can help identify the IP addresses and ports of any assets required to support the workload.

> **IMPORTANT**
>
> - A subject matter expert with an understanding of asset placement and IP address schemas is required to identify assets that reside in a secondary datacenter.
> - Evaluate both downstream dependencies and clients in the visualization to understand bidirectional dependencies.

Migration process changes

Migrating multiple datacenters is similar to consolidating datacenters. After migration, the cloud becomes the singular datacenter solution for multiple assets. The most likely scope expansion during the migration process is the validation and alignment of IP addresses.

Suggested action during the migration process

The following are activities that greatly affect the success of a cloud migration:

- **Evaluate network conflicts:** When you're consolidating datacenters into a single cloud provider, you're likely to create network, DNS, or other conflicts. During migration, it's important to test for conflicts to avoid interruptions to production systems hosted in the cloud.
- **Update routing tables:** Often, modifications to routing tables are required when consolidating networks or datacenters.

Optimize and promote process changes

During optimization, additional testing might be required.

Suggested action during the optimize and promote process

Prior to promotion, provide additional levels of testing during this scope expansion. During testing, it's important to test for routing or other network conflicts. Further, it's important to isolate the deployed application, and retest to validate that all dependencies have been migrated to the cloud. In this case, isolation means separating the deployed environment from production networks. Doing so can catch overlooked assets that are still running on-premises.

Secure and manage process changes

Secure and manage processes are unlikely to be changed by this scope expansion.

Next steps

Return to the checklist to ensure that your migration method is fully aligned.

Migration best practices checklist

Azure regions decision guide

3/6/2021 • 15 minutes to read • Edit Online

Azure comprises many regions around the world. Each Azure region has specific characteristics that make choosing which region to use incredibly important. These include available services, capacity, constraints, and sovereignty:

- **Available services:** Services that are deployed to each region differ, based on various factors. Select a region for your workload that contains your desired service. For more information, see Products available by region.
- **Capacity:** Each region has a maximum capacity. This can affect which types of subscriptions can deploy which types of services and under what circumstances. This is different than subscription quotas. If you're planning a large-scale datacenter migration to Azure, you might want to consult with your local Azure field team or account manager to confirm that you can deploy at the scale necessary.
- **Constraints:** Certain constraints are placed on the deployment of services in certain regions. For example, some regions are only available as a backup or failover target. Other constraints that are important to note are data sovereignty requirements.
- **Sovereignty:** Certain regions are dedicated to specific sovereign entities. While all regions are Azure regions, these sovereign regions are completely isolated from the rest of Azure. They aren't necessarily managed by Microsoft and might be restricted to certain types of customers. These sovereign regions are:
  - Azure China 21Vianet
  - Azure Germany: Azure Germany is being deprecated in favor of standard nonsovereign Azure regions in Germany.
  - Azure US government
  - Two regions in Australia are managed by Microsoft but are provided for the Australian government and its customers and contractors. Therefore, these regions carry client constraints similar to the other sovereign clouds.

Operate in multiple geographic regions

When businesses operate in multiple geographic regions, which can be essential for resiliency, this introduces potential complexity in the following forms:

- Asset distribution
- User access profiles
- Compliance requirements
- Regional resiliency

Regional selection is very important to your overall cloud adoption strategy. Let's start with network considerations.

Network considerations

Any robust cloud deployment requires a well-considered network that takes into account Azure regions. You should account for the following:

- Azure regions are deployed in pairs. In the event of a catastrophic region failure, another region within the same geopolitical boundary is designated as its paired region. Consider deploying into paired regions as a primary and secondary resiliency strategy. One exception to this strategy is `Brazil South`, which is

paired with `South Central US`. For more information, see Azure paired regions.

- Azure Storage supports geo-redundant storage (GRS). This means that three copies of your data are stored within your primary region, and three additional copies are stored in the paired region. You can't change the storage pairing for GRS.
- Services that rely on Azure Storage GRS can take advantage of this paired region capability. To do so, your applications and the network must be oriented to support that.
- If you don't plan to use GRS to support your regional resiliency needs, you shouldn't use the paired region as your secondary. In the event of a regional failure, there will be intense pressure on resources in the paired region as resources migrate. You can avoid that pressure by recovering to an alternate site and gaining additional speed during your recovery.

> **WARNING**
> Do not attempt to use Azure GRS for VM backups or recovery. Instead, use Azure Backup and Azure Site Recovery, along with Azure managed disks, to support your infrastructure as a service (IaaS) workload resiliency.

- Azure Backup and Azure Site Recovery work in tandem with your network design to facilitate regional resiliency for your IaaS and data backup needs. Make sure the network is optimized so data transfers remain on the Microsoft backbone and use virtual network peering, if possible. Some larger organizations with global deployments might instead use ExpressRoute Premium, to route traffic between regions and potentially save regional egress charges.
- Azure resource groups are regional specific. It's normal, however, for resources within a resource group to span multiple regions. Consider that in the event of a regional failure, control plane operations against a resource group will fail in the affected region, even though the resources in other regions (within that resource group) will continue to operate. This can affect both your network design and your resource group design.
- Many platform as a service (PaaS) services within Azure Support service endpoints or Azure Private Link. Both of these solutions affect your network considerations substantially with regard to regional resiliency, migration, and governance.
- Many PaaS services rely on their own regional resiliency solutions. For example, both Azure SQL Database and Azure Cosmos DB allow you to easily replicate to additional regions. Services such as Azure DNS don't have regional dependencies. As you consider which services you will use in your adoption process, make sure to clearly understand the failover capabilities and recovery steps that can be required for each Azure service.
- In addition to deploying to multiple regions to support disaster recovery, many organizations choose to deploy in an active-active pattern to not rely on failover. This method offers the additional benefits of global load balancing, additional fault tolerance, and network performance boosts. To take advantage of this pattern, your applications must support running active-active in multiple regions.

> **WARNING**
> Azure regions are highly available constructs, with SLAs applied to the services running in them. But you should never take a single region dependency on mission-critical applications. Always plan for regional failure, and practice recovery and mitigation steps.

After considering the network topology, you must next look at additional documentation and process alignment that might be necessary. The following approach can help assess the potential challenges and establish a general course of action:

- Consider a more robust readiness and governance implementation.
- Inventory the affected geographies. Compile a list of the regions and countries that are affected.
- Document data sovereignty requirements. Do the countries identified have compliance requirements that govern data sovereignty?
- Document the user base. Will employees, partners, or customers in the identified country be affected by the cloud migration?
- Document datacenters and assets. Are there assets in the identified country that might be included in the migration effort?
- Document regional SKU availability and failover requirements.

Align changes across the migration process to address the initial inventory.

Document complexity

The following table can aid in documenting the findings from the previous steps:

REGION	COUNTRY	LOCAL EMPLOYEES	LOCAL EXTERNAL USERS	LOCAL DATACENTERS OR ASSETS	DATA SOVEREIGNTY REQUIREMENTS
North America	United States	Yes	Partners and customers	Yes	No
North America	Canada	No	Customers	Yes	Yes
Europe	Germany	Yes	Partners and customers	No - network only	Yes
Asia Pacific	South Korea	Yes	Partners	Yes	No

Relevance of data sovereignty

Around the world, government organizations have begun establishing data sovereignty requirements, like General Data Protection Regulation (GDPR). Compliance requirements of this nature often require localization within a specific region or even within a specific country to protect their citizens. In some cases, data pertaining to customers, employees, or partners must be stored on a cloud platform within the same region as the end user.

Addressing this challenge has been a significant motivation for cloud migrations for companies that operate on a global scale. To maintain compliance requirements, some companies have chosen to deploy duplicate IT assets to cloud providers within the region. In the preceding table, Germany is a good example of this scenario. This example includes customers, partners, and employees but not current IT assents in Germany. This company might choose to deploy some assets to a datacenter within the GDPR area, potentially using the German Azure datacenters. An understanding of the data affected by GDPR would help the cloud adoption team understand the best migration approach in this case.

Why is the location of end users relevant?

Companies that support end users in multiple countries have developed technical solutions for addressing end-user traffic. In some cases, this involves localization of assets. In other scenarios, the company might choose to implement global wide area network (WAN) solutions to address disparate user bases via network focused solutions. In either case, the migration strategy can be affected by the usage profiles of those disparate end users.

Because the company supports employees, partners, and customers in Germany without currently having

datacenters there, this company probably implemented a leased-line solution. This type of solution routes traffic to datacenters in other countries. This existing routing presents a significant risk to the perceived performance of migrated applications. Injecting additional hops in an established and tuned global WAN can create the perception of underperforming applications after migration. Finding and fixing those issues can add significant delays to a project.

In each of the following processes, guidance for addressing this complexity is included across prerequisites, assess, migrate, and optimize processes. Understanding user profiles in each country is critical to properly manage this complexity.

Why is the location of datacenters relevant?

The location of existing datacenters can affect a migration strategy. For example:

Architecture decisions: The target region is one of the first steps in migration strategy design. This is often influenced by the location of the existing assets. Additionally, the availability of cloud services and the unit cost of those services can vary from one region to the next. Understanding where current and future assets are located affects architecture decisions and can influence budget estimates.

Datacenter dependencies: The data in the preceding table shows that dependencies between various global datacenters are likely. Those dependencies might not be documented or understood clearly in many organizations that operate on this type of scale. Your company's approach to evaluating user profiles helps to identify some of these dependencies in your organization. In addition, your team should explore additional assessment steps that can mitigate the risks and complexities that arise from dependencies.

Implement the general approach

The following approach uses a data-driven model for addressing global migration complexities. When the scope for a migration includes multiple regions, the cloud adoption team should evaluate the following readiness considerations:

- Data sovereignty might require localization of some assets, but many assets might not be governed by those compliance constraints. Things like logging, reporting, network routing, identity, and other central IT services might be eligible to be hosted as shared services across multiple subscriptions, or even multiple regions. The cloud adoption team should evaluate by using a shared service model for those services, as outlined in the reference architecture for a hub and spoke topology with shared services.
- When you're deploying multiple instances of similar environments, an environment factory can create consistency, improve governance, and accelerate deployment. The governance guide for complex enterprises establishes an approach that creates an environment that scales across multiple regions.

When the team is comfortable with the baseline approach and readiness is aligned, you should then consider a few data-driven prerequisites:

- **General discovery:** Complete the documenting complexity table.
- **Perform a user profile analysis on each affected country:** It's important to understand general end-user routing early in the migration process. Changing global lease lines and adding connections like ExpressRoute to a cloud datacenter can require months of networking delays. Address this as early in the process as possible.
- **Initial digital estate rationalization:** Whenever complexity is introduced into a migration strategy, you should complete an initial digital estate rationalization. See the guidance on digital estate rationalization.
  - **Additional digital estate requirements:** Establish tagging policies to identify any workload affected by data sovereignty requirements. Required tags should begin in the digital estate rationalization and carry through to the migrated assets.
- **Evaluate a hub and spoke model:** Distributed systems often share common dependencies. Those dependencies can often be addressed through the implementation of a hub and spoke model. While such a

model is out of scope for the migration process, it should be flagged for consideration during future iterations of the ready processes.

- **Prioritization of the migration backlog:** When network changes are required to support the production deployment of a workload that supports multiple regions, the cloud strategy team should track and manage escalations regarding those network changes. The higher level of executive support helps to accelerate the change by freeing the strategy team to reprioritize the backlog and ensure that global workloads aren't blocked by network changes. Such workloads should only be prioritized after the network changes are complete.

These prerequisites help establish processes that can address this complexity during execution of the migration strategy.

Assess process changes

When facing global asset and user base complexities in migration scenarios, you should add a few key activities to assessing your migration candidates. These activities produce data to clarify obstacles and outcomes for global users and assets.

Suggested action during the assess process

Evaluate cross-datacenter dependencies: The dependency visualization tools in Azure Migrate can help pinpoint dependencies. Using these tools before migration is a best practice. When dealing with global complexity, it becomes a necessary step to the assessment process. Through dependency grouping, the visualization can help identify the IP addresses and ports of any assets required to support the workload.

> **IMPORTANT**
>
> - A subject matter expert with an understanding of asset placement and IP address schemas is required to identify assets that reside in a secondary datacenter.
> - Evaluate both downstream dependencies and clients in the visualization to understand bidirectional dependencies.

Identify global user impact: The outputs from the prerequisite user profile analysis should identify any workload affected by global user profiles. When a migration candidate is in the affected workload list, the architect preparing for migration should consult networking and operations subject matter experts. They help to validate network routing and performance expectations. At a minimum, the architecture should include an ExpressRoute connection between the closest network operations center and Azure. The reference architecture for ExpressRoute connections can aid in the configuration of the necessary connection.

Design for compliance: The outputs from the prerequisite user profile analysis should identify any workload affected by data sovereignty requirements. During the architecture activities of the assess process, the assigned architect should consult compliance subject matter experts. They help to understand any requirements for migration and deployment across multiple regions. Those requirements significantly affect design strategies. The reference architectures for multiregion web applications and multiregion n-tier applications can assist design.

> **WARNING**
>
> When you're using either of the reference architectures above, it might be necessary to exclude specific data elements from replication processes to adhere to data sovereignty requirements. This will add an additional step to the promotion process.

Migration process changes

When you're migrating an application that must be deployed to multiple regions, the cloud adoption team must

account for a few considerations. These consist of Azure Site Recovery vault design, configuration and process server design, network bandwidth designs, and data synchronization.

Suggested action during the migration process

Azure Site Recovery vault design: Azure Site Recovery is the suggested tool for cloud-native replication and synchronization of digital assets to Azure. Site Recovery replicates data about the asset to a Site Recovery vault, which is bound to a specific subscription in a specific region and Azure datacenter. When you're replicating assets to a second region, you might also need a second Site Recovery vault.

Configuration and process server design: Site Recovery works with a local instance of a configuration and process server, which is bound to a single Site Recovery vault. This means that you might need to install a second instance of these servers in the source datacenter to facilitate replication.

Network bandwidth design: During replication and ongoing synchronization, you move binary data over the network, from the source datacenter to the Site Recovery vault in the target Azure datacenter. This process consumes bandwidth. Duplication of the workload to a second region doubles the amount of bandwidth consumed. When bandwidth is limited or a workload involves a large amount of configuration or data drift, it can interfere with the time required to complete the migration. More importantly, it can affect the experience of users or applications that still depend on the bandwidth of the source datacenter.

Data synchronization: Often the largest bandwidth drain comes from synchronization of the data platform. As defined in the reference architectures for multiregion web applications and multiregion n-tier applications, data synchronization is often required to keep the applications aligned. If this is the desired operational state of the application, it might be wise to complete a synchronization between the source data platform and each of the cloud platforms. You should do this before migrating the application and middle tier assets.

Azure-to-Azure disaster recovery: An alternative option can reduce complexity further. If timelines and data synchronization approach a two-step deployment, Azure-to-Azure disaster recovery might be an acceptable solution. In this scenario, you migrate the workload to the first Azure datacenter by using a single Site Recovery vault and configuration or process server design. After you test the workload, you can recover it to a second Azure datacenter from the migrated assets. This approach reduces the impact to resources in the source datacenter and takes advantage of faster transfer speeds and high bandwidth limits available between Azure datacenters.

> **NOTE**
> This approach can increase short-term migration costs through additional egress bandwidth charges.

Optimize and promote process changes

As you address global complexity during optimization and promotion, you might require duplicated efforts in each of the additional regions. When a single deployment is acceptable, you might still need to duplicate business testing and business change plans.

Suggested action during the optimize and promote process

Pretest optimization: Initial automation testing can identify potential optimization opportunities, as with any migration effort. In the case of global workloads, test the workload in each region independently. Minor configuration changes in the network or the target Azure datacenter can affect performance.

Business change plans: For any complex migration scenario, create a business change plan. This ensures clear communication regarding any changes to business processes, user experiences, and the timing of efforts required to integrate the changes. In the case of global migration efforts, the plan should include considerations for end users in each affected geography.

Business testing: In conjunction with the business change plan, business testing might be required in each

region. This ensures adequate performance and adherence to the modified networking routing patterns.

Promotion flights: Often promotion happens as a single activity, rerouting production traffic to the migrated workloads. In the case of global release efforts, you should deliver promotion in flights (or predefined collections of users). This allows the cloud strategy team and the cloud adoption team to better observe performance and improve support of users in each region. Promotion flights are often controlled at the networking level by changing the routing of specific IP ranges from the source workload assets to the newly migrated assets. After a specified collection of end users have been migrated, the next group can be rerouted.

Flight optimization: One of the benefits of promotion flights is that it allows for deeper observations and additional optimization of the deployed assets. After a brief period of production usage by the first flight, additional refinement of the migrated assets is suggested, when allowed by IT operation procedures.

Best practices when data requirements exceed network capacity during a migration effort

3/6/2021 • 5 minutes to read • Edit Online

In a cloud migration, you replicate and synchronize assets over the network between the existing datacenter and the cloud. It's not uncommon for the existing data size requirements of various workloads to exceed network capacity. In such a scenario, the process of migration can be radically slowed, or in some cases, stopped entirely. The following guidance expands the scope of the Azure migration guide to provide a solution that works around network limitations.

General scope expansion

Most of the effort required in this scope expansion occurs during the prerequisite planning, assessment, and migration phases of a migration.

Suggested prerequisites

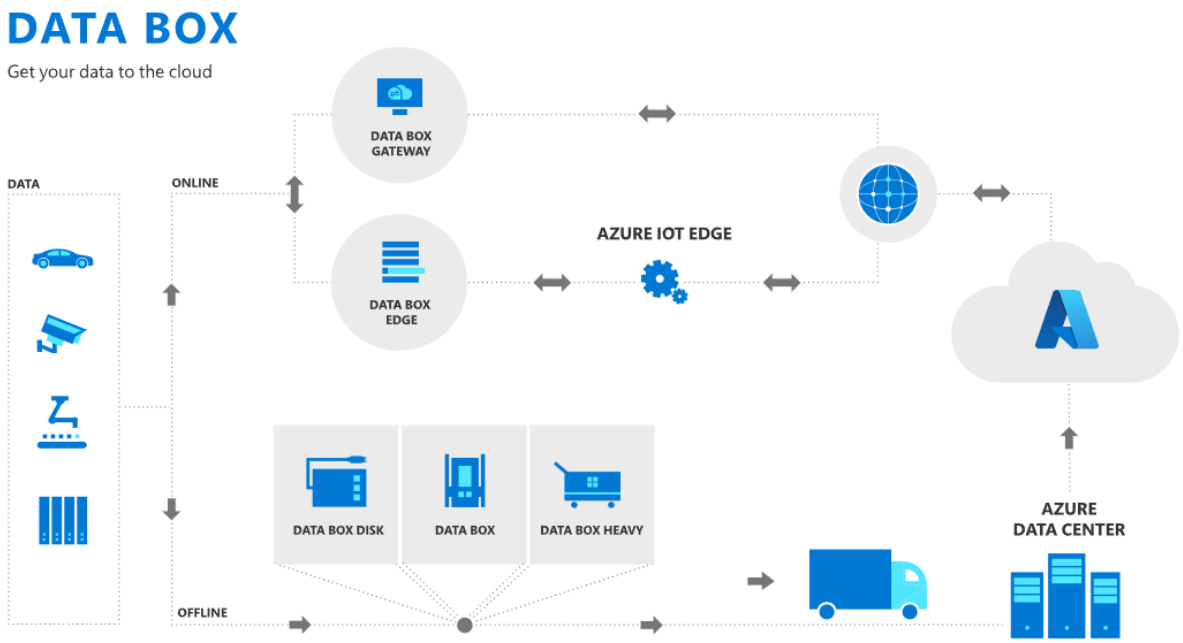
Validate network capacity risks: Digital estate rationalization is a highly recommended prerequisite, especially if there are concerns of overburdening the available network capacity. During digital estate rationalization, you collect an inventory of digital assets. That inventory should include existing storage requirements across the digital estate.

As outlined in Replication risks: physics of replication, you can use that inventory to estimate total migration data size, which can be compared to total available migration bandwidth. If that comparison doesn't align with the required time to business change, then this article can help accelerate migration velocity reducing the time required to migrate the datacenter.

Offline transfer of independent data stores: The following diagram shows examples of both online and offline data transfers with Azure Data Box. You can use these approaches to ship large volumes of data to the cloud, prior to workload migration. In an offline data transfer, you copy source data to Azure Data Box, which is then physically shipped to Microsoft for transfer into an Azure Storage account as a file or a blob. Prior to other migration efforts, you can use this process to ship data that isn't directly tied to a specific workload. Doing this reduces the amount of data that needs to be shipped over the network and supports completing a migration within network constraints.

You can use this approach to transfer data from HDFS, backups, archives, file servers, and applications. Existing technical guidance explains how to use this approach to transfer data from an HDFS store or from disks by using SMB, NFS, rest, or data copy service to Data Box.

There are also third-party partner solutions that use Azure Data Box for a migration. With these solutions, you move a large volume of data via an offline transfer, but you synchronize it later at a lower scale over the network.



Assess process changes

If the storage requirements of a workload (or workloads) exceed network capacity, then you can still use Azure Data Box in an offline data transfer.

Network transmission is the recommended approach unless the network is unavailable. The speed of transferring data over the network, even when bandwidth is constrained, is typically faster than physically shipping the data by using an offline transfer mechanism.

If connectivity to Azure is available, you should conduct an analysis before using Data Box, especially if migration of the workload is time sensitive. Data Box is only advisable when the time to transfer the necessary data exceeds the time to populate, ship, and restore it.

Suggested action during the assess process

Network capacity analysis: When workload-related data transfer requirements are at risk of exceeding network capacity, the cloud adoption team adds an additional analysis task to the assess process called network capacity analysis. During this analysis, a member of the team estimates the amount of available network capacity and required data transfer time. Note that this team member should have subject matter expertise regarding the local network and network connectivity.

Available capacity is compared to the storage requirements of all assets to be migrated during the current release. If the storage requirements exceed the available bandwidth, then assets supporting the workload are selected for offline transfer.

> **IMPORTANT**
> At the conclusion of the analysis, you might need to update the release plan to reflect the time required to ship, restore, and synchronize the assets to be transferred offline.

Drift analysis: Analyze each asset to be transferred offline for storage and configuration drift. *Storage drift* is the amount of change in the underlying storage over time. *Configuration drift* is change in the configuration of the asset over time. From the time the storage is copied to the time the asset is promoted to production, any drift might be lost. If that drift needs to be reflected in the migrated asset, you'll need to synchronize the local asset and the migrated asset. Flag this for consideration during migration execution.

Migration process changes

When you're using offline transfer mechanisms, replication processes aren't typically required, whereas synchronization processes might still be necessary. If an asset is being transferred offline, understanding the drift analysis results from the assess process will inform the tasks required during migration.

Suggested action during the migration process

Copy storage: You can use this approach to transfer data of HDFS, backups, archives, file servers, or applications. Existing technical guidance explains how to use this approach to transfer data from an HDFS store or from disks by using SMB, NFS, rest, or data copy service to Data Box.

There are also third-party partner solutions that use Azure Data Box for a migration. With these solutions, you move a large volume of data via an offline transfer, but you synchronize it later at a lower scale over the network.

Ship the device: After you copy the data, you can ship the device to Microsoft. After the data is received and imported, it's available in an Azure Storage account.

Restore the asset: Verify that the data is available in the storage account. If so, you can use the data as a blob or in Azure Files. If the data is a VHD/VHDX file, you can convert the file to managed disks. Those managed disks can then be used to instantiate a virtual machine, which creates a replica of the original on-premises asset.

Synchronization: If synchronization of drift is a requirement for a migrated asset, you can use one of the third-party partner solutions to synchronize the files until the asset is restored.

Optimize and promote process changes

Optimize activities aren't likely to be affected by this change in scope.

Secure and manage process changes

Secure and manage activities aren't likely to be affected by this change in scope.

Next steps

Return to the checklist to ensure that your migration method is fully aligned.

Migration best practices checklist

Best practices to set up networking for workloads migrated to Azure

5/26/2021 • 28 minutes to read • Edit Online

As you plan and design for migration, in addition to the migration itself, one of the most critical steps is the design and implementation of Azure networking. This article describes best practices for networking when you're migrating to infrastructure as a service (IaaS) and platform as a service (PaaS) implementations in Azure.

> **IMPORTANT**
> The best practices and opinions described in this article are based on the Azure platform and service features available at the time of writing. Features and capabilities change over time. Not all recommendations might be applicable for your deployment, so select those that work for you.

Design virtual networks

Azure provides virtual networks with these capabilities:

- Azure resources communicate privately, directly, and securely with each other over virtual networks.
- You can configure endpoint connections on virtual networks for VMs and services that require internet communication.
- A virtual network is a logical isolation of the Azure cloud that's dedicated to your subscription.
- You can implement multiple virtual networks within each Azure subscription and Azure region.
- Each virtual network is isolated from other virtual networks.
- Virtual networks can contain private and public IP addresses defined in RFC 1918, expressed in classless inter-domain routing (CIDR) notation. Public IP addresses specified in a virtual network's address space aren't directly accessible from the internet.
- Virtual networks can connect to each other by using virtual network peering. Connected virtual networks can be in the same region or different regions; resources in one virtual network can connect to resources in other virtual networks.
- By default, Azure routes traffic between subnets within a virtual network, connected virtual networks, on-premises networks, and the internet.

When planning your virtual network topology, you should consider how to arrange IP address spaces, how to implement a hub and spoke network topology, how to segment virtual networks into subnets, setting up DNS, and implementing Azure Availability Zones.

Best practice: Plan IP addressing

When you create virtual networks as part of your migration, it's important to plan out your virtual network IP address space.

You should assign an address space that isn't larger than a CIDR range of `/16` for each virtual network. Virtual networks allow for the use of 65,536 IP addresses. Assigning a smaller prefix than `/16`, such as a `/15`, which has 131,072 addresses, will result in the excess IP addresses becoming unusable elsewhere. It's important not to waste IP addresses, even if they're in the private ranges defined by RFC 1918.

Other tips for planning are:

- The virtual network address space shouldn't overlap with on-premises network ranges.
- Overlapping addresses can cause networks that can't be connected, and routing that doesn't work properly.
- If networks overlap, you'll need to redesign the network.
- If you absolutely can't redesign the network, network address translation (NAT) can help but should be avoided or limited as much as possible.

Learn more:

- Read the Azure Virtual Network overview.
- Review the Azure Virtual Network FAQ.
- Learn about Azure networking limits.

Best practice: Implement a hub and spoke network topology

A hub and spoke network topology isolates workloads while sharing services, such as identity and security. The hub is an Azure virtual network that acts as a central point of connectivity. The spokes are virtual networks that connect to the hub virtual network by using peering. Shared services are deployed in the hub, while individual workloads are deployed as spokes.

Consider the following:

- Implementing a hub and spoke topology in Azure centralizes common services, such as connections to on-premises networks, firewalls, and isolation between virtual networks. The hub virtual network provides a central point of connectivity to on-premises networks, and a place to host services used by workloads hosted in spoke virtual networks.
- A hub and spoke configuration is typically used by larger enterprises. Smaller networks might consider a simpler design to save on costs and complexity.
- You can use spoke virtual networks to isolate workloads, with each spoke managed separately from other spokes. Each workload can include multiple tiers, and multiple subnets that are connected with Azure load balancers.
- You can implement hub and spoke virtual networks in different resource groups, and even in different subscriptions. When you peer virtual networks in different subscriptions, the subscriptions can be associated to the same, or different, Azure Active Directory (Azure AD) tenants. This allows for decentralized management of each workload, while sharing services maintained in the hub network.

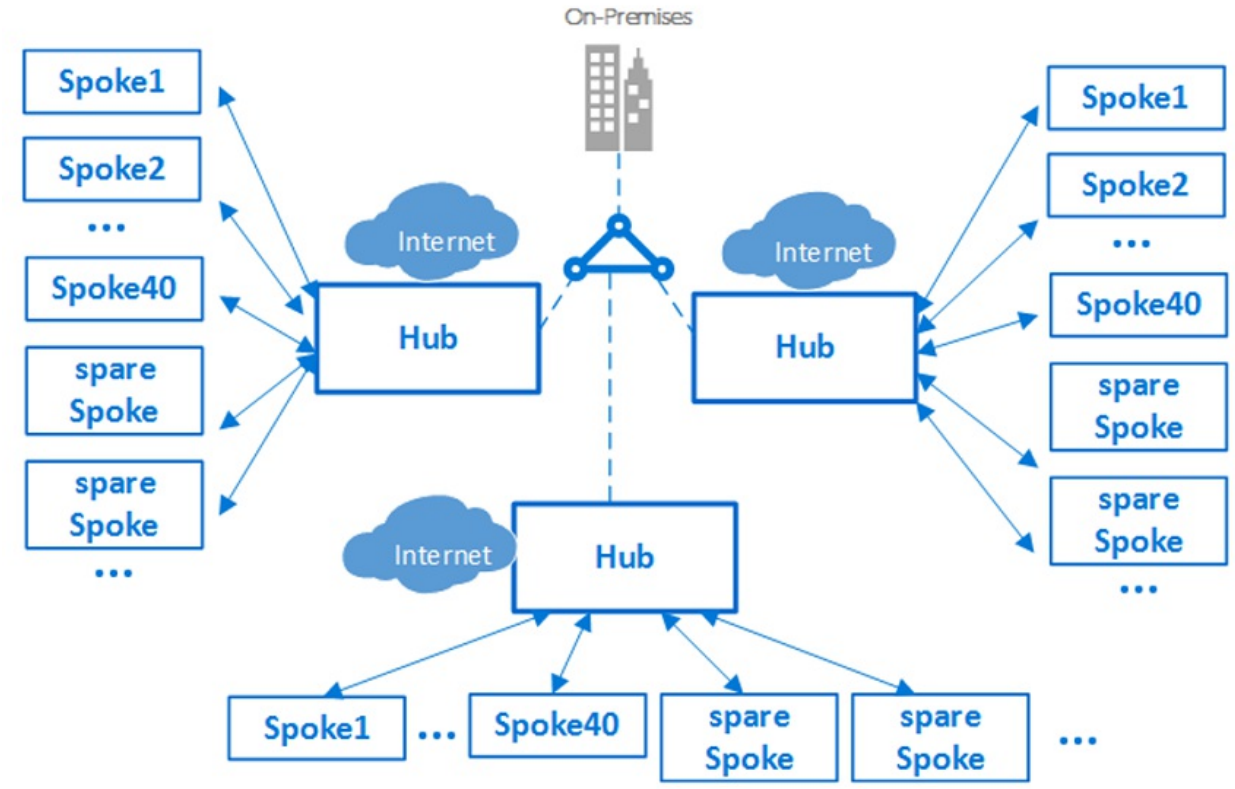


Figure 1: Hub and spoke topology.

Learn more:

- Read about a hub and spoke topology.
- Get network recommendations for running Windows VMs and Linux VMs in Azure.
- Learn about virtual network peering.

Best practice: Design subnets

To provide isolation within a virtual network, you segment it into one or more subnets, and allocate a portion of the virtual network's address space to each subnet.

- You can create multiple subnets within each virtual network.
- By default, Azure routes network traffic between all subnets in a virtual network.
- Your subnet decisions are based on your technical and organizational requirements.
- You create subnets by using CIDR notation.

When you're deciding on network range for subnets, be aware that Azure retains five IP addresses from each subnet that can't be used. For example, if you create the smallest available subnet of `/29` (with eight IP addresses), Azure will retain five addresses. In this case, you only have three usable addresses that can be assigned to hosts on the subnet. For most cases, use `/28` as the smallest subnet.

Example:

The table shows an example of a virtual network with an address space of `10.245.16.0/20` segmented into subnets, for a planned migration.

SUBNET	CIDR	ADDRESSES	USAGE
`DEV-FE-EUS2`	`10.245.16.0/22`	1019	Front-end or web-tier VMs
`DEV-APP-EUS2`	`10.245.20.0/22`	1019	Application-tier VMs

SUBNET	CIDR	ADDRESSES	USAGE
`DEV-DB-EUS2`	`10.245.24.0/23`	507	Database VMs

Learn more:

- Learn about designing subnets.
- Learn how Contoso, a fictional company, prepared their networking infrastructure for migration.

Best practice: Set up a DNS server

Azure adds a DNS server by default when you deploy a virtual network. This allows you to rapidly build virtual networks and deploy resources. But this DNS server only provides services to the resources on that virtual network. If you want to connect multiple virtual networks together, or connect to an on-premises server from virtual networks, you need additional name resolution capabilities. For example, you might need Active Directory to resolve DNS names between virtual networks. To do this, you deploy your own custom DNS server in Azure.

- DNS servers in a virtual network can forward DNS queries to the recursive resolvers in Azure. This enables you to resolve host names within that virtual network. For example, a domain controller that runs in Azure can respond to DNS queries for its own domains, and forward all other queries to Azure.
- DNS forwarding allows VMs to see both your on-premises resources (via the domain controller) and Azure-provided host names (using the forwarder). You can access the recursive resolvers in Azure by using the virtual IP address `168.63.129.16`.
- DNS forwarding also enables DNS resolution between virtual networks, and allows on-premises machines to resolve host names provided by Azure.
    - To resolve a VM host name, the DNS server VM must reside in the same virtual network, and be configured to forward host name queries to Azure.
    - Because the DNS suffix is different in each virtual network, you can use conditional forwarding rules to send DNS queries to the correct virtual network for resolution.
- When you use your own DNS servers, you can specify multiple DNS servers for each virtual network. You can also specify multiple DNS servers per network interface (for Azure Resource Manager), or per cloud service (for the classic deployment model).
- DNS servers specified for a network interface or cloud service take precedence over DNS servers specified for the virtual network.
- In Azure Resource Manager, you can specify DNS servers for a virtual network and a network interface, but the best practice is to use the setting only on virtual networks.

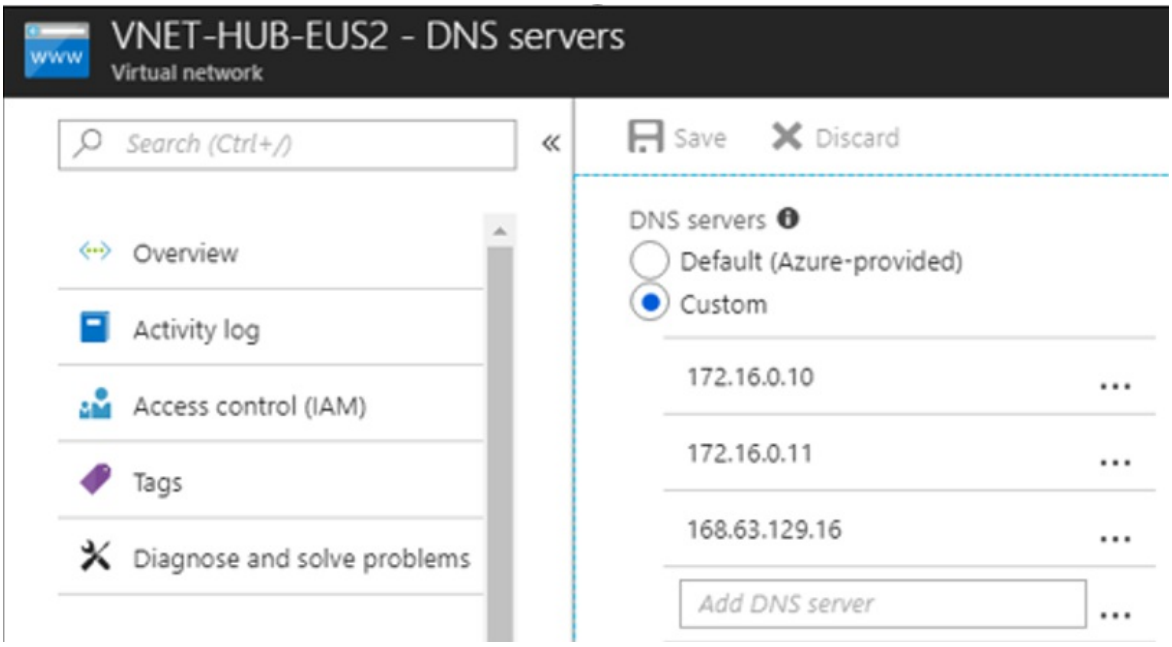


Figure 2: DNS servers for a virtual network.

Learn more:

- Learn about name resolution when you use your own DNS server.
- Learn about DNS naming rules and restrictions.

Best practice: Set up Availability Zones

Availability Zones increase high-availability to protect your applications and data from datacenter failures. Availability Zones are unique physical locations within an Azure region. Each zone is made up of one or more datacenters equipped with independent power, cooling, and networking. To ensure resiliency, there's a minimum of three separate zones in all enabled regions. The physical separation of Availability Zones within a region protects applications and data from datacenter failures.

Here are a few additional points to be aware of as you set up Availability Zones:

- Zone-redundant services replicate your applications and data across Availability Zones to protect from single points of failure.
- With Availability Zones, Azure offers an SLA of 99.99 percent for VM uptime.

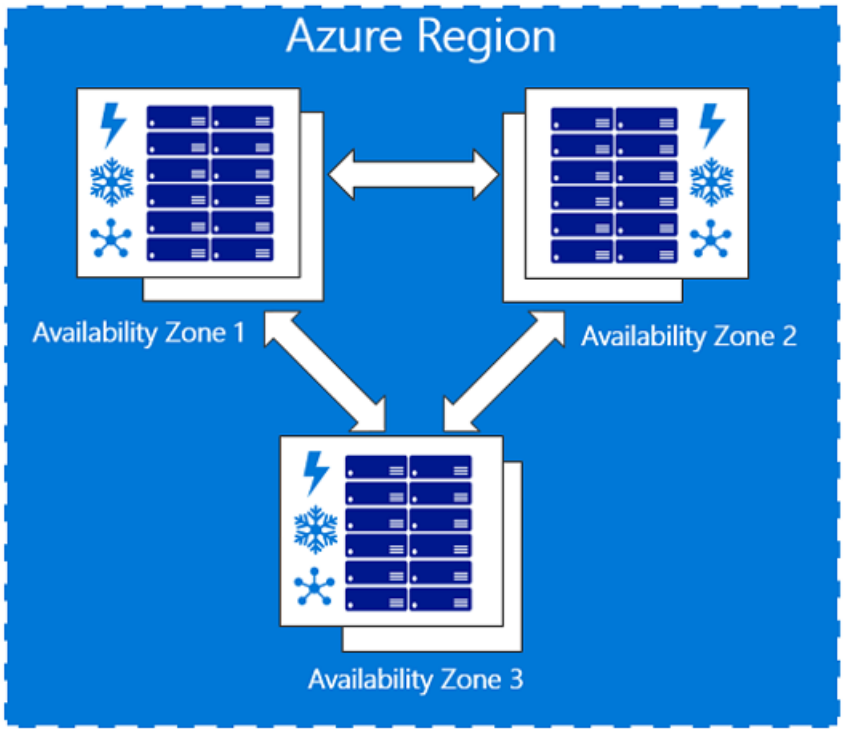


Figure 3: Availability Zones.

- You can plan and build high-availability into your migration architecture by colocating compute, storage, networking, and data resources within a zone, and replicating them in other zones. Azure services that support Availability Zones fall into two categories:
  - **Zonal services:** You associate a resource with a specific zone, such as VMs, managed disks, or IP addresses.
  - **Zone-redundant services:** The resource replicates automatically across zones, such as zone-redundant storage or Azure SQL Database.
- To provide zonal fault tolerance, you can deploy a standard Azure Load Balancer instance with internet-facing workloads or application tiers.

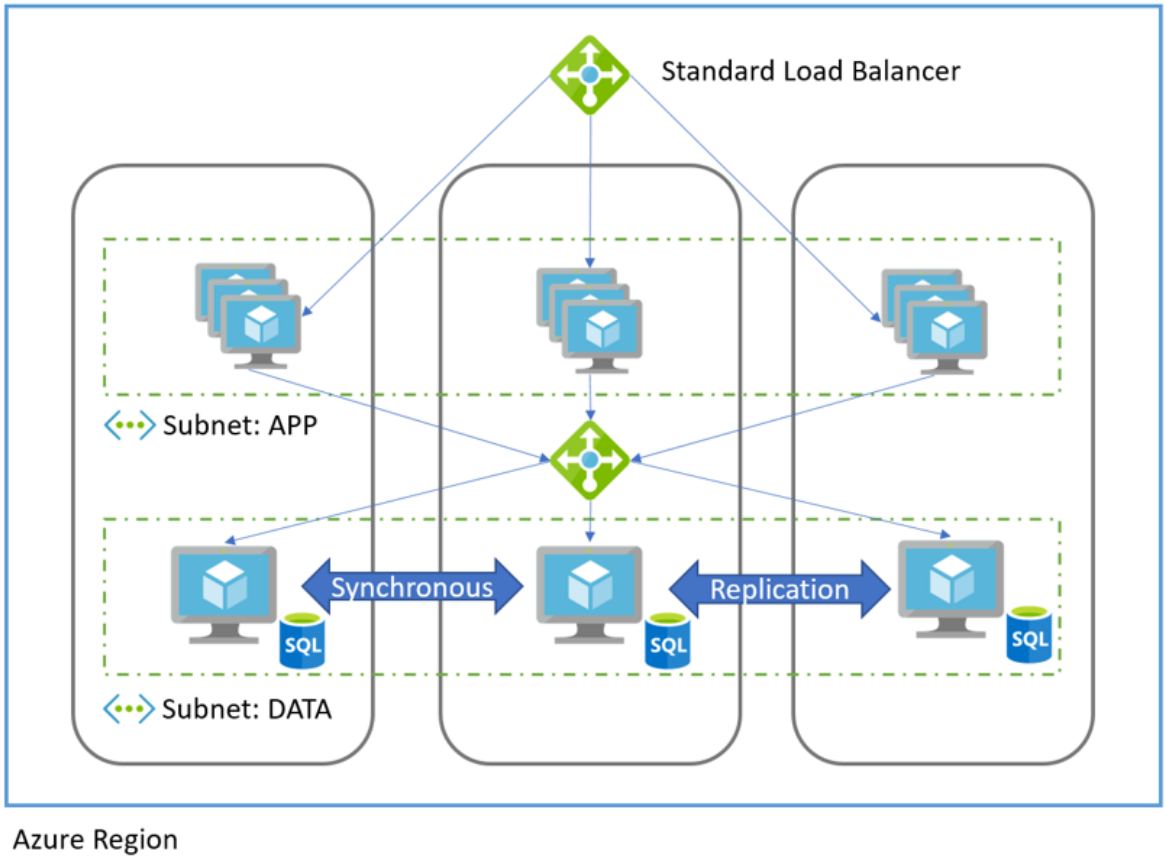


Figure 4: Load balancer.

Learn more:

- Read the Availability Zones overview.

Design hybrid cloud networking

For a successful migration, it's critical to connect on-premises corporate networks to Azure. This creates an always-on connection known as a hybrid-cloud network, where services are provided from the Azure cloud to corporate users. There are two options for creating this type of network:

- **Site-to-Site VPN:** You establish a Site-to-Site VPN connection between your compatible on-premises VPN device and an Azure VPN gateway that's deployed in a virtual network. Any authorized, on-premises resource can access virtual networks. Site-to-Site communications are sent through an encrypted tunnel over the internet.
- **Azure ExpressRoute:** You establish an Azure ExpressRoute connection between your on-premises network and Azure, through an ExpressRoute partner. This connection is private, and traffic doesn't go over the internet.

Learn more:

- Learn more about hybrid-cloud networking.

Best practice: Implement a highly available Site-to-Site VPN

To implement a Site-to-Site VPN, you set up a VPN gateway in Azure.

- A VPN gateway is a specific type of virtual network gateway. It sends encrypted traffic between an Azure virtual network and an on-premises location over the public internet.
- A VPN gateway can also send encrypted traffic between virtual networks in Azure over the Microsoft network.
- Each virtual network can have only one VPN gateway.

- You can create multiple connections to the same VPN gateway. When you create multiple connections, all VPN tunnels share the available gateway bandwidth.

Every Azure VPN gateway consists of two instances in an active-standby configuration:

- For planned maintenance or unplanned disruption to the active instance, failover occurs and the standby instance takes over automatically. This instance resumes the Site-to-Site or network-to-network connection.
- The switchover causes a brief interruption.
- For planned maintenance, connectivity should be restored within 10 to 15 seconds.
- For unplanned issues, the connection recovery takes longer, up to 1.5 minutes in the worst case.
- Point-to-Site VPN client connections to the gateway are disconnected, and users need to reconnect from client machines.

When setting up a Site-to-Site VPN:

- You need a virtual network whose address range doesn't overlap with the on-premises network to which the VPN will connect.
- You create a gateway subnet in the network.
- You create a VPN gateway, specify the gateway type (VPN), and whether the gateway is policy-based or route-based. A route-based VPN is considered more capable and future-proof.
- You create a local network gateway on-premises, and configure your on-premises VPN device.
- You create a failover Site-to-Site VPN connection between the virtual network gateway and the on-premises device. Using route-based VPN allows for either active-passive or active-active connections to Azure. The route-based option also supports both Site-to-Site (from any computer) and Point-to-Site (from a single computer) connections, concurrently.
- You specify the gateway SKU that you want to use. This depends on your workload requirements, throughput, features, and SLAs.
- Border Gateway Protocol (BGP) is an optional feature. You can use it with Azure ExpressRoute and route-based VPN gateways to propagate your on-premises BGP routes to your virtual networks.

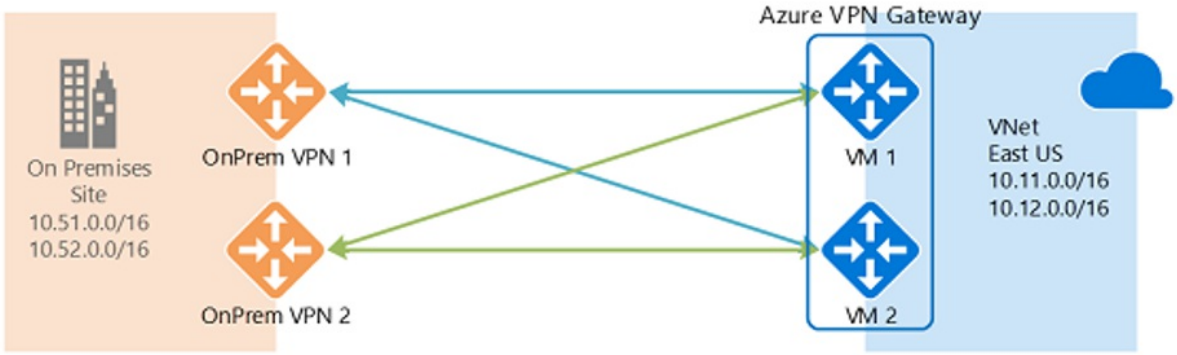


Figure 5: Site-to-Site VPN.

Learn more:

- Review compatible on-premises VPN devices.
- Read the Azure VPN gateways overview.
- Learn about highly available VPN connections.
- Learn about planning and designing a VPN gateway.
- Review VPN gateway settings.
- Review gateway SKUs.
- Read about setting up BGP with Azure VPN gateways.

Best practice: Configure a gateway for VPN gateways

When you create a VPN gateway in Azure, you must use a special subnet named `GatewaySubnet`. When you create this subnet, note these best practices:

- `GatewaySubnet` can have a maximum prefix length of 29 (for example, `10.119.255.248/29`). The current recommendation is that you use a prefix length of 27 (for example, `10.119.255.224/27`).
- When you define the address space of the gateway subnet, use the very last part of the virtual network address space.
- When you're using the Azure gateway subnet, never deploy any VMs or other devices, such as Azure Application Gateway, to the gateway subnet.
- Don't assign a network security group (NSG) to this subnet. It will cause the gateway to stop functioning.

Best practice: Implement Azure Virtual WAN for branch offices

For multiple VPN connections, Azure Virtual WAN is a networking service that provides optimized and automated, branch-to-branch connectivity through Azure.

- Virtual WAN allows you to connect and configure branch devices to communicate with Azure. You can do this manually, or by using preferred provider devices through an Azure Virtual WAN partner.
- Using preferred provider devices allows for simple use, connectivity, and configuration management.
- The Azure Virtual WAN built-in dashboard provides instant troubleshooting insights that save time, and provide an easy way to track large-scale, site-to-site connectivity.

Learn more: Learn about Azure Virtual WAN.

Best practice: Implement ExpressRoute for mission-critical connections

The Azure ExpressRoute service extends your on-premises infrastructure into the Microsoft cloud, by creating private connections between the virtual Azure datacenter and on-premises networks. Here are a few implementation details:

- ExpressRoute connections can be over an any-to-any (IP VPN) network, a point-to-point Ethernet network, or through a connectivity provider. They don't go over the public internet.
- ExpressRoute connections offer higher security, reliability, and higher speeds (up to 10 Gbps), along with consistent latency.
- ExpressRoute is useful for virtual datacenters, as customers can get the benefits of compliance rules associated with private connections.
- With ExpressRoute Direct, you can connect directly to Microsoft routers at 100 Gbps, for larger bandwidth needs.
- ExpressRoute uses BGP to exchange routes between on-premises networks, Azure instances, and Microsoft public addresses.

Deploying ExpressRoute connections usually involves engaging with an ExpressRoute service provider. For a rapid start, it's common to initially use a Site-to-Site VPN to establish connectivity between the virtual datacenter and on-premises resources. Then you migrate to an ExpressRoute connection when a physical interconnection with your service provider is established.

Learn more:

- Read an overview of ExpressRoute.
- Learn about ExpressRoute Direct.

Best practice: Optimize ExpressRoute routing with BGP communities

When you have multiple ExpressRoute circuits, you have more than one path to connect to Microsoft. As a result, suboptimal routing can happen and your traffic might take a longer path to reach Microsoft, and Microsoft to your network. The longer the network path, the higher the latency. Latency directly affects application

performance and the user experience.

Example:

Let's review an example:

- You have two offices in the US, one in Los Angeles and one in New York City.
- Your offices are connected on a WAN, which can be either your own backbone network or your service provider's IP VPN.
- You have two ExpressRoute circuits, one in `West US` and one in `East US`, which are also connected on the WAN. Obviously, you have two paths to connect to the Microsoft network.

Problem:

Now imagine that you have an Azure deployment (for example, Azure App Service) in both `West US` and `East US`.

- You want users in each office to access their nearest Azure services for an optimal experience.
- Thus, you want to connect users in Los Angeles to Azure `West US`, and users in New York to Azure `East US`.
- This works for east coast users, but not for those on the west coast. The problem is:
  - On each ExpressRoute circuit, we advertise both prefixes in Azure: `East US` (`23.100.0.0/16`) and Azure `West US` (`13.100.0.0/16`).
  - Without knowing which prefix is from which region, prefixes aren't treated differently.
  - Your WAN network can assume that both prefixes are closer to `East US` than `West US`, and thus route users from both offices to the ExpressRoute circuit in `East US`. This provides a worse experience for users in the Los Angeles office.

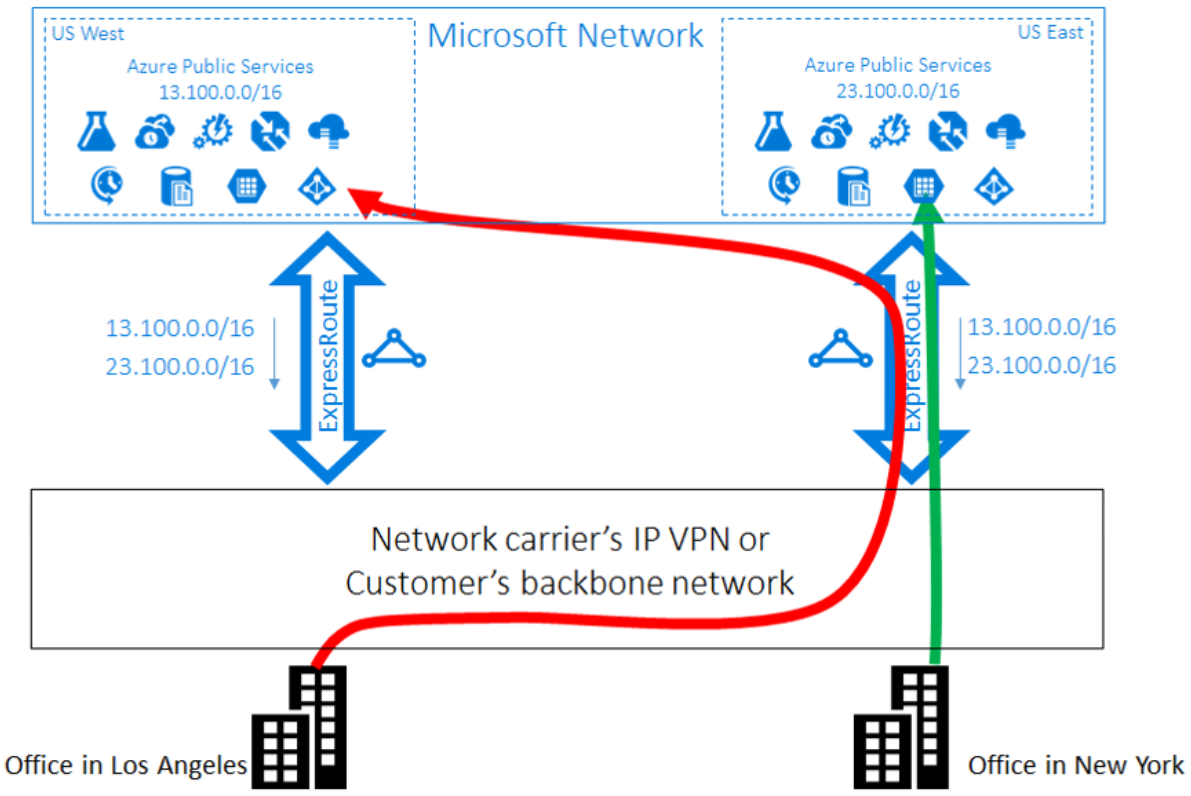


Figure 6: BGP communities unoptimized connection.

Solution:

To optimize routing for both offices, you need to know which prefix is from Azure `West US` and which prefix is from Azure `East US`. You can encode this information by using BGP community values.

- You assign a unique BGP community value to each Azure region. For example, 12076:51004 for `East US`; 12076:51006 for `West US`.
- Now that it's clear which prefix belongs to which Azure region, you can configure a preferred ExpressRoute circuit.
- Because you're using BGP to exchange routing information, you can use BGP's local preference to influence routing.
- In our example, you assign a higher local preference value to `13.100.0.0/16` in `West US` than in `East US`. Similarly, you assign a higher local preference value to `23.100.0.0/16` in `East US` than in `West US`.
- This configuration ensures that when both paths to Microsoft are available, users in Los Angeles connect to the `West US` region by using the west circuit, and users in New York connect to the `East US` region by using the east circuit.

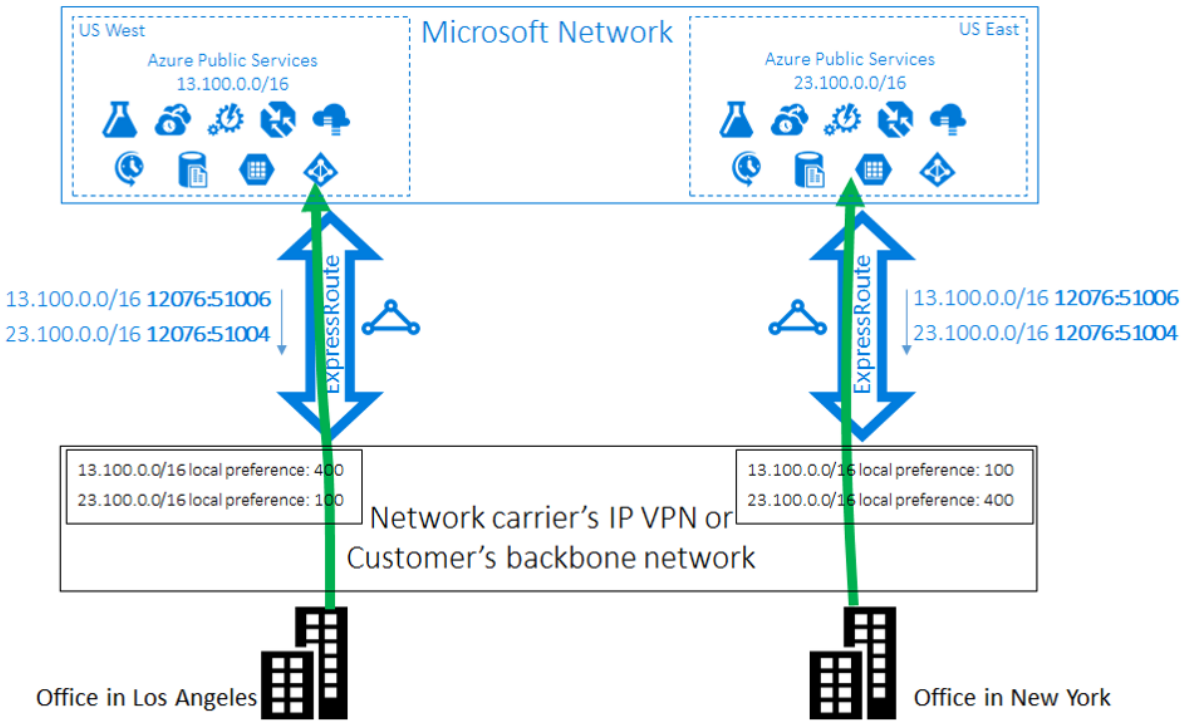


Figure 7: BGP communities optimized connection.

Learn more:

- Learn about optimizing routing.

Secure virtual networks

The responsibility for securing virtual networks is shared between Microsoft and you. Microsoft provides many networking features, as well as services that help keep resources secure. When you're designing security for virtual networks, it's best to implement a perimeter network, use filtering and security groups, secure access to resources and IP addresses, and implement attack protection.

Learn more:

- Read an overview of best practices for network security.
- Learn how to design for secure networks.

Best practice: Implement an Azure perimeter network

Although Microsoft invests heavily in protecting the cloud infrastructure, you must also protect your cloud services and resource groups. A multilayered approach to security provides the best defense. Putting a

perimeter network in place is an important part of that defense strategy.

- A perimeter network protects internal network resources from an untrusted network.
- It's the outermost layer that's exposed to the internet. It generally sits between the internet and the enterprise infrastructure, usually with some form of protection on both sides.
- In a typical enterprise network topology, the core infrastructure is heavily fortified at the perimeters, with multiple layers of security devices. The boundary of each layer consists of devices and policy enforcement points.
- Each layer can include a combination of the network security solutions, such as firewalls, denial of service (DoS) prevention, intrusion detection/intrusion protection systems (IDS/IPS), and VPN devices.
- Policy enforcement on the perimeter network can use firewall policies, access control lists (ACLs), or specific routing.
- As incoming traffic arrives from the internet, it's intercepted and handled by a combination of defense solutions. The solutions block attacks and harmful traffic, while allowing legitimate requests into the network.
- Incoming traffic can route directly to resources in the perimeter network. The perimeter network resource can then communicate with other resources deeper in the network, moving traffic forward into the network after validation.

Here's an example of a single subnet perimeter network in a corporate network, with two security boundaries.

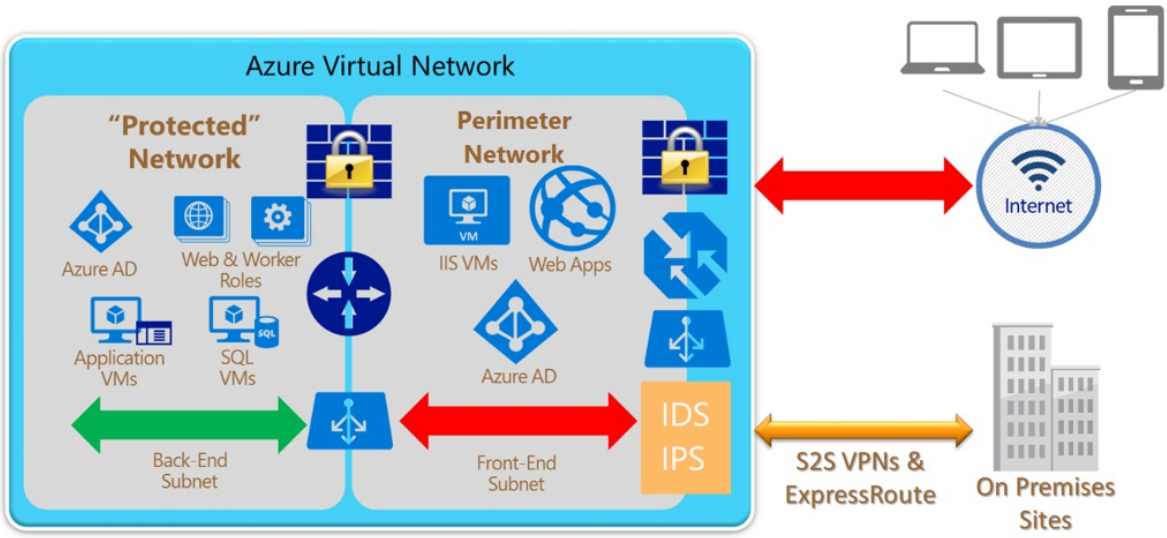


Figure 8: Perimeter network deployment.

Learn more:

- Learn how to deploy a perimeter network between Azure and your on-premises datacenter.

Best practice: Filter virtual network traffic with NSGs

Network security groups (NSGs) contain multiple inbound and outbound security rules that filter traffic going to and from resources. Filtering can be by source and destination IP address, port, and protocol.

- NSGs contain security rules that allow or deny inbound network traffic to (or outbound network traffic from) several types of Azure resources. For each rule, you can specify source and destination, port, and protocol.
- NSG rules are evaluated by priority by using five-tuple information (source, source port, destination, destination port, and protocol), to allow or deny the traffic.
- A flow record is created for existing connections. Communication is allowed or denied based on the connection state of the flow record.
- A flow record allows an NSG to be stateful. For example, if you specify an outbound security rule to any address over port 80, you don't need an inbound security rule to respond to the outbound traffic. You only

need to specify an inbound security rule if communication is initiated externally.

- The opposite is also true. If inbound traffic is allowed over a port, you don't need to specify an outbound security rule to respond to traffic over the port.
- Existing connections aren't interrupted when you remove a security rule that enabled the flow. Traffic flows are interrupted when connections are stopped, and no traffic is flowing in either direction, for at least a few minutes.
- When creating NSGs, create as few as possible, but as many as necessary.

Best practice: Secure north/south and east/west traffic

To secure virtual networks, consider attack vectors. Note the following points:

- Using only subnet NSGs simplifies your environment, but only secures traffic into your subnet. This is known as north/south traffic.
- Traffic between VMs on the same subnet is known as east/west traffic.
- It's important to use both forms of protection, so that if a hacker gains access from the outside, they'll be stopped when trying to attack machines located in the same subnet.

Use service tags on NSGs

A service tag represents a group of IP address prefixes. Using a service tag helps minimize complexity when you create NSG rules.

- You can use service tags instead of specific IP addresses when you create rules.
- Microsoft manages the address prefixes associated with a service tag, and automatically updates the service tag as addresses change.
- You can't create your own service tag, or specify which IP addresses are included within a tag.

Service tags take the manual work out of assigning a rule to groups of Azure services. For example, if you want to allow a subnet containing web servers to access to Azure SQL Database, you can create an outbound rule to port 1433, and use the **Sql** service tag.

- This **Sql** tag denotes the address prefixes of the Azure SQL Database and Azure SQL Data Warehouse services.
- If you specify **Sql** as the value, traffic is allowed or denied to SQL.
- If you only want to allow access to **Sql** in a specific region, you can specify that region. For example, if you want to allow access only to Azure SQL Database in the East US region, you can specify **Sql.EastUS** for the service tag.
- The tag represents the service, but not specific instances of the service. For example, the tag represents the Azure SQL Database service, but doesn't represent a particular SQL Database or server.
- All address prefixes represented by this tag are also represented by the **Internet** tag.

Learn more:

- Read about network security groups (NSGs).
- Review the service tags available for NSGs.

Best practice: Use application security groups

Application security groups enable you to configure network security as a natural extension of an application structure.

- You can group VMs and define network security policies based on application security groups.
- Application security groups enable you to reuse your security policy at scale, without manual maintenance of explicit IP addresses.
- Application security groups handle the complexity of explicit IP addresses and multiple rule sets, allowing

you to focus on your business logic.

Example:

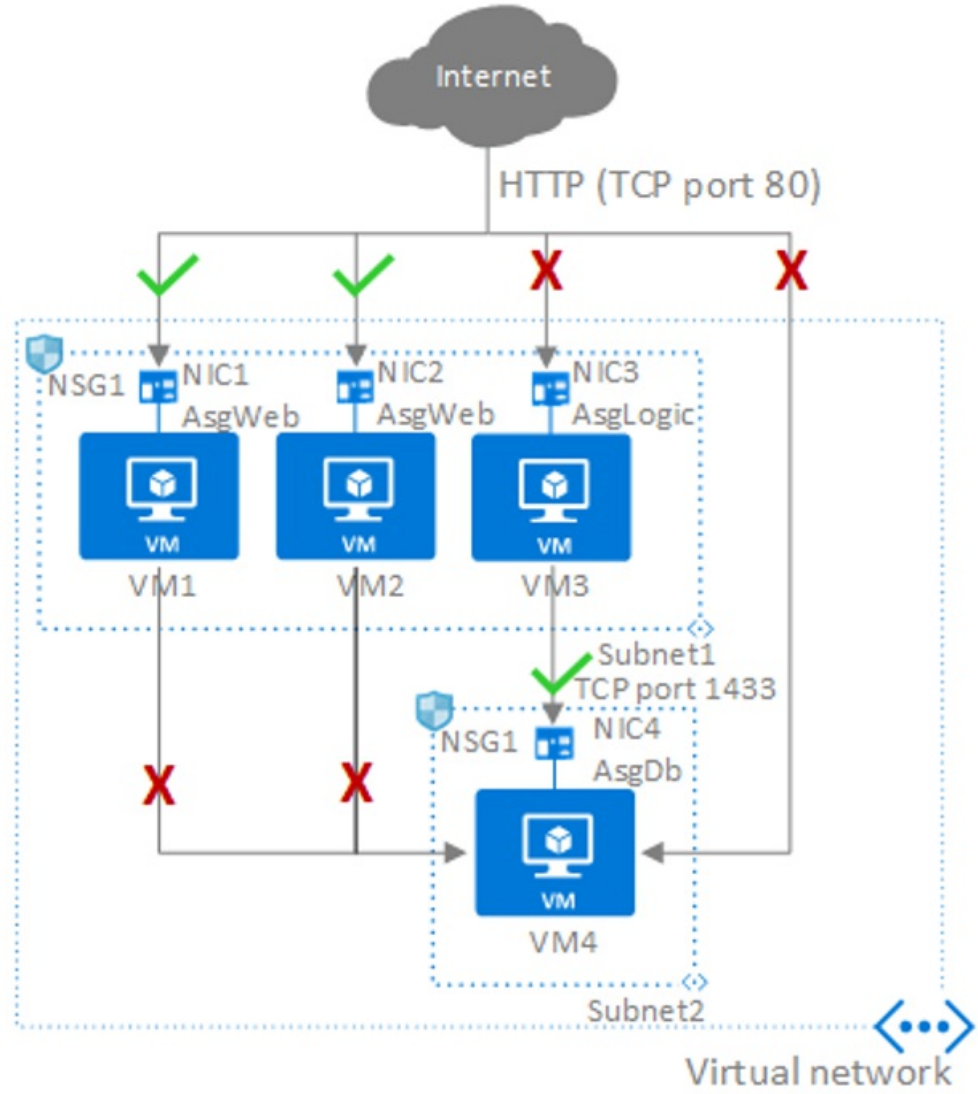


Figure 9: Application security group example.

NETWORK INTERFACE	APPLICATION SECURITY GROUP
`NIC1`	`AsgWeb`
`NIC2`	`AsgWeb`
`NIC3`	`AsgLogic`
`NIC4`	`AsgDb`

In our example, each network interface belongs to only one application security group, but in fact an interface can belong to multiple groups, in accordance with Azure limits. None of the network interfaces have an associated NSG. `NSG1` is associated with both subnets, and contains the following rules:

RULE NAME	PURPOSE	DETAILS

RULE NAME	PURPOSE	DETAILS
`Allow-HTTP-Inbound-Internet`	Allow traffic from the internet to the web servers. Inbound traffic from the internet is denied by the `DenyAllInbound` default security rule, so no additional rule is needed for the `AsgLogic` or `AsgDb` application security groups.	Priority: `100` Source: `internet` Source port: `*` Destination: `AsgWeb` Destination port: `80` Protocol: `TCP` Access: `Allow`
`Deny-Database-All`	`AllowVNetInBound` default security rule allows all communication between resources in the same virtual network. This rule is needed to deny traffic from all resources.	Priority: `120` Source: `*` Source port: `*` Destination: `AsgDb` Destination port: `1433` Protocol: `All` Access: `Deny`
`Allow-Database-BusinessLogic`	Allow traffic from the `AsgLogic` application security group to the `AsgDb` application security group. The priority for this rule is higher than the `Deny-Database-All` rule, so this rule is processed first. Therefore, traffic from the `AsgLogic` application security group is allowed, and all other traffic is blocked.	Priority: `110` Source: `AsgLogic` Source port: `*` Destination: `AsgDb` Destination port: `1433` Protocol: `TCP` Access: `Allow`

The rules that specify an application security group as the source or destination are only applied to the network interfaces that are members of the application security group. If the network interface isn't a member of an application security group, the rule is not applied to the network interface, even though the network security group is associated with the subnet.

Learn more:

- Learn about application security groups.

Best practice: Secure access to PaaS by using virtual network service endpoints

Virtual network service endpoints extend your virtual network private address space and identity to Azure services over a direct connection.

- Endpoints allow you to secure critical Azure service resources to your virtual networks only. Traffic from your virtual network to the Azure service always remains on the Azure backbone network.

- Virtual network private address space can be overlapping, and thus can't be used to uniquely identify traffic originating from a virtual network.
- After you enable service endpoints in your virtual network, you can secure Azure service resources by adding a virtual network rule to the service resources. This provides improved security by fully removing public internet access to resources, and allowing traffic only from your virtual network.

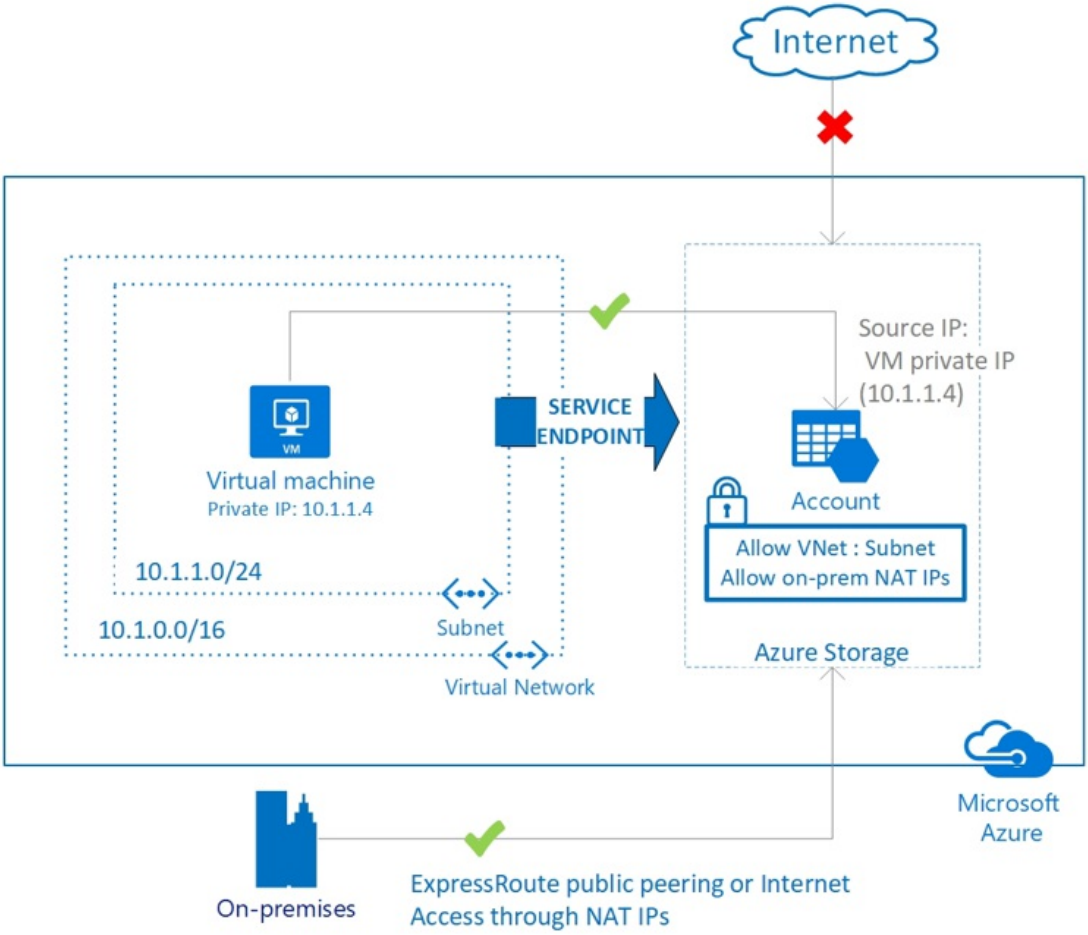


Figure 10: Service endpoints.

Learn more:

- Learn about virtual network service endpoints.

Best practice: Control public IP addresses

Public IP addresses in Azure can be associated with VMs, load balancers, application gateways, and VPN gateways.

- Public IP addresses allow internet resources to communicate inbound to Azure resources, and Azure resources to communicate outbound to the internet.
- Public IP addresses are created using either a Basic SKU or Standard SKU. Standard SKUs can be assigned to any service, but are usually configured on VMs, load balancers, and application gateways.
- A Basic public IP address doesn't have an NSG automatically configured. You need to configure your own, and assign rules to control access. Standard SKU IP addresses have an NSG, and rules assigned by default.
- As a best practice, VMs shouldn't be configured with a public IP address.
  - If you need a port opened, it should only be for web services, such as port 80 or 443.
  - Standard remote management ports, such as SSH (22) and RDP (3389), along with all other ports, should be set to deny by using NSGs.
- A better practice is to put VMs behind Azure Load Balancer or Azure Application Gateway. Then, if you need

access to remote management ports, you can use just-in-time VM access in Azure Security Center.

Learn more:

- Public IP addresses in Azure
- Manage virtual machine access by using just-in-time

Take advantage of Azure security features for networking

Azure has platform-level security features, including Azure Firewall, Azure Web Application Firewall (WAF), and Azure Network Watcher.

Best practice: Deploy Azure Firewall

Azure Firewall is a managed, cloud-based, network security service that helps protect your virtual network resources. It's a fully stateful, managed firewall, with built-in high availability and unrestricted cloud scalability.

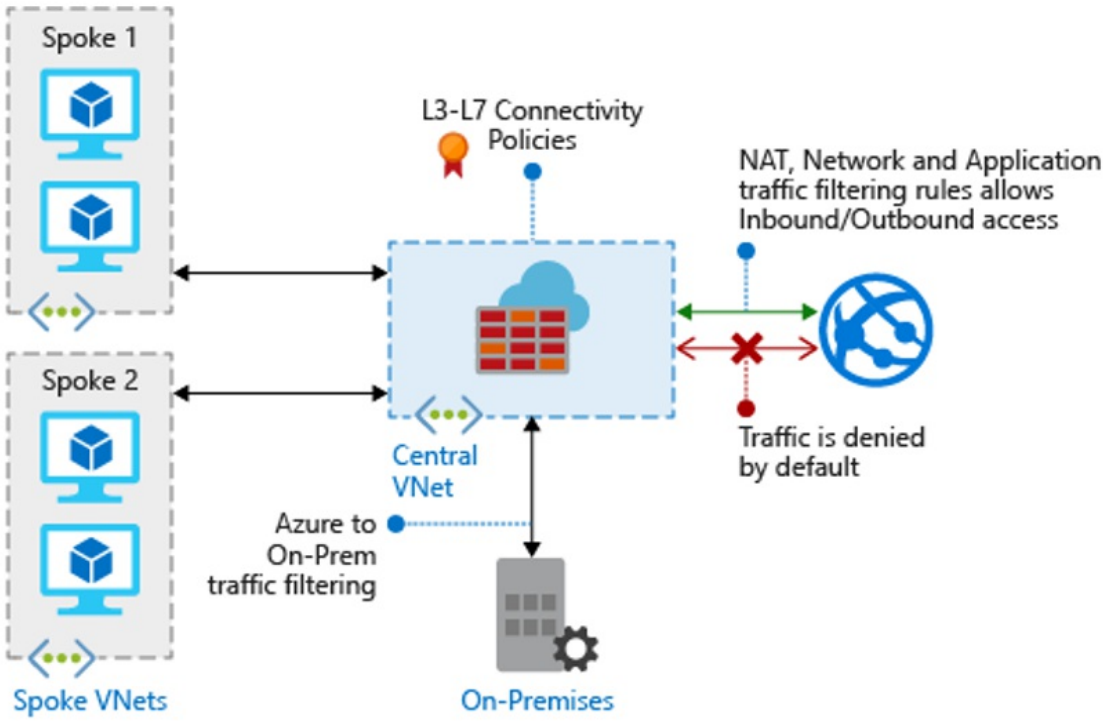


Figure 11: Azure Firewall.

Here are a few points to be aware of if you deploy the service:

- Azure Firewall can centrally create, enforce, and log application and network connectivity policies across subscriptions and virtual networks.
- Azure Firewall uses a static, public IP address for your virtual network resources. This allows outside firewalls to identify traffic originating from your virtual network.
- Azure Firewall is fully integrated with Azure Monitor for logging and analytics.
- When you're creating Azure Firewall rules, it's best to use the FQDN tags.
  - An FQDN tag represents a group of FQDNs associated with well-known Microsoft services.
  - You can use an FQDN tag to allow the required outbound network traffic through the firewall.
- For example, to manually allow Windows Update network traffic through your firewall, you would need to create multiple application rules. By using FQDN tags, you create an application rule, and include the Windows Update tag. With this rule in place, network traffic to Microsoft Windows Update endpoints can flow through your firewall.

Learn more:

- Read the Azure Firewall overview.
- Learn about FQDN tags in Azure Firewall.

Best practice: Deploy Web Application Firewall

Web applications are increasingly targets of malicious attacks that exploit commonly known vulnerabilities. Exploits include SQL injection attacks and cross-site scripting attacks. Preventing such attacks in application code can be challenging, and can require rigorous maintenance, patching, and monitoring at multiple layers of the application topology.

Web Application Firewall (WAF), a feature of Azure Application Gateway, helps make security management much simpler, and helps application administrators guard against threats or intrusions. You can react to security threats faster, by patching known vulnerabilities at a central location, instead of securing individual web applications. You can easily convert existing application gateways to Application Gateway that is enabled for Web Application Firewall.

Here are some additional notes about WAF:

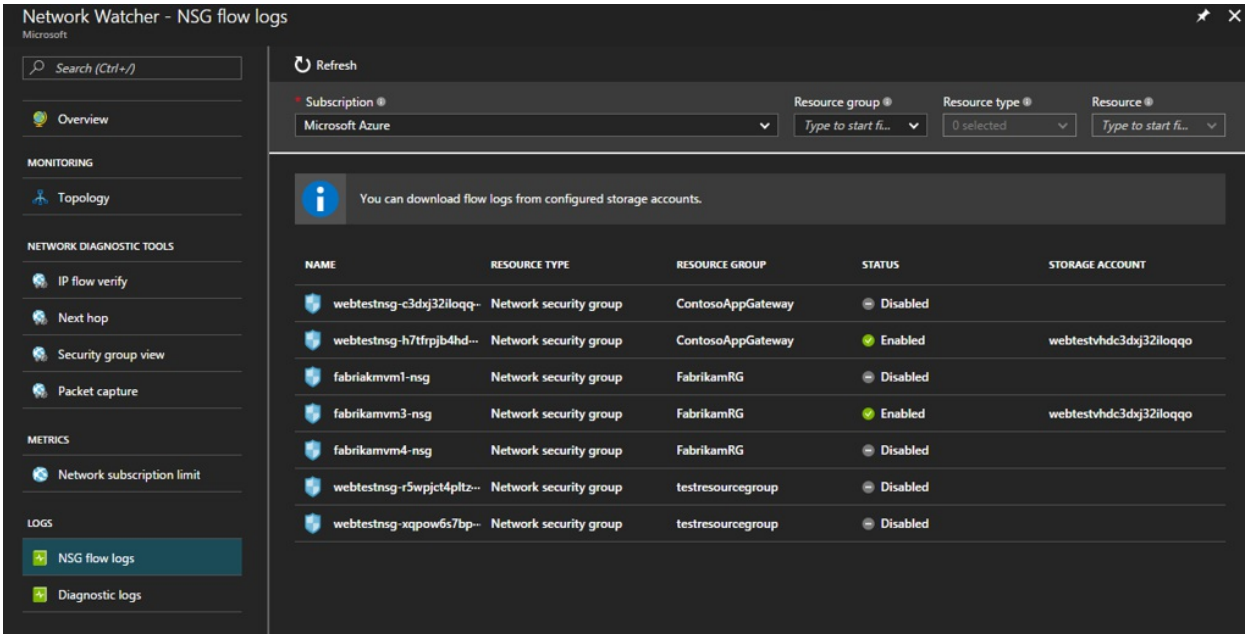
- WAF provides centralized protection of your web applications from common exploits and vulnerabilities.
- You don't need to modify your code to make use of WAF.
- It can protect multiple web apps at the same time, behind Application Gateway.
- WAF is integrated with Azure Security Center.
- You can customize WAF rules and rule groups to suit your application requirements.
- As a best practice, you should use a WAF in front of any web-facing application, including applications on Azure VMs or in Azure App Service.

Learn more:

- Learn about WAF.
- Review WAF limitations and exclusions.

Best practice: Implement Network Watcher

Network Watcher provides tools to monitor resources and communications in an Azure virtual network. For example, you can monitor communications between a VM and an endpoint, such as another VM or FQDN. You can also view resources and resource relationships in a virtual network, or diagnose network traffic issues.



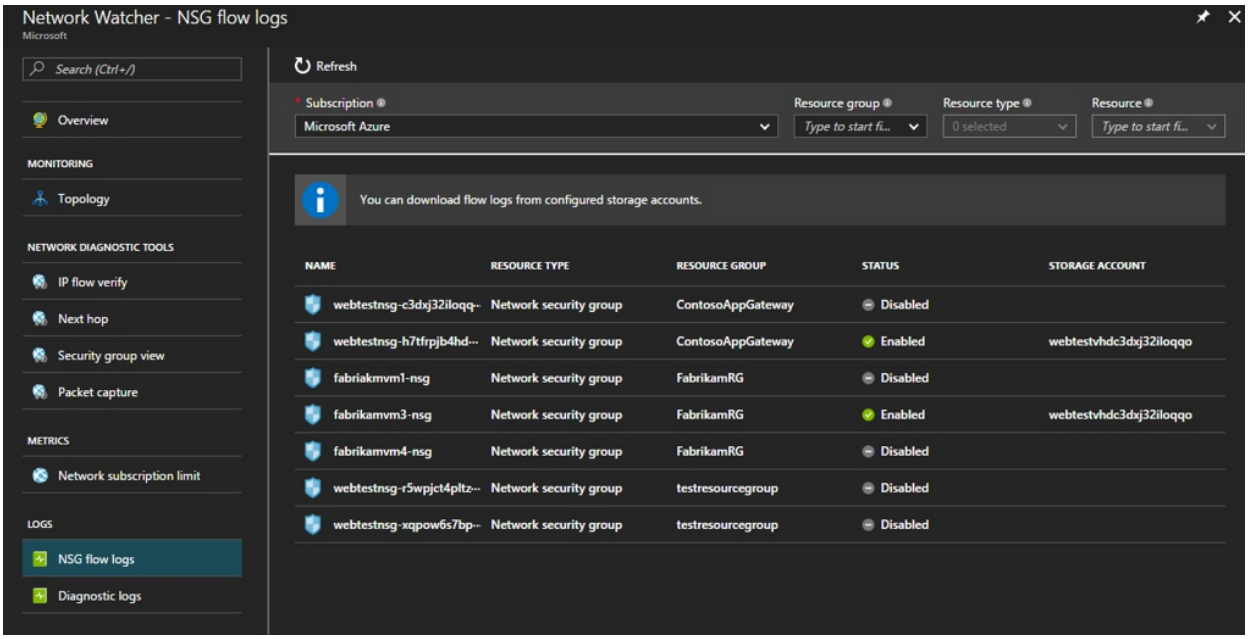


Figure 12: Network Watcher.

Here are a few more details:

- With Network Watcher, you can monitor and diagnose networking issues without signing into VMs.
- You can trigger packet capture by setting alerts, and gain access to real-time performance information at the packet level. When you see an issue, you can investigate it in detail.
- As a best practice, use Network Watcher to review NSG flow logs.
  - NSG flow logs in Network Watcher allow you to view information about ingress and egress IP traffic through an NSG.
  - Flow logs are written in JSON format.
  - Flow logs show outbound and inbound flows on a per-rule basis, and the network interface (NIC) to which the flow applies. They also show 5-tuple information about the flow (source/destination IP, source/destination port, and protocol), and whether the traffic was allowed or denied.

Learn more:

- Read the Network Watcher overview.
- Learn more about NSG flow logs.

Use partner tools in Azure Marketplace

For more complex network topologies, you might use security products from Microsoft partners, in particular network virtual appliances (NVAs).

- An NVA is a VM that performs a network function, such as a firewall, WAN optimization, or other network function.
- NVAs bolster virtual network security and network functions. They can be deployed for highly available firewalls, intrusion prevention, intrusion detection, WAFs, WAN optimization, routing, load balancing, VPN, certificate management, Active Directory, and multi-factor authentication.
- NVAs are available from numerous vendors in Azure Marketplace.

Best practice: Implement firewalls and NVAs in hub networks

In the hub, you normally manage the perimeter network (with access to the internet) through Azure Firewall, a firewall farm, or a WAF. The following table provides comparisons of these.

FIREWALL TYPE	DETAILS
WAFs	Web applications are common, and tend to suffer from vulnerabilities and potential exploits. WAFs are designed to detect attacks against web applications (HTTP/HTTPS). Compared with traditional firewall technology, WAFs have a set of specific features that protect internal web servers from threats.
Azure Firewall	Like NVA firewall farms, Azure Firewall uses a common administration mechanism and a set of security rules to protect workloads hosted in spoke networks. Azure Firewall also helps control access to on-premises networks. Azure Firewall has built-in scalability.
NVA firewalls	Like Azure Firewall, NVA firewall farms have a common administration mechanism and a set of security rules to protect workloads hosted in spoke networks. NVA firewalls also help control access to on-premises networks. NVA firewalls can be manually scaled behind a load balancer. Though an NVA firewall has less specialized software than a WAF, it has broader application scope to filter and inspect any type of traffic in egress and ingress.

We recommend using one set of Azure firewalls (or NVAs) for traffic originating on the internet, and another for traffic originating on-premises. Using only one set of firewalls for both is a security risk, as it provides no security perimeter between the two sets of network traffic. Using separate firewall layers reduces the complexity of checking security rules, and it's clear which rules correspond to which incoming network request.

Learn more:

- Learn about using NVAs in an Azure virtual network.

Next steps

Review other best practices:

- Best practices for security and management after migration.
- Best practices for cost management after migration.

Deploy a migration infrastructure

3/6/2021 • 38 minutes to read • Edit Online

This article shows how the fictional company Contoso prepares its on-premises infrastructure for migration, sets up an Azure infrastructure in preparation for migration, and runs the business in a hybrid environment.

When you use this example to help plan your own infrastructure migration efforts, keep in mind that the provided sample architecture is specific to Contoso. Review your organization's business needs, structure, and technical requirements when making important infrastructure decisions about subscription design or network architecture.

Whether you need all the elements described in this article depends on your migration strategy. For example, you might need a less complex network structure if you're building only cloud-native applications in Azure.

Overview

Before Contoso can migrate to Azure, it's critical to prepare an Azure infrastructure. Generally, Contoso needs to think about six areas:

- **Step 1: Azure subscriptions.** How will IT purchase Azure and interact with the Azure platform and services?
- **Step 2: Hybrid identity.** How will IT manage and control access to on-premises and Azure resources after migration? How does IT extend or move identity management to the cloud?
- **Step 3: Disaster recovery and resilience.** How will IT ensure that its applications and infrastructure are resilient if outages and disasters occur?
- **Step 4: Network.** How should IT design a network infrastructure and establish connectivity between its on-premises datacenter and Azure?
- **Step 5: Security.** How will IT secure the hybrid deployment?
- **Step 6: Governance.** How will IT keep the deployment aligned with security and governance requirements?

Before you start

Before we start reviewing the infrastructure, consider reading some background information about relevant Azure capabilities:

- Several options are available for purchasing Azure access, including pay-as-you-go subscriptions, a Microsoft Enterprise Agreement (EA), Open Licensing from Microsoft resellers, or purchasing from Microsoft partners in the Cloud Solution Provider (CSP) program. Learn about purchase options, and read about how Azure subscriptions are organized.
- Get an overview of Azure identity and access management (IAM). Learn about Azure Active Directory (Azure AD) and extending on-premises Active Directory to the cloud.
- Azure provides a robust network infrastructure with options for hybrid connectivity. Get an overview of networking and network access control.
- Read the introduction to Azure security and learn how to create a plan for Azure governance.

On-premises architecture

Here's a diagram that shows the current Contoso on-premises infrastructure.

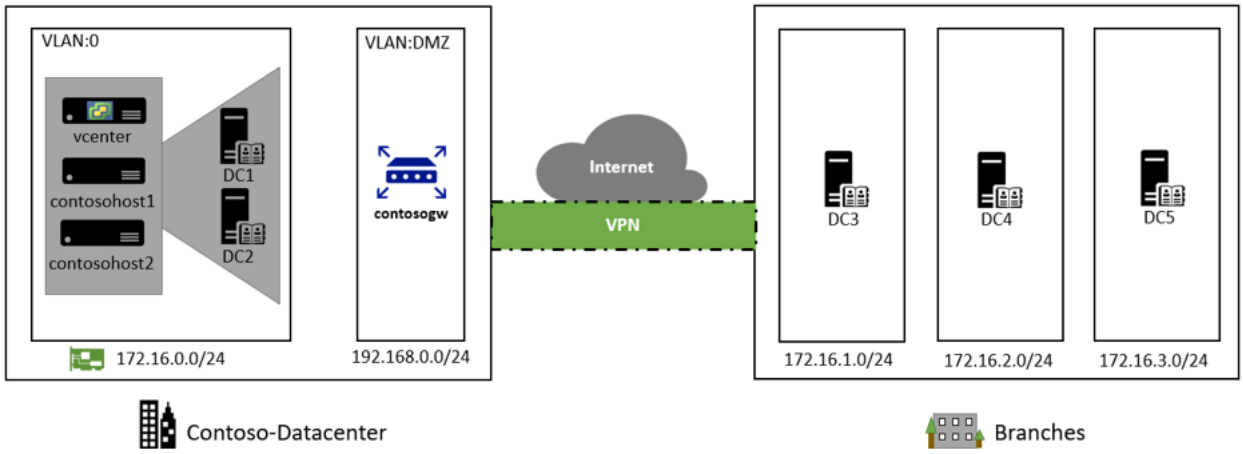


Figure 1: Contoso on-premises architecture.

- Contoso has one main datacenter located in New York City in the eastern United States.
- There are three additional local branches across the United States.
- The main datacenter is connected to the internet with a fiber-optic Metro Ethernet connection (500 Mbps).
- Each branch is connected locally to the internet through business-class connections, with IPsec VPN tunnels back to the main datacenter. This approach allows the entire network to be permanently connected and optimizes internet connectivity.
- The main datacenter is fully virtualized with VMware. Contoso has two ESXi 6.5 virtualization hosts managed by vCenter Server 6.5.
- Contoso uses Active Directory for identity management and Domain Name System (DNS) servers on the internal network.
- The domain controllers in the datacenter run on VMware virtual machines (VMs). The domain controllers at local branches run on physical servers.

Step 1: Buy and subscribe to Azure

Contoso needs to figure out how to buy Azure, how to manage subscriptions, and how to license services and resources.

Buy Azure

Contoso is enrolling in an Enterprise Agreement. This agreement entails an upfront monetary commitment to Azure, which entitles Contoso to earn benefits like flexible billing options and optimized pricing.

Here are the details:

- Contoso estimated what its yearly Azure spend will be. When Contoso signed the agreement, it paid for the first year in full.
- Contoso needs to use all commitments before the year is over or lose the value for those dollars.
- If for some reason Contoso exceeds its commitment and spends more, Microsoft will invoice for the difference.
- Any cost incurred above the commitment will be at the same rates as those in the Contoso contract. There are no penalties for going over.

Manage subscriptions

After paying for Azure, Contoso needs to figure out how to manage Azure subscriptions. Because Contoso has an EA, there's no limit on the number of Azure subscriptions it can create. An Azure Enterprise Agreement enrollment defines how a company shapes and uses Azure services, and defines a core governance structure.

As a first step, Contoso has defined a structure known as an *enterprise scaffold* for its enrollment. Contoso used the Azure enterprise scaffold guidance to help understand and design a scaffold.

For now, Contoso has decided to use a functional approach to manage subscriptions:

- Inside the enterprise, it will use a single IT department that controls the Azure budget. This will be the only group with subscriptions.
- Contoso will extend this model in the future, so that other corporate groups can join as departments in the enrollment hierarchy.
- Inside the IT department, Contoso has structured two subscriptions, `Production` and `Development`.
- If Contoso needs more subscriptions in the future, it will also need to manage access, policies, and compliance for those subscriptions. Contoso will do that by introducing Azure management groups as an additional layer above subscriptions.

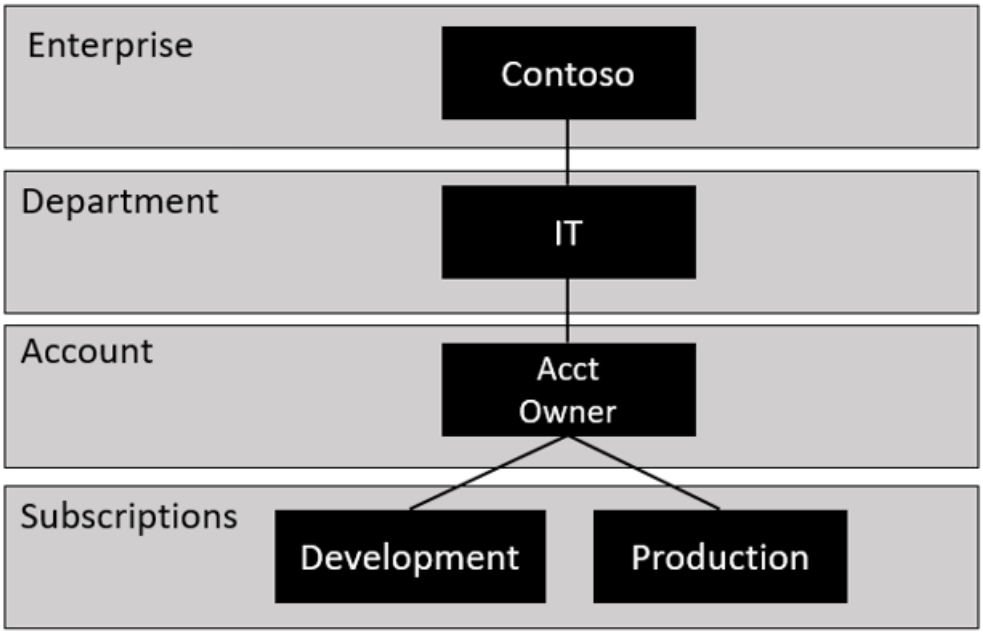


Figure 2: Enterprise hierarchy.

Examine licensing

With subscriptions configured, Contoso can look at Microsoft licensing. The licensing strategy will depend on the resources that Contoso wants to migrate to Azure and how VMs and services are selected and deployed in Azure.

Azure Hybrid Benefit

For deploying VMs in Azure, standard images include a license that will charge Contoso by the minute for the software being used. However, Contoso has been a long-term Microsoft customer and has maintained EAs and open licenses with Software Assurance.

Azure Hybrid Benefit provides a cost-effective method for migration. It allows Contoso to save on Azure VMs and SQL Server workloads by converting or reusing Windows Server Datacenter and Standard edition licenses covered with Software Assurance. This allows Contoso to pay a lower base compute rate for VMs and SQL Server. For more information, see Azure Hybrid Benefit.

License Mobility

License Mobility through Software Assurance gives Microsoft Volume Licensing customers like Contoso the flexibility to deploy eligible server applications with active Software Assurance on Azure. This eliminates the need to purchase new licenses. With no associated mobility fees, existing licenses can easily be deployed in Azure. For more information, see License Mobility through Software Assurance on Azure.

Reserved instances for predictable workloads

Predictable workloads always need to be available with VMs running, such as line-of-business applications like an SAP ERP system. Unpredictable workloads are variable, like VMs that are on during high demand and off when demand is low.

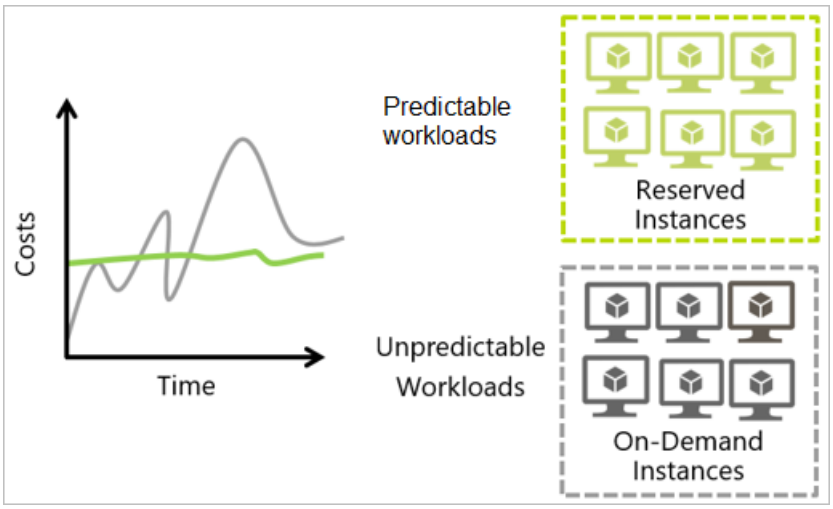


Figure 3: Azure Reserved Virtual Machine Instances.

In exchange for using reserved instances for specific VM instances that must be maintained for long durations, Contoso can get both a discount and prioritized capacity. Using Azure Reserved Virtual Machine Instances together with Azure Hybrid Benefit can save Contoso up to 82 percent off regular pay-as-you-go pricing (as of April 2018).

Step 2: Manage hybrid identity

Giving and controlling user access to Azure resources with identity and access management is an important step in pulling together an Azure infrastructure.

Contoso decides to extend its on-premises Active Directory into the cloud, rather than build a new separate system in Azure. Because Contoso isn't using Microsoft 365 yet, it needs to provision an Azure AD instance. If Contoso were using Microsoft 365, it would already have an existing Azure AD tenant and directory, which it could use as its primary Azure AD instance.

Learn more about Microsoft 365 identity models and Azure Active Directory. You can also learn how to associate or add an Azure subscription to your Azure Active Directory tenant.

Create an Azure AD directory

Contoso is using the Azure AD Free edition that's included with an Azure subscription. Contoso admins create an Azure AD directory:

1. In the Azure portal, they go to **Create a resource** > **Identity** > **Azure Active Directory**.
2. In **Create directory**, they specify a name for the directory, an initial domain name, and the region where the directory should be created.

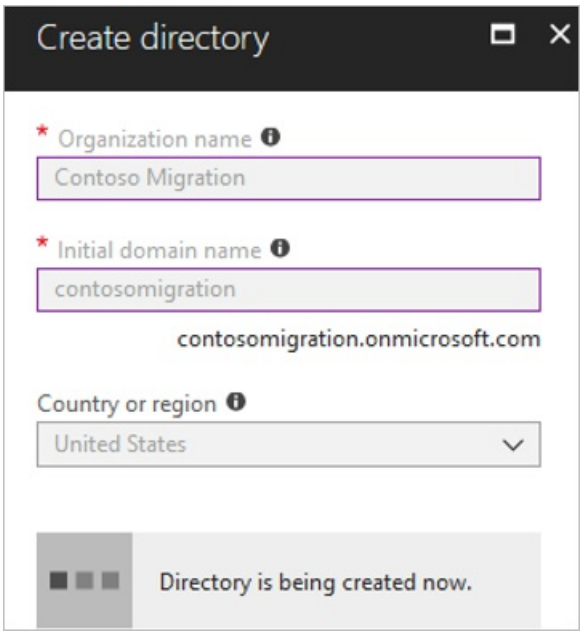


Figure 4: Create an Azure AD directory.

> **NOTE**
> The directory that's created has an initial domain name in the form `domain-name.onmicrosoft.com`. The name can't be changed or deleted. Instead, the admins need to add its registered domain name to Azure AD.

Add the domain name

To use the standard domain name, Contoso admins need to add it as a custom domain name to Azure AD. This option allows them to assign familiar user names. For example, a user can sign in with the email address `billg@contoso.com` instead of `billg@contosomigration.onmicrosoft.com`.

To set up a custom domain name, the admins add it to the directory, add a DNS entry, and then verify the name in Azure AD.

1. In **Custom domain names** > **Add custom domain**, they add the domain.
2. To use a DNS entry in Azure, they need to register it with their domain registrar:
   - In the **Custom domain names** list, they note the DNS information for the name. It's using an MX record.
   - They need access to the name server. They sign in to the `contoso.com` domain and create a new MX record for the DNS entry provided by Azure AD, by using the details noted.
3. After the DNS records propagate, they select **Verify** to check the custom domain name in the details for the domain.

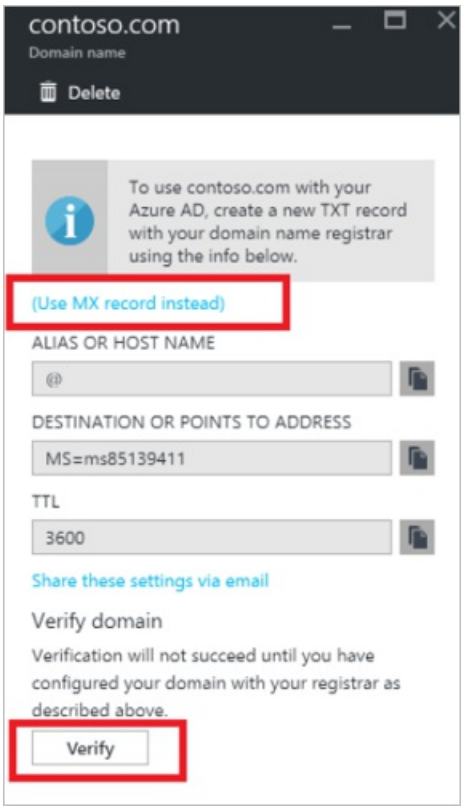


Figure 5: Checking the domain name.

Set up on-premises and Azure groups and users

Now that the Azure AD directory is established, Contoso admins need to add employees to on-premises Active Directory groups that will synchronize to Azure AD. They should use on-premises group names that match the names of resource groups in Azure. This makes it easier to identify matches for synchronization purposes.

Create resource groups in Azure

Azure resource groups gather Azure resources together. Using a resource group ID allows Azure to perform operations on the resources within the group.

An Azure subscription can have multiple resource groups. A resource group exists in a single subscription. In addition, a single resource group can have multiple resources. A resource belongs to a single resource group.

Contoso admins set up Azure resource groups as shown in the following table.

RESOURCE GROUP	DETAILS
`ContosoCobRG`	This group contains all resources related to continuity of business. It includes vaults that Contoso will use for the Azure Site Recovery service and the Azure Backup service. It also includes resources used for migration, including Azure Migrate and Azure Database Migration Service.
`ContosoDevRG`	This group contains dev/test resources.
`ContosoFailoverRG`	This group serves as a landing zone for failed-over resources.
`ContosoNetworkingRG`	This group contains all network resources.
`ContosoRG`	This group contains resources related to production applications and databases.

They create resource groups as follows:

1. In the Azure portal > **Resource groups**, they add a group.
2. For each group, they specify a name, the subscription to which the group belongs, and the region.
3. Resource groups appear in the **Resource groups** list.

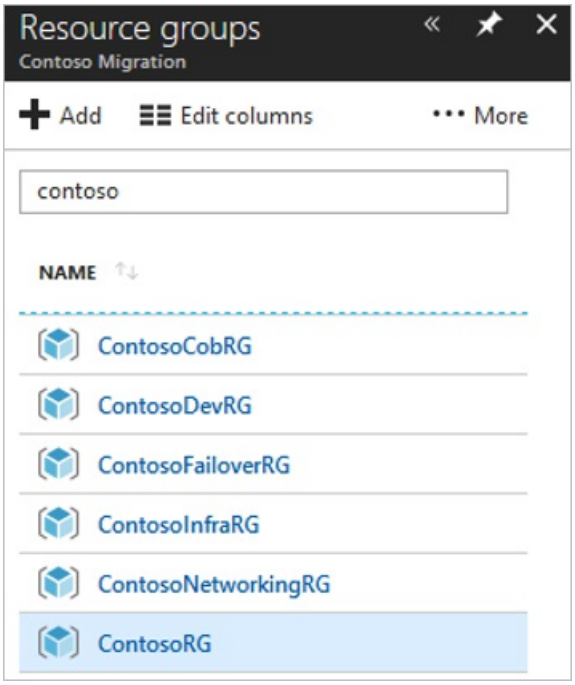


Figure 6: Resource groups.

Scale resource groups

In future, Contoso will add other resource groups based on needs. For example, it might define a resource group for each application or service so that each can be managed and secured independently.

Create matching security groups on-premises

In the on-premises Active Directory instance, Contoso admins set up security groups with names that match the names of the Azure resource groups.

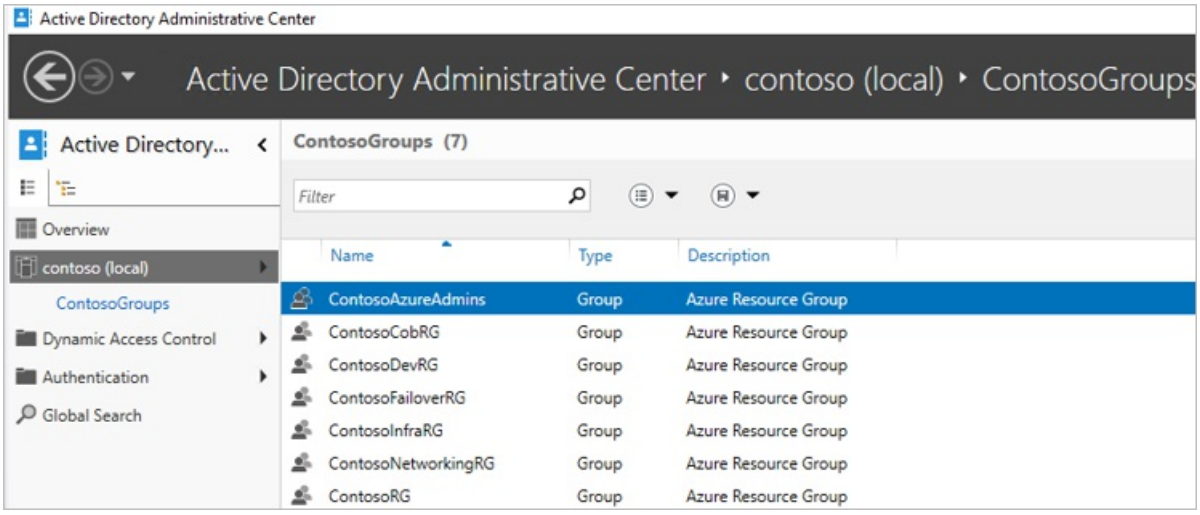


Figure 7: On-premises Active Directory security groups.

For management purposes, they create an additional group that will be added to all of the other groups. This group will have rights to all resource groups in Azure. A limited number of global admins will be added to this group.

Synchronize Active Directory

Contoso wants to provide a common identity for accessing resources on-premises and in the cloud. To do this, it

will integrate the on-premises Active Directory instance with Azure AD. With this model, users and organizations can take advantage of a single identity to access on-premises applications and cloud services, such as Microsoft 365, or thousands of other sites on the internet. Admins can use the groups in Active Directory to implement Azure role-based access control (Azure RBAC).

To facilitate integration, Contoso uses the Azure AD Connect tool. When you install and configure the tool on a domain controller, it synchronizes the on-premises Active Directory identities to Azure AD.

Download the tool

1. In the Azure portal, Contoso admins go to **Azure Active Directory** > **Azure AD Connect** and download the latest version of the tool to the server they're using for synchronization.

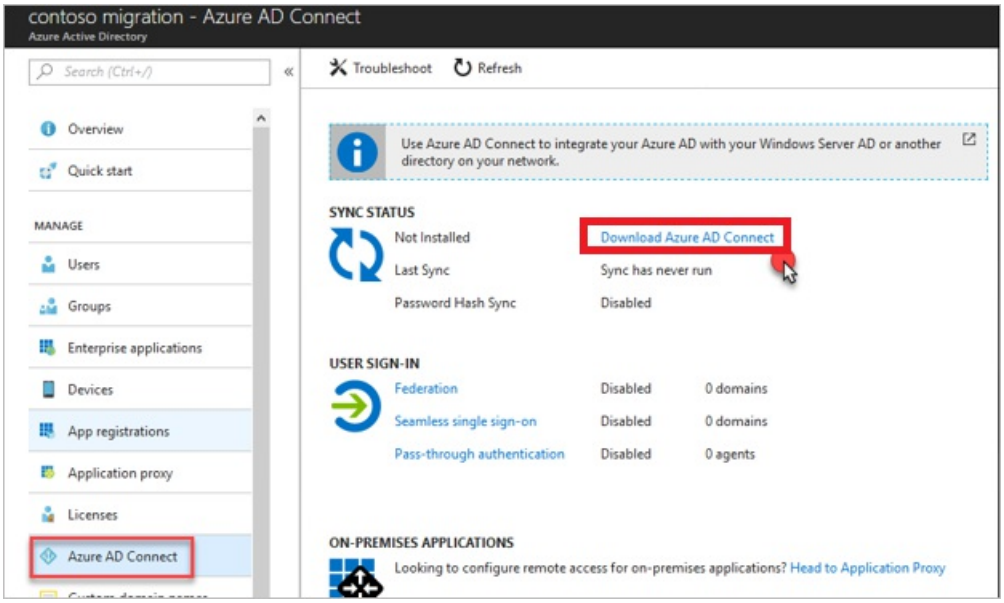


Figure 8: Downloading Azure AD Connect.

2. They start the `AzureADConnect.msi` installation by using **Express Settings**. This is the most common installation, and it can be used for a single-forest topology with password-hash synchronization for authentication.

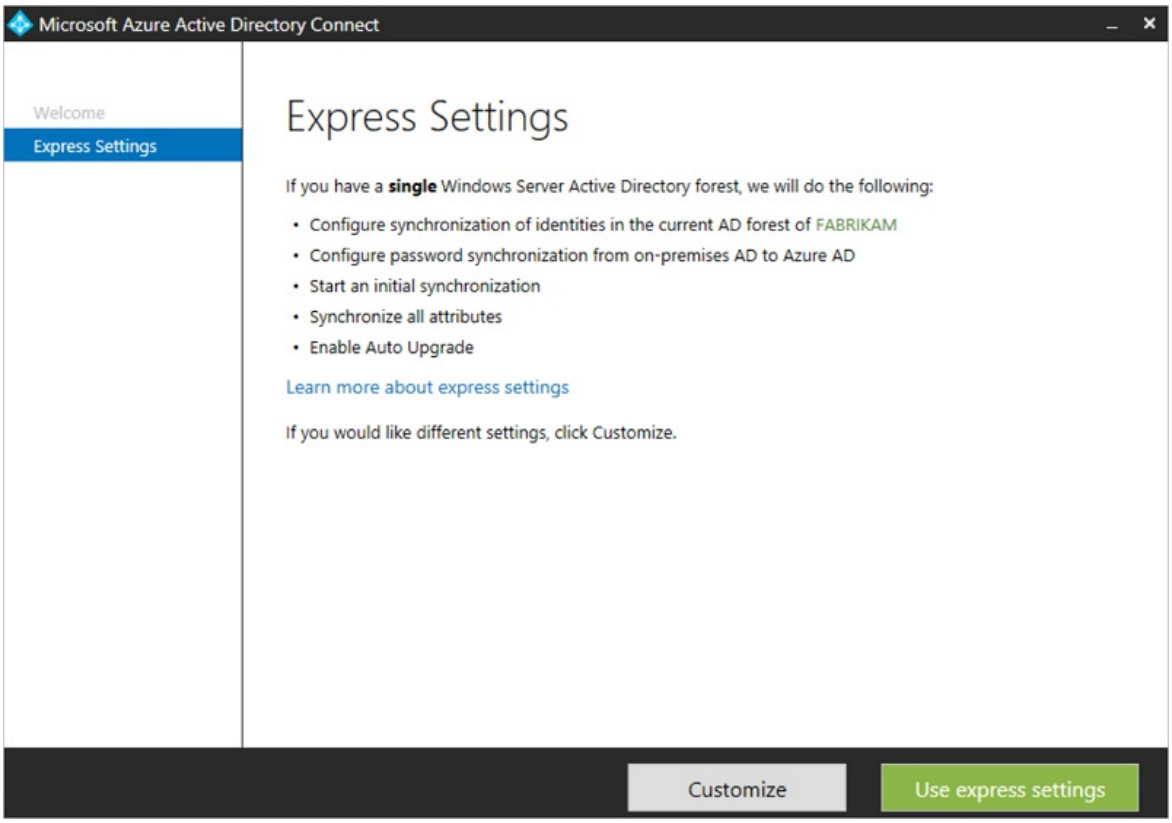


Figure 9: Azure AD Connect Wizard.

3. In **Connect to Azure AD**, they specify the credentials for connecting to Azure AD (in the form `admin@contoso.com` or `admin@contoso.onmicrosoft.com`).

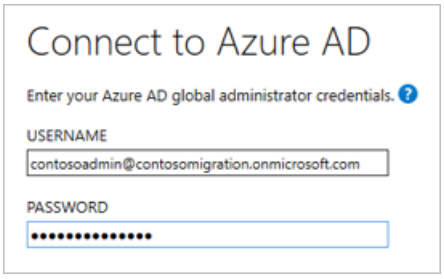


Figure 10: Azure AD Connect Wizard: Connect to Azure AD.

4. In **Connect to AD DS**, they specify credentials for the on-premises directory (in the form `CONTOSO\admin` or `contoso.com\admin`).

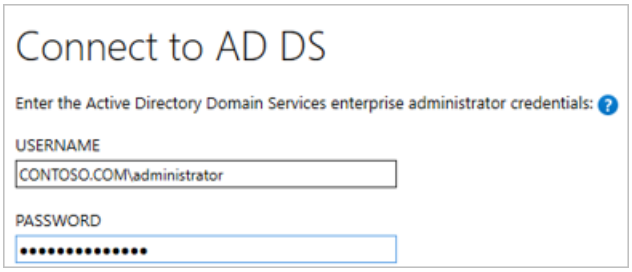


Figure 11: Azure AD Connect Wizard: Connect to AD DS.

5. In **Ready to configure**, they select **Start the synchronization process when configuration completes** to start the sync immediately. Then they install.

   Note the following:

   - Contoso has a direct connection to Azure. If your on-premises Active Directory instance is behind a proxy, review troubleshoot Azure AD connectivity.
   - After the first synchronization, on-premises Active Directory objects are visible in the Azure AD directory.

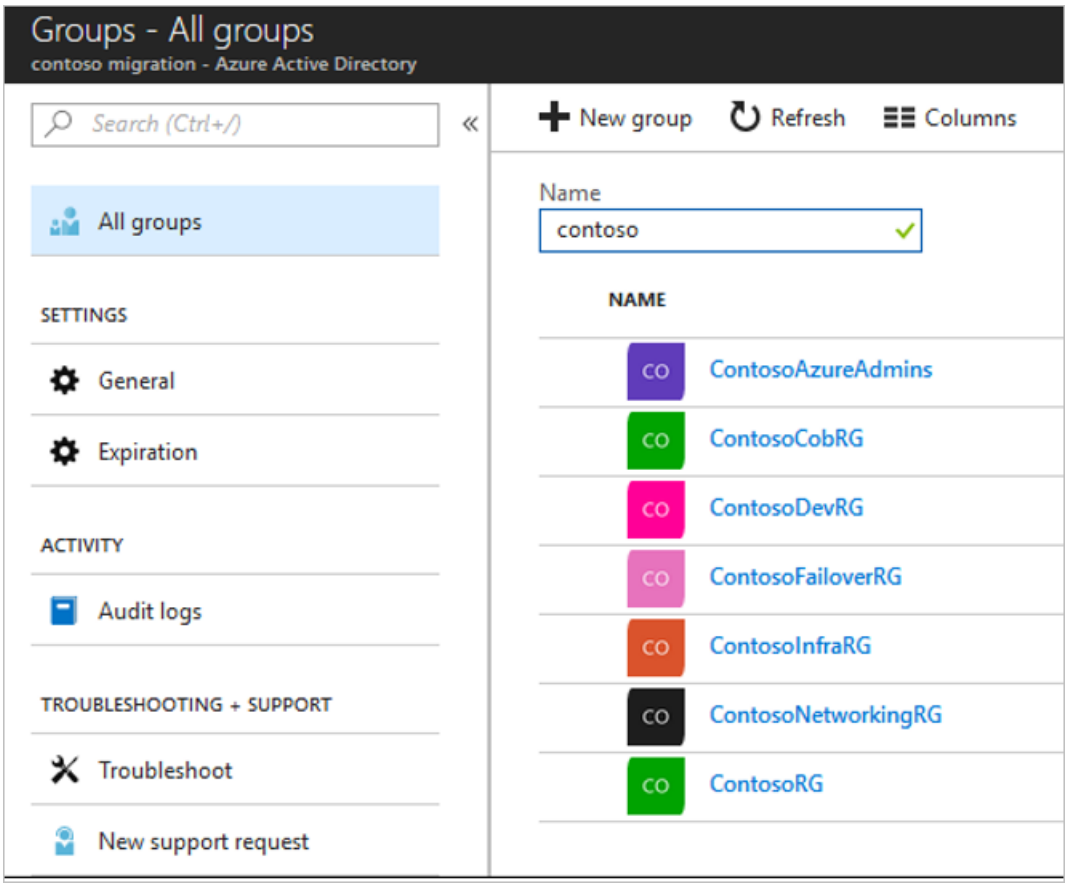


Figure 12: On-premises Active Directory objects visible in Azure AD.

- The Contoso IT team is represented in each group and is based on its role.

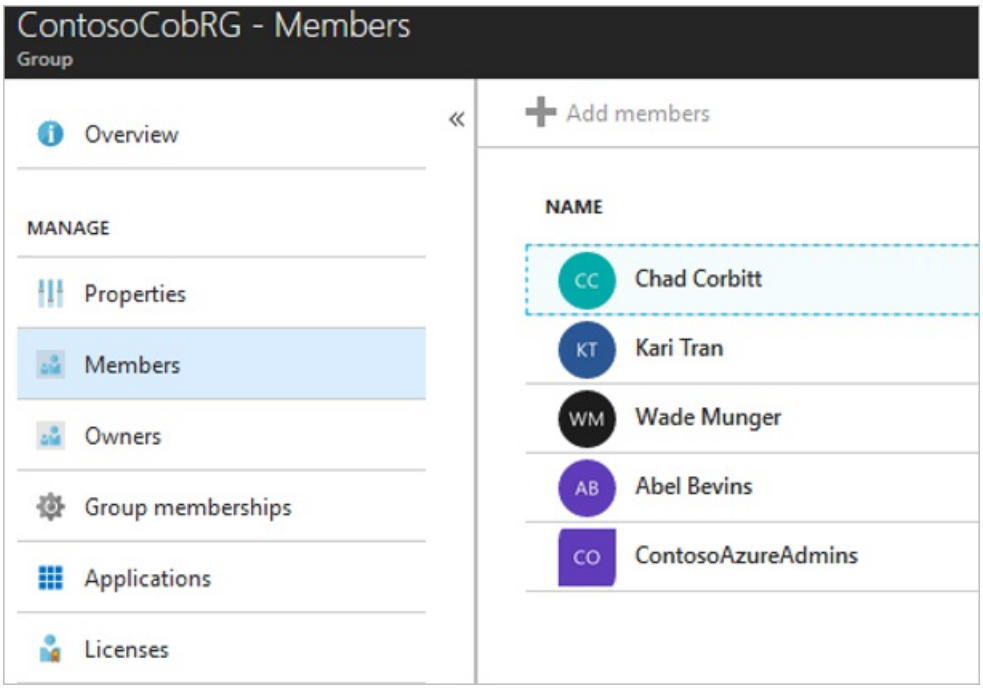


Figure 13: Group membership.

Set up Azure RBAC

Azure RBAC enables fine-grained access management for Azure. By using Azure RBAC, you can grant only the amount of access that users need to perform tasks. You assign the appropriate Azure role to users, groups, and applications at a scope level. The scope of a role assignment can be a subscription, a resource group, or a single resource.

Contoso admins then assign roles to the Active Directory groups that they synchronized from on-premises.

1. In the `ControlCobRG` resource group, they select **Access control (IAM)** > **Add role assignment**.
2. In **Add role assignment** > **Role** > **Contributor**, they select the `ContosoCobRG` security group from the list. The group then appears in the **Selected members** list.
3. They repeat this with the same permissions for the other resource groups (except for `ContosoAzureAdmins`) by adding **Contributor** permissions to the security group that matches the resource group.
4. For the `ContosoAzureAdmins` security group, they assign the **Owner** role.

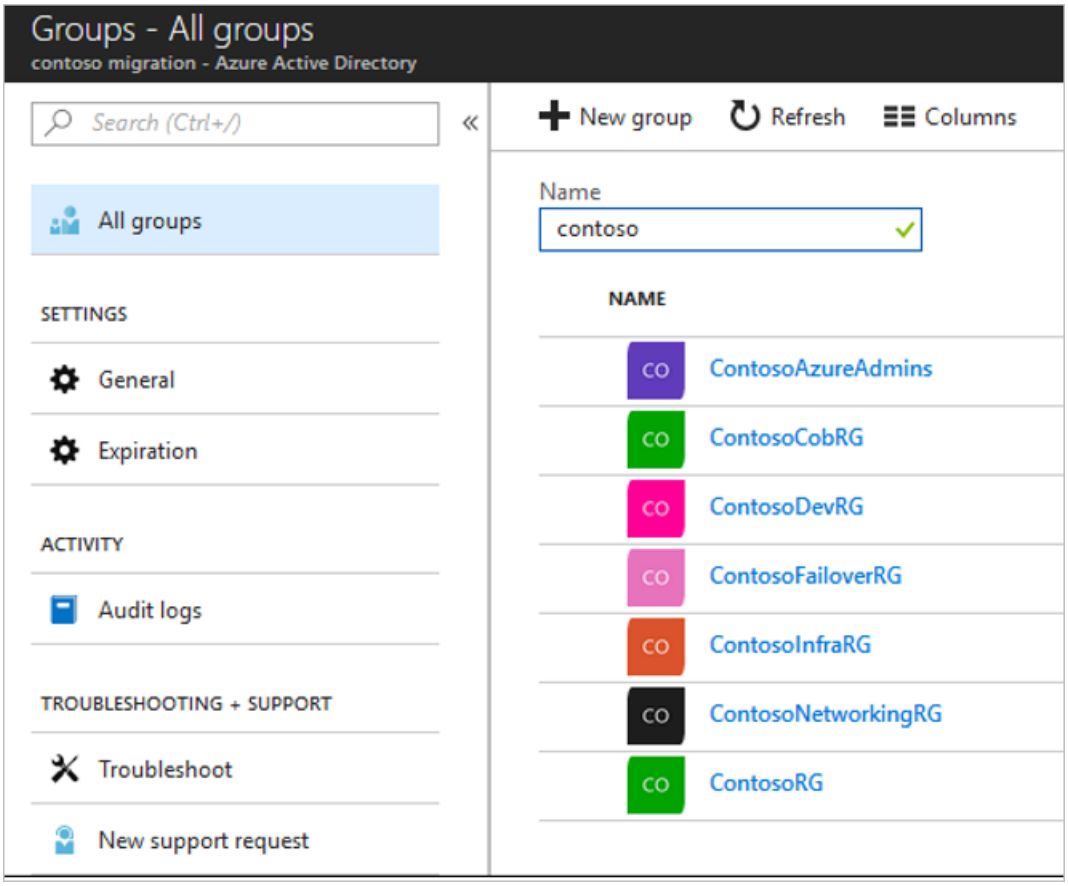


Figure 14: Assigning roles to security groups.

Step 3: Design for resiliency

Set up regions

Azure resources are deployed within regions. Regions are organized into geographies. Data residency, sovereignty, compliance, and resiliency requirements are honored within geographical boundaries.

A region consists of a set of datacenters. These datacenters are deployed within a latency-defined perimeter, and connected through a dedicated regional low-latency network.

Each Azure region is paired with a different region for resiliency. Read about Azure regions, and understand how regions are paired.

Contoso has decided to use `East US 2` (located in Virginia) as the primary region and `Central US` (located in Iowa) as the secondary region, for these reasons:

- The Contoso datacenter is located in New York, and Contoso considered latency to the closest datacenter.
- `East US 2` has all the services and products that Contoso needs. Not all Azure regions have the same products and services available. For more information, see Azure products by region.
- `Central US` is the Azure paired region for `East US 2`.

As it thinks about the hybrid environment, Contoso needs to consider how to build resilience and a disaster recovery strategy into the region design. The simplest strategy is a single-region deployment, which relies on Azure platform features such as fault domains and regional pairing for resilience. The most complex is a full active-active model in which cloud services and database are deployed and servicing users from two regions.

Contoso has decided to take a middle road. It will deploy applications and resources in a primary region and keep a full copy of the infrastructure in the secondary region. With that strategy, the copy is ready to act as a full backup if a complete application disaster or regional failure occurs.

Set up availability

Availability sets

Availability sets help protect applications and data from a local hardware and network outage within a datacenter. Availability sets distribute Azure VMs across physical hardware within a datacenter.

Fault domains represent underlying hardware with a common power source and network switch within the datacenter. VMs in an availability set are distributed across fault domains to minimize outages caused by a single hardware or network failure.

Update domains represent underlying hardware that can undergo maintenance or be rebooted at the same time. Availability sets also distribute VMs across multiple update domains to ensure that at least one instance will be running at all times.

Contoso will implement availability sets whenever VM workloads require high availability. For more information, see Manage the availability of Windows VMs in Azure.

Availability Zones

Availability Zones help protect applications and data from failures that affect an entire datacenter within a region.

Each Availability Zone represents a unique physical location within an Azure region. Each zone consists of one or more datacenters equipped with independent power, cooling, and networking.

There's a minimum of three separate zones in all enabled regions. The physical separation of zones within a region protects applications and data from datacenter failures.

Contoso will use Availability Zones whenever applications need greater scalability, availability, and resilience. For more information, see Regions and Availability Zones in Azure.

Configure backup

Azure Backup

You can use Azure Backup to back up and restore Azure VM disks.

Azure Backup allows automated backups of VM disk images stored in Azure Storage. Backups are application consistent to ensure that backed-up data is transactionally consistent and that applications will start post-restore.

Azure Backup supports locally redundant storage (LRS) to replicate multiple copies of backup data within a datacenter if a local hardware failure occurs. If a regional outage occurs, Azure Backup also supports geo-redundant storage (GRS), which replicates backup data to a secondary paired region.

Azure Backup encrypts data in transit by using AES-256. Backed-up data at rest is encrypted through Azure Storage encryption.

Contoso will use Azure Backup with GRS on all production VMs to ensure that workload data is backed up and can be quickly restored if a disruption occurs. For more information, see An overview of Azure VM backup.

Set up disaster recovery

Azure Site Recovery

Azure Site Recovery helps ensure business continuity by keeping business applications and workloads running during regional outages.

Azure Site Recovery continuously replicates Azure VMs from a primary to a secondary region, ensuring functional copies in both locations. In the event of an outage in the primary region, the application or service fails over to using the VM instances replicated in the secondary region. This failover minimizes potential disruption. When operations return to normal, the applications or services can fail back to VMs in the primary region.

Contoso will implement Azure Site Recovery for all production VMs used in mission-critical workloads, ensuring minimal disruption during an outage in the primary region.

Step 4: Design a network infrastructure

With the regional design in place, Contoso is ready to consider a network strategy. It needs to think about how the on-premises datacenter and Azure connect and communicate with each other, and how to design the network infrastructure in Azure. Specifically, Contoso needs to:

- **Plan hybrid network connectivity.** Figure out how it's going to connect networks across on-premises and Azure.
- **Design an Azure network infrastructure.** Decide how it will deploy networks over regions. How will networks communicate within the same region and across regions?
- **Design and set up Azure networks.** Set up Azure networks and subnets, and decide what will reside in them.

Plan hybrid network connectivity

Contoso considered several architectures for hybrid networking between Azure and the on-premises datacenter. For more information, see Choose a solution for connecting an on-premises network to Azure.

As a reminder, the Contoso on-premises network infrastructure currently consists of the datacenter in New York, and local branches in the eastern half of the United States. All locations have a business-class connection to the internet. Each of the branches is then connected to the datacenter via an IPsec VPN tunnel over the internet.

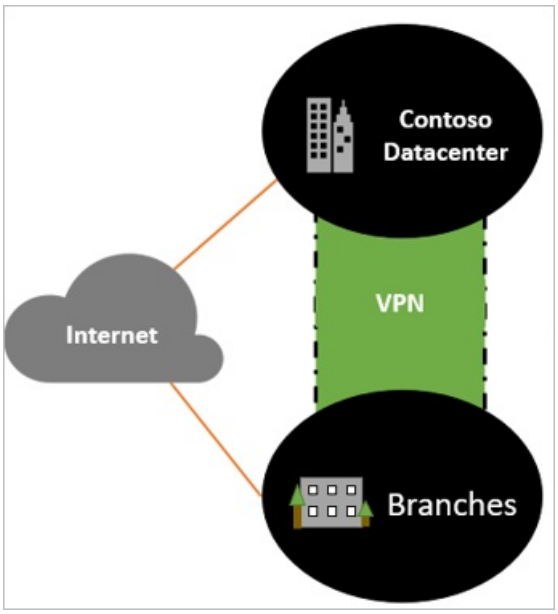


Figure 15: The Contoso network.

Here's how Contoso decided to implement hybrid connectivity:

1. Set up a new Site-to-Site VPN connection between the Contoso datacenter in New York and the two Azure regions, `East US 2` and `Central US`.
2. Branch office traffic bound for virtual networks in Azure will route through the main Contoso datacenter.
3. As Contoso scales up Azure deployment, it will establish an Azure ExpressRoute connection between the

datacenter and the Azure regions. When this happens, Contoso will retain the VPN Site-to-Site connection for failover purposes only.

- Learn more about choosing between a VPN and ExpressRoute hybrid solution.
- Verify ExpressRoute locations and support.

VPN only:

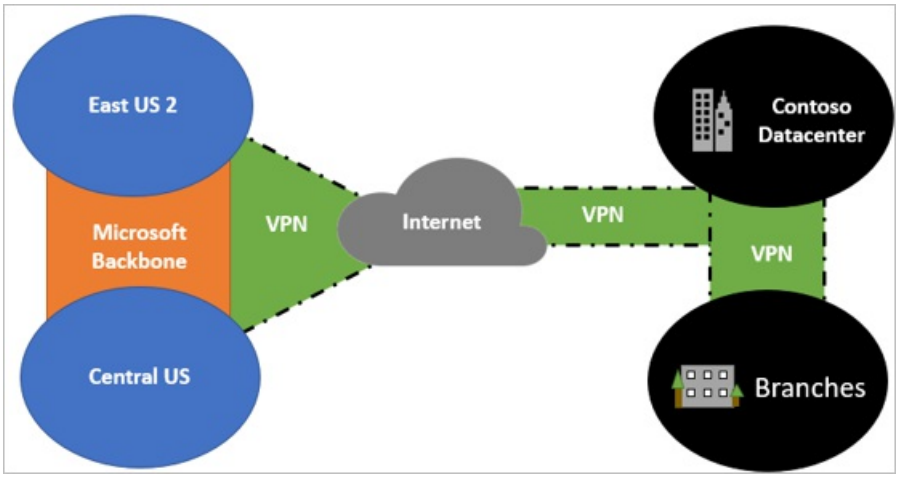


Figure 16: The Contoso VPN.

VPN and ExpressRoute:

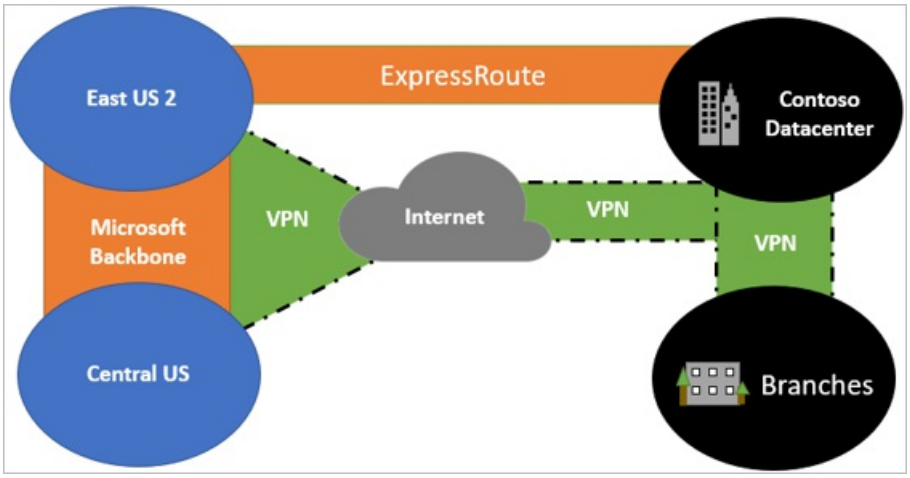


Figure 17: Contoso VPN and ExpressRoute.

Design the Azure network infrastructure

Contoso's network configuration must make the hybrid deployment secure and scalable. Contoso is taking a long-term approach to this, designing virtual networks to be resilient and enterprise ready. For more information, see Plan virtual networks.

To connect the two regions, Contoso will implement a hub-to-hub network model. Within each region, Contoso will use a hub-and-spoke model. To connect networks and hubs, Contoso will use Azure network peering.

Network peering

Virtual network peering in Azure connects virtual networks and hubs. Global peering allows connections between virtual network or hubs in different regions. Local peering connects virtual networks in the same region.

Virtual network peering provides several advantages:

- Network traffic between peered virtual networks is private.
- Traffic between the virtual networks is kept on the Microsoft backbone network. No public internet, gateways, or encryption is required in the communication between the virtual networks.
- Peering provides a default, low-latency, high-bandwidth connection between resources in different virtual

networks.

Hub-to-hub model across regions

Contoso will deploy a hub in each region. A hub is a virtual network in Azure that acts as a central point of connectivity to your on-premises network. The hub virtual networks will connect to each other via global virtual network peering, which connects virtual networks across Azure regions. The hub in each region is peered to its partner hub in the other region. The hub is peered to every network in its region, and it can connect to all network resources.

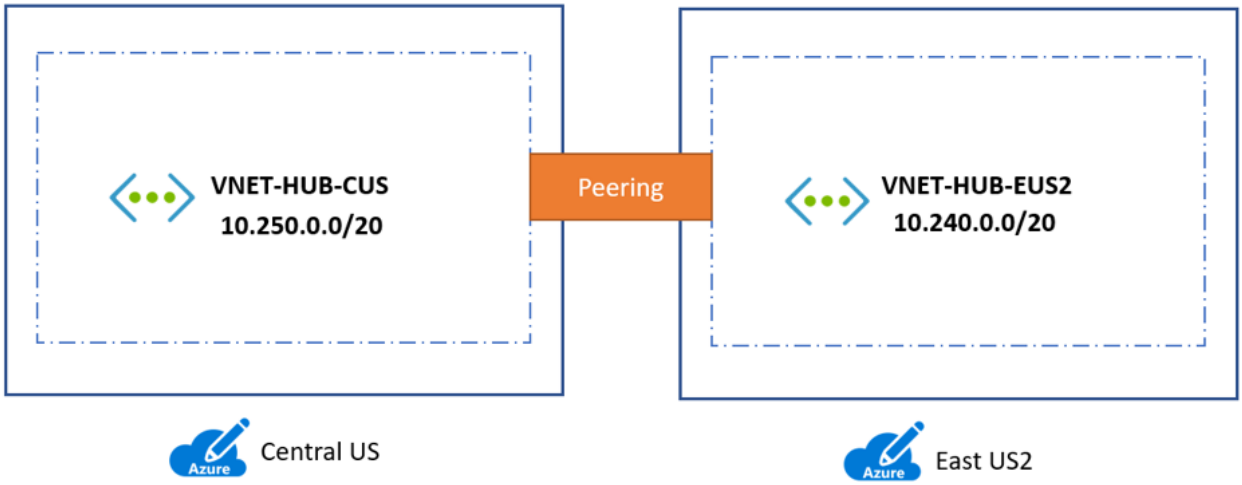


Figure 18: Global peering.

Hub-and-spoke model within a region

Within each region, Contoso will deploy virtual networks for different purposes as spoke networks from the region hub. Virtual networks within a region use peering to connect to their hub and to each other.

Design the hub network

Within the hub-and-spoke model, Contoso needed to think about how traffic from the on-premises datacenter and from the internet would be routed. Here's how Contoso decided to handle routing for both the `East US 2` and `Central US` hubs:

- Contoso is designing a network that allows traffic from the internet and from the corporate network by using a VPN to Azure.
- The network architecture has two boundaries, an untrusted front-end perimeter zone and a back-end trusted zone.
- A firewall will have a network adapter in each zone, controlling access to trusted zones.
- From the internet:
  - Internet traffic will hit a load-balanced public IP address on the perimeter network.
  - This traffic is routed through the firewall and subject to firewall rules.
  - After network access controls are implemented, traffic will be forwarded to the appropriate location in the trusted zone.
  - Outbound traffic from the virtual network will be routed to the internet through user-defined routes. The traffic is forced through the firewall and inspected in line with Contoso policies.
- From the Contoso datacenter:
  - Incoming traffic over Site-to-Site VPN or ExpressRoute hits the public IP address of the Azure VPN gateway.
  - Traffic is routed through the firewall and subject to firewall rules.
  - After the application of firewall rules, traffic is forwarded to an internal load balancer (Standard SKU) on the trusted internal zone subnet.
  - Outbound traffic from the trusted subnet to the on-premises datacenter over the VPN is routed through the firewall. Rules are applied before traffic goes over the Site-to-Site VPN connection.

Design and set up Azure networks

With a network and routing topology in place, Contoso is ready to set up Azure networks and subnets:

- Contoso will implement a class-A private network in Azure (`10.0.0.0/8`). This works because of on-premises; it currently has a class-B private address space (`172.160.0.0/16`). Contoso can be sure there won't be any overlap between address ranges.
- Contoso will deploy virtual networks in both the primary and secondary regions.
- Contoso will use a naming convention that includes the prefix `VNET` and the region abbreviation `EUS2` or `CUS`. Using this standard, the hub networks will be named `VNET-HUB-EUS2` in the `East US 2` region and `VNET-HUB-CUS` in the `Central US` region.

Virtual networks in `East US 2`

`East US 2` is the primary region that Contoso will use to deploy resources and services. Here's how Contoso will design networks in that region:

- **Hub:** The hub virtual network in `East US 2` is considered Contoso's primary connectivity to the on-premises datacenter.
- **Virtual networks:** The spoke virtual networks in `East US 2` can be used to isolate workloads if necessary. In addition to the hub virtual network, Contoso will have two spoke virtual networks in `East US 2`:
  - `VNET-DEV-EUS2`. This virtual network will provide the dev/test team with a fully functional network for dev projects. It will act as a production pilot area, and will rely on the production infrastructure to function.
  - `VNET-PROD-EUS2`. Azure IaaS production components will be located in this network.

  Each virtual network will have its own unique address space without overlap. Contoso intends to configure routing without requiring network address translation (NAT).
- **Subnets:** There will be a subnet in each network for each application tier. Each subnet in the production network will have a matching subnet in the development virtual network. The production network has a subnet for domain controllers.

The following table summarizes virtual networks in `East US 2`.

VIRTUAL NETWORK	RANGE	PEER
`VNET-HUB-EUS2`	`10.240.0.0/20`	`VNET-HUB-CUS2`, `VNET-DEV-EUS2`, `VNET-PROD-EUS2`
`VNET-DEV-EUS2`	`10.245.16.0/20`	`VNET-HUB-EUS2`
`VNET-PROD-EUS2`	`10.245.32.0/20`	`VNET-HUB-EUS2`, `VNET-PROD-CUS`

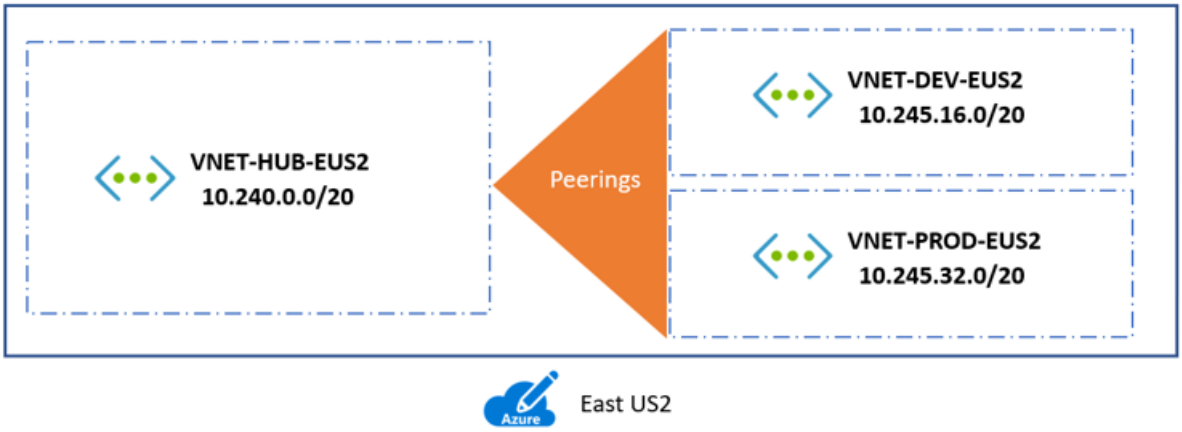


Figure 19: A hub-and-spoke model.

Subnets in the `East US 2 Hub` network (`VNET-HUB-EUS2`)

SUBNET/ZONE	CIDR	USABLE IP ADDRESSES
`IB-UntrustZone`	`10.240.0.0/24`	251
`IB-TrustZone`	`10.240.1.0/24`	251
`OB-UntrustZone`	`10.240.2.0/24`	251
`OB-TrustZone`	`10.240.3.0/24`	251
`GatewaySubnet`	`10.240.10.0/24`	251

Subnets in the `East US 2` development network (`VNET-DEV-EUS2`)

The development team uses the development virtual network as a production pilot area. It has three subnets.

SUBNET	CIDR	ADDRESSES	IN SUBNET
`DEV-FE-EUS2`	`10.245.16.0/22`	1019	Front ends/web-tier VMs
`DEV-APP-EUS2`	`10.245.20.0/22`	1019	Application-tier VMs
`DEV-DB-EUS2`	`10.245.24.0/23`	507	Database VMs

Subnets in the `East US 2` production network (`VNET-PROD-EUS2`)

Azure IaaS components are located in the production network. Each application tier has its own subnet. Subnets match those in the development network, with the addition of a subnet for domain controllers.

SUBNET	CIDR	ADDRESSES	IN SUBNET
`PROD-FE-EUS2`	`10.245.32.0/22`	1019	Front ends/web-tier VMs
`PROD-APP-EUS2`	`10.245.36.0/22`	1019	Application-tier VMs
`PROD-DB-EUS2`	`10.245.40.0/23`	507	Database VMs
`PROD-DC-EUS2`	`10.245.42.0/24`	251	Domain controller VMs

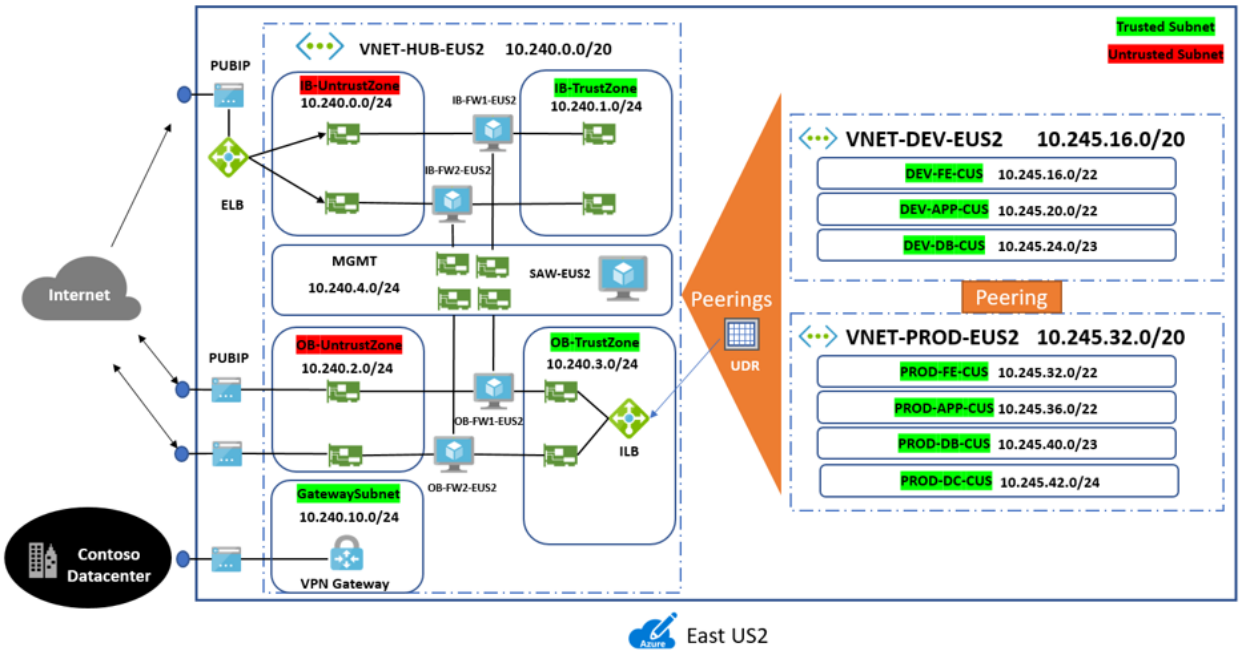


Figure 20: Hub network architecture.

Virtual networks in `Central US` (secondary region)

`Central US` is Contoso's secondary region. Here's how Contoso will architect networks within it:

- **Hub:** The hub virtual network in `Central US` is considered the secondary point of connectivity to the on-premises datacenter. The spoke virtual networks in `Central US` can be used to isolate workloads if necessary, managed separately from other spokes.
- **Virtual networks:** Contoso will have two virtual networks in `Central US`:
  - `VNET-PROD-CUS`: This is a production network and can be thought of as a secondary hub.
  - `VNET-ASR-CUS`: This virtual network will act as a location in which VMs are created after failover from on-premises or as a location for Azure VMs failed over from the primary to the secondary region. This network is similar to the production networks but without any domain controllers on it.

  Each virtual network in the region will have its own address space without overlap. Contoso will configure routing without NAT.
- **Subnets:** The subnets will be designed in a similar way to those in `East US 2`.

The following table summarizes virtual networks in `Central US`.

VIRTUAL NETWORK	RANGE	PEER
`VNET-HUB-CUS`	`10.250.0.0/20`	`VNET-HUB-EUS2`, `VNET-ASR-CUS`, `VNET-PROD-CUS`
`VNET-ASR-CUS`	`10.255.16.0/20`	`VNET-HUB-CUS`, `VNET-PROD-CUS`
`VNET-PROD-CUS`	`10.255.32.0/20`	`VNET-HUB-CUS`, `VNET-ASR-CUS`, `VNET-PROD-EUS2`

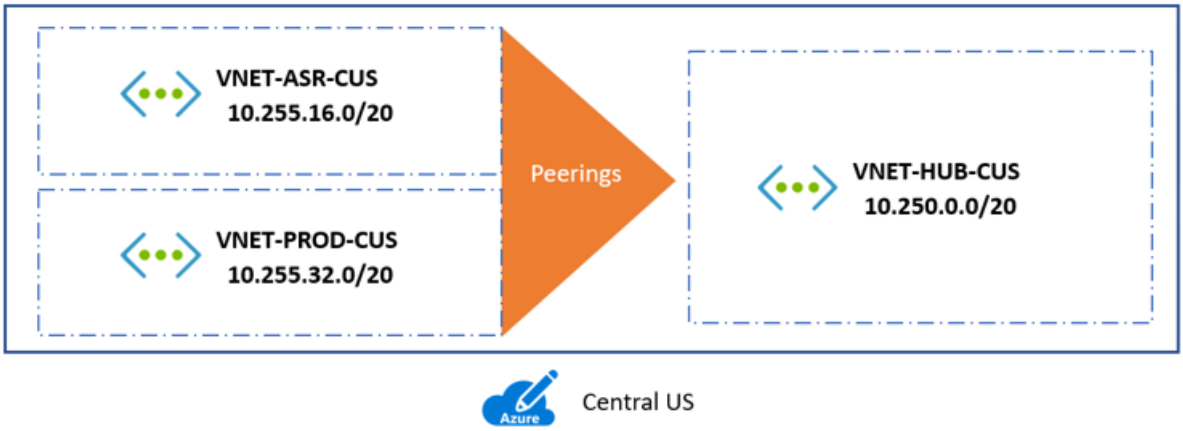


Figure 21: A hub-and-spoke model in a paired region.

Subnets in the `Central US` hub network (`VNET-HUB-CUS`)

SUBNET	CIDR	USABLE IP ADDRESSES
`IB-UntrustZone`	`10.250.0.0/24`	251
`IB-TrustZone`	`10.250.1.0/24`	251
`OB-UntrustZone`	`10.250.2.0/24`	251
`OB-TrustZone`	`10.250.3.0/24`	251
`GatewaySubnet`	`10.250.10.0/24`	251

Subnets in the `Central US` production network (`VNET-PROD-CUS`)

In parallel with the production network in the primary region (`East US 2`), there's a production network in the secondary region (`Central US`).

SUBNET	CIDR	ADDRESSES	IN SUBNET
`PROD-FE-CUS`	`10.255.32.0/22`	1019	Front ends/web-tier VMs
`PROD-APP-CUS`	`10.255.36.0/22`	1019	Application-tier VMs
`PROD-DB-CUS`	`10.255.40.0/23`	507	Database VMs
`PROD-DC-CUS`	`10.255.42.0/24`	251	Domain controller VMs

Subnets in the `Central US` failover/recovery network (`VNET-ASR-CUS`)

The `VNET-ASR-CUS` network is used for failover between regions. Site Recovery will be used to replicate and fail over Azure VMs between the regions. It also functions as a Contoso datacenter to the Azure network for protected workloads that remain on-premises but fail over to Azure for disaster recovery.

`VNET-ASR-CUS` is the same basic subnet as the production virtual network in `East US 2` but without the need for a domain controller subnet.

SUBNET	CIDR	ADDRESSES	IN SUBNET
`ASR-FE-CUS`	`10.255.16.0/22`	1019	Front ends/web-tier VMs

SUBNET	CIDR	ADDRESSES	IN SUBNET
`ASR-APP-CUS`	`10.255.20.0/22`	1019	Application-tier VMs
`ASR-DB-CUS`	`10.255.24.0/23`	507	Database VMs

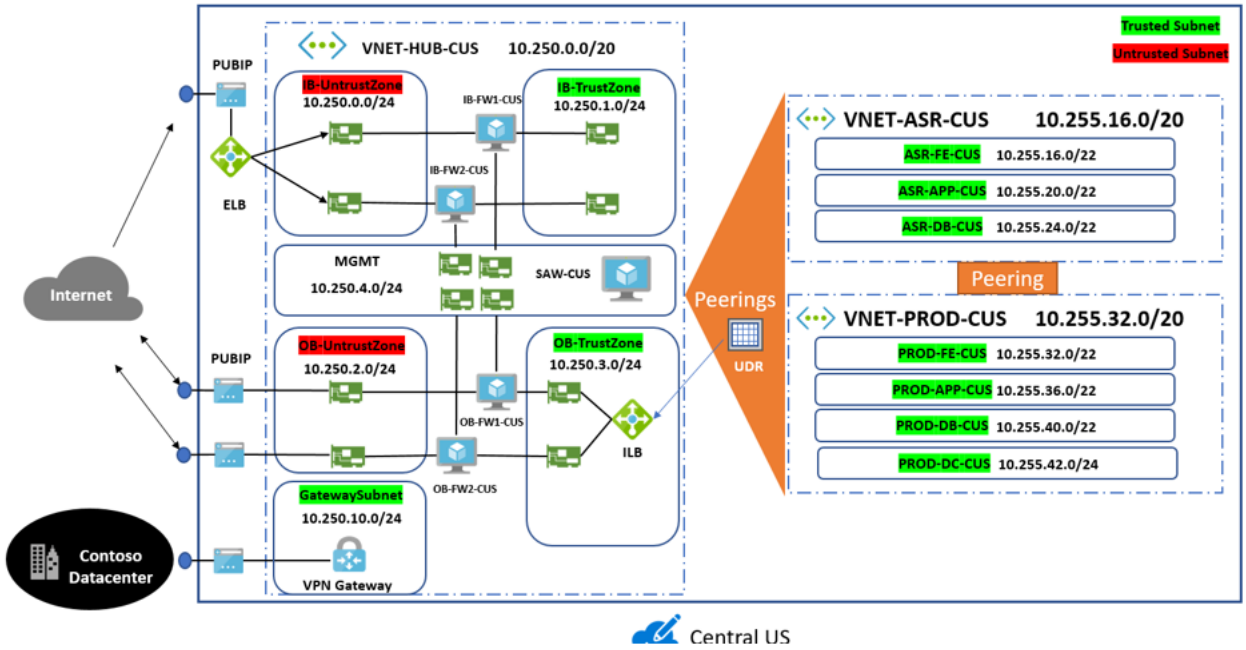


Figure 22: Hub network architecture.

Configure peered connections

The hub in each region will be peered to the hub in the other region and to all virtual networks within the hub region. This configuration allows for hubs to communicate and to view all virtual networks within a region. Note that peering creates a two-sided connection. One is from the initiating peer on the first virtual network, and the other is on the second virtual network.

In a hybrid deployment, traffic that passes between peers needs to be visible from the VPN connection between the on-premises datacenter and Azure. To enable this, Contoso must use specific settings on peered connections. For any connections from spoke virtual networks through the hub to the on-premises datacenter, Contoso needs to allow traffic to be forwarded and to cross the VPN gateways.

Domain controller

For the domain controllers in the `VNET-PROD-EUS2` network, Contoso wants traffic to flow both between the `EUS2` hub/production network and over the VPN connection to on-premises. To do this, Contoso admins must allow the following:

1. **Allow forwarded traffic** and **Allow gateway transit configurations** on the peered connection. In our example, this would be the connection from `VNET-HUB-EUS2` to `VNET-PROD-EUS2`.

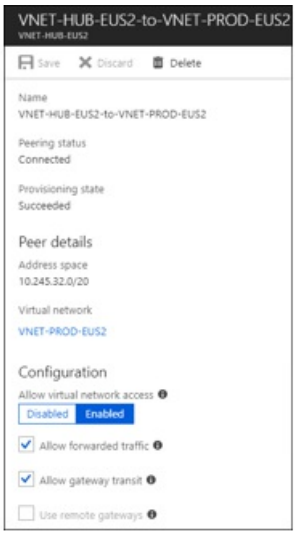


Figure 23: A peered connection.

2. **Allow forwarded traffic** and **Use remote gateways** on the other side of the peering, on the connection from `VNET-PROD-EUS2` to `VNET-HUB-EUS2`.

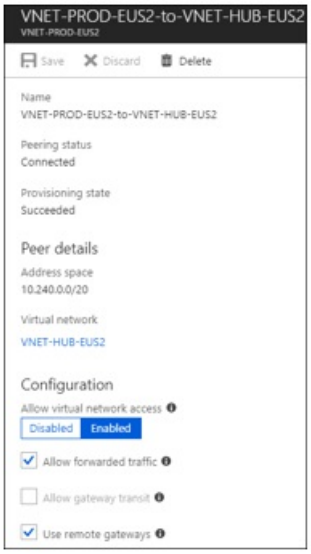


Figure 24: A peered connection.

3. On-premises, they set up a static route that directs the local traffic to route across the VPN tunnel to the virtual network. The configuration is completed on the gateway that provides the VPN tunnel from Contoso to Azure. They use routing and remote access service (RRAS) for the static route.

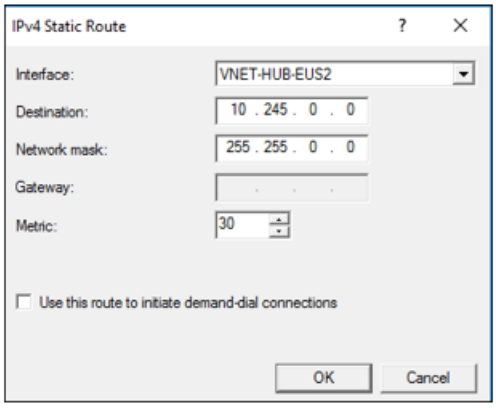


Figure 25: A peered connection.

Production networks

A spoked peer network can't see a spoked peer network in another region via a hub. For Contoso's production networks in both regions to see each other, Contoso admins need to create a direct peered connection for `VNET-PROD-EUS2` and `VENT-PROD-CUS`.

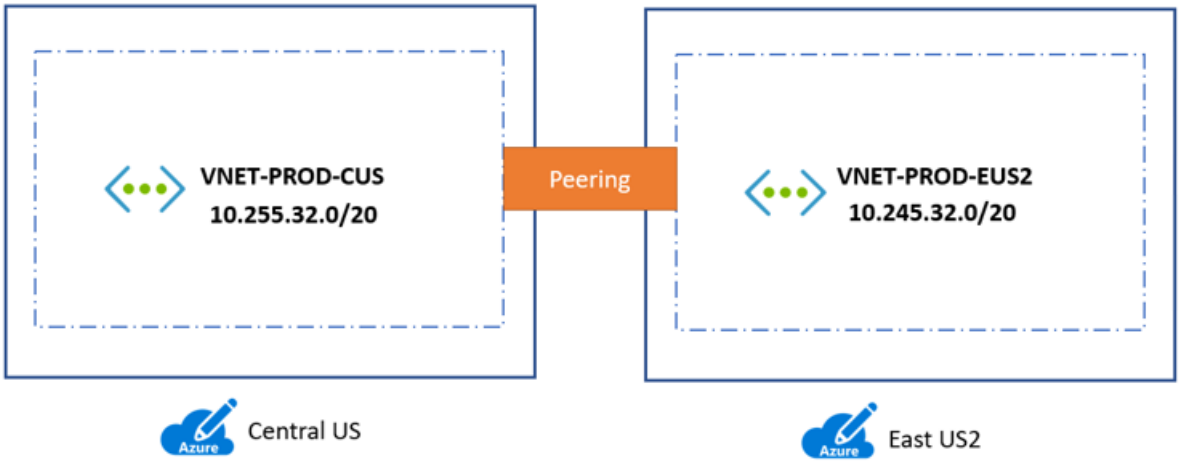


Figure 26: Creating a direct peered connection.

Set up DNS

When you deploy resources in virtual networks, you have a couple of choices for domain name resolution. You can use name resolution provided by Azure or provide DNS servers for resolution. The type of name resolution that you use depends on how your resources need to communicate with each other. Get more information about the Azure DNS service.

Contoso admins have decided that the Azure DNS service isn't a good choice in the hybrid environment. Instead, they'll use the on-premises DNS servers. Here are the details:

- Because this is a hybrid network, all VMs on-premises and in Azure need to be able to resolve names to function properly. This means that custom DNS settings must be applied to all the virtual networks.
- Contoso currently has domain controllers (DCs) deployed in the Contoso datacenter and at the branch offices. The primary DNS servers are `contosodc1` (`172.16.0.10`) and `contosodc2` (`172.16.0.1`).
- After the virtual networks are deployed, the on-premises domain controllers are configured as DNS servers in the networks.
- If an optional custom DNS is specified for the virtual network, the virtual IP address `168.63.129.16` for the recursive resolvers in Azure must be added to the list. To do this, Contoso configures DNS server settings on each virtual network. For example, the custom DNS settings for the `VNET-HUB-EUS2` network would be as follows:

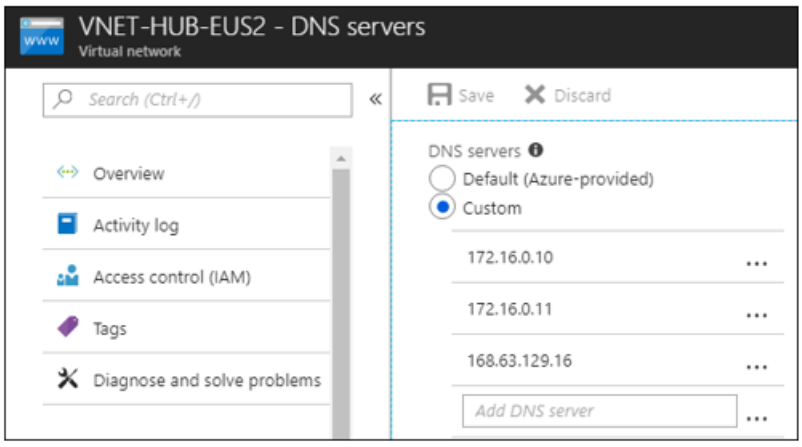


Figure 27: A custom DNS.

In addition to the on-premises domain controllers, Contoso will implement four domain controllers to support the Azure networks (two for each region):

REGION	DC	VIRTUAL NETWORK	SUBNET	IP ADDRESS
East US 2	contosodc3	VNET-PROD-EUS2	PROD-DC-EUS2	10.245.42.4
East US 2	contosodc4	VNET-PROD-EUS2	PROD-DC-EUS2	10.245.42.5
Central US	contosodc5	VNET-PROD-CUS	PROD-DC-CUS	10.255.42.4
Central US	contosodc6	VNET-PROD-CUS	PROD-DC-CUS	10.255.42.4

After deploying the on-premises domain controllers, Contoso needs to update the DNS settings on networks on either region to include the new domain controllers in the DNS server list.

Set up domain controllers in Azure

After updating network settings, Contoso admins are ready to build out the domain controllers in Azure.

1. In the Azure portal, they deploy a new Windows Server VM to the appropriate virtual network.
2. They create availability sets in each location for the VM. Availability sets ensure that the Azure fabric separates the VMs into different infrastructures in the Azure region. Availability sets also allow Contoso to be eligible for the 99.95 percent service-level agreement (SLA) for VMs in Azure.

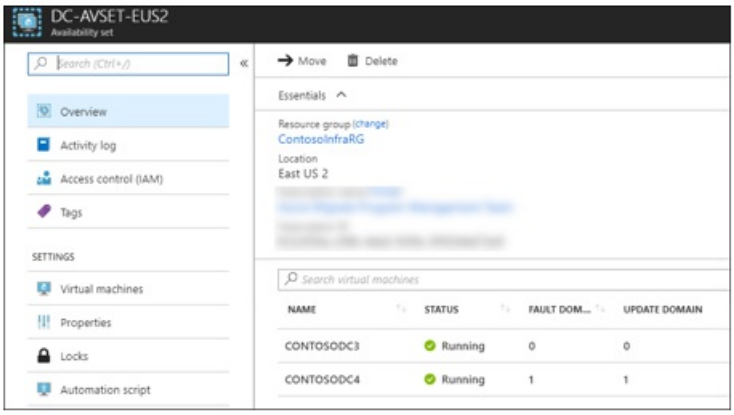


Figure 28: An availability set.

3. After the VM is deployed, they open the network interface for the VM. They set the private IP address to static and specify a valid address.

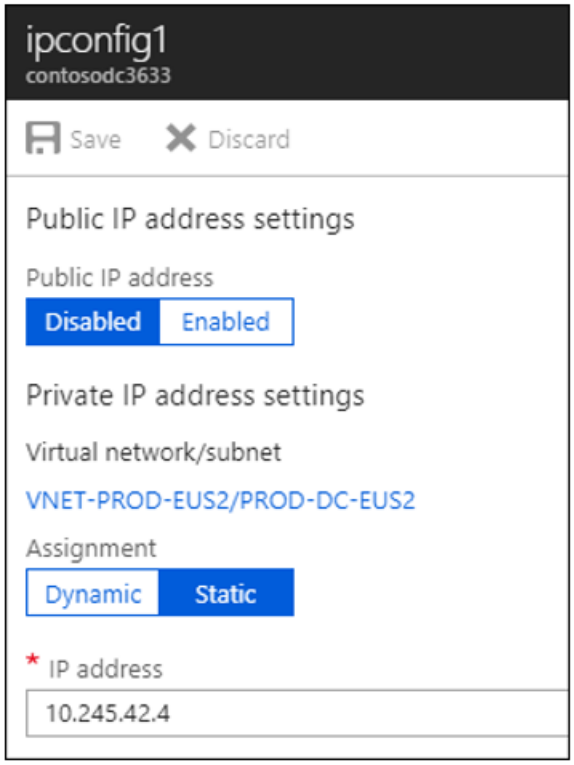


Figure 29: A VM NIC.

4. They attach a new data disk to the VM. This disk contains the Active Directory database and the sysvol share.

   The size of the disk will determine the number of IOPS that it supports. Over time, the disk size might need to increase as the environment grows.

   > **NOTE**
   > The disk shouldn't be set to read/write for host caching. Active Directory databases don't support this.

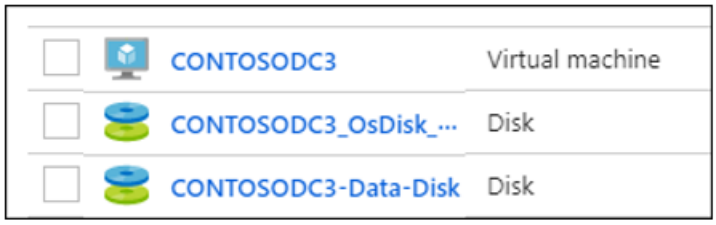


Figure 30: An Active Directory disk.

5. After the disk is added, they connect to the VM over Remote Desktop Services and open Server Manager.
6. In **File and Storage Services**, they run the New Volume Wizard. They ensure that the drive is assigned the letter F or above on the local VM.

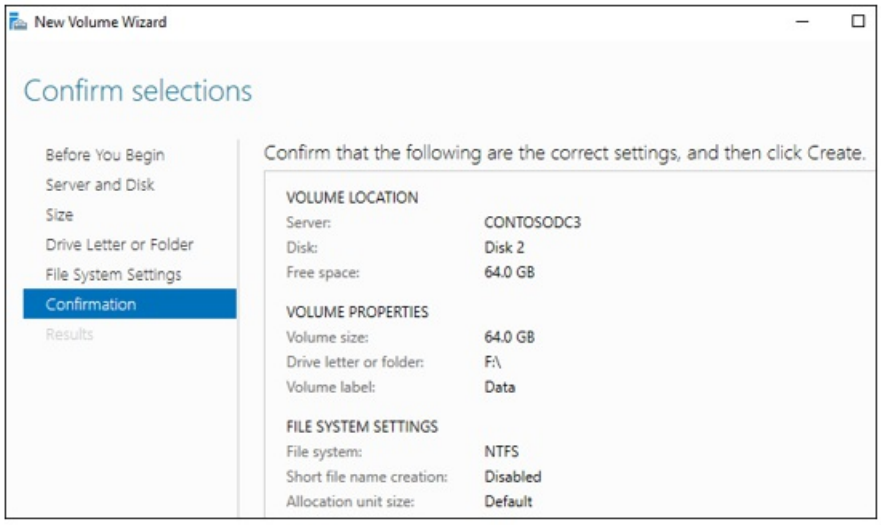


Figure 31: New Volume Wizard.

7. In Server Manager, they add the **Active Directory Domain Services** role. Then, they configure the VM as a domain controller.

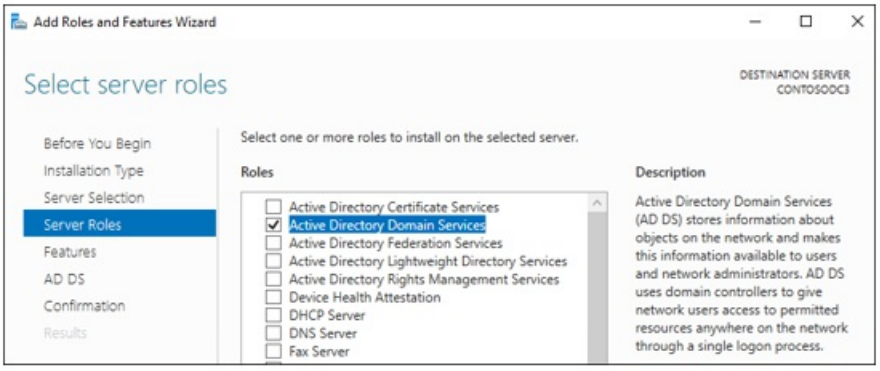


Figure 32: Adding the server role.

8. After the VM is configured as a DC and restarted, they open DNS manager and configure the Azure DNS resolver as a forwarder. This allows the DC to forward DNS queries it can't resolve in the Azure DNS.

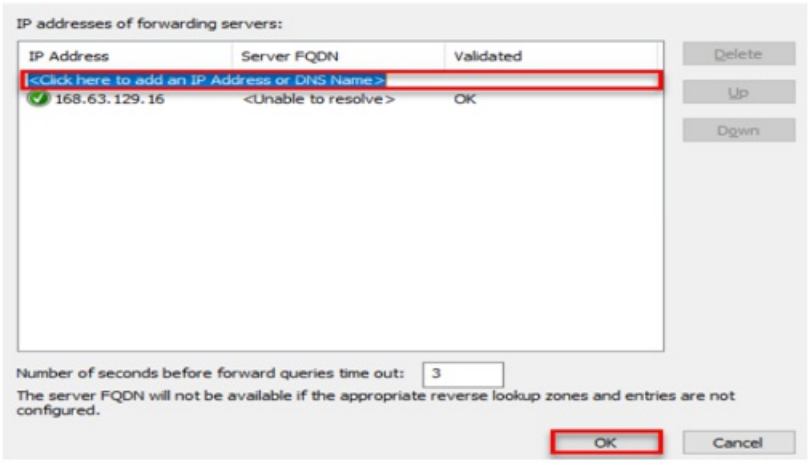


Figure 33: Configuring the Azure DNS resolver.

9. They update the custom DNS settings for each virtual network with the appropriate domain controller for the virtual network region. They include on-premises DCs in the list.

Set up Active Directory

Active Directory is a critical service for a network and must be configured correctly. Contoso admins will build

Active Directory sites for the Contoso datacenter and for the `East US 2` and `Central US` regions.

1. They create two new sites (`AZURE-EUS2` and `AZURE-CUS`) along with the datacenter site (`contoso-datacenter`).
2. After creating the sites, they create subnets in the sites, to match the virtual networks and datacenter.

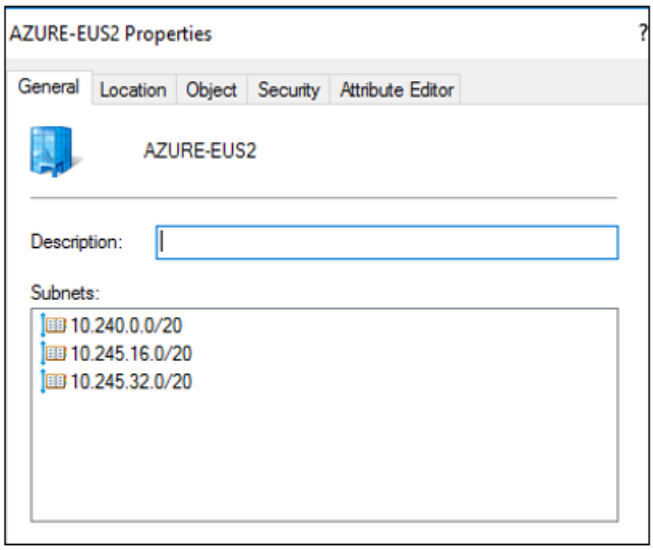


Figure 34: Datacenter subnets.

3. They create two site links to connect everything. The domain controllers should then be moved to their location.

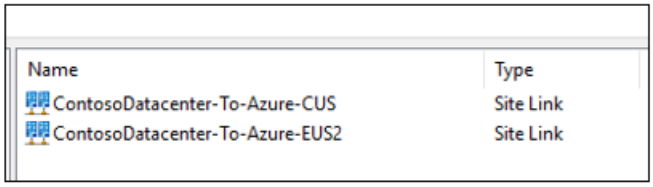


Figure 35: Datacenter links.

4. They confirm that the Active Directory replication topology is in place.

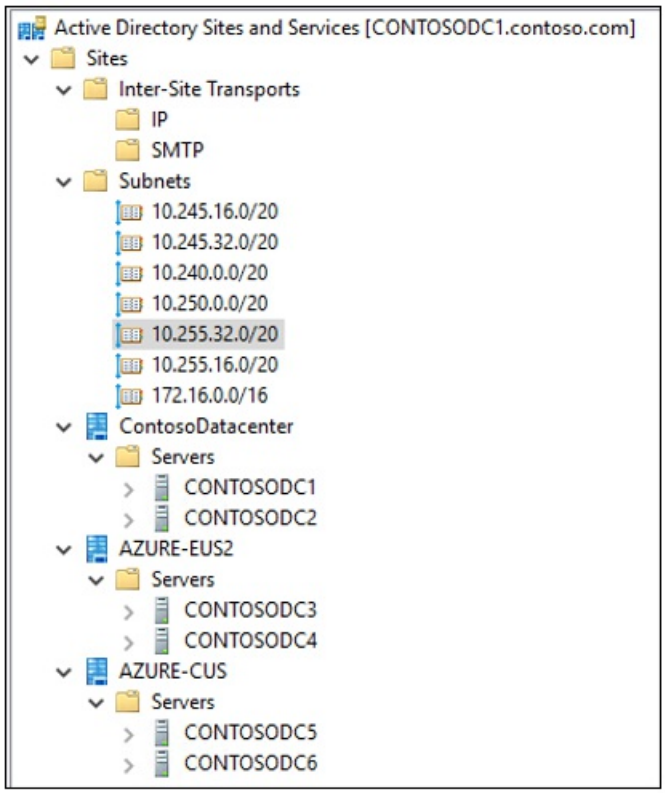


Figure 36: Datacenter replication.

With everything complete, a list of the domain controllers and sites is shown in the on-premises Active Directory Administrative Center.

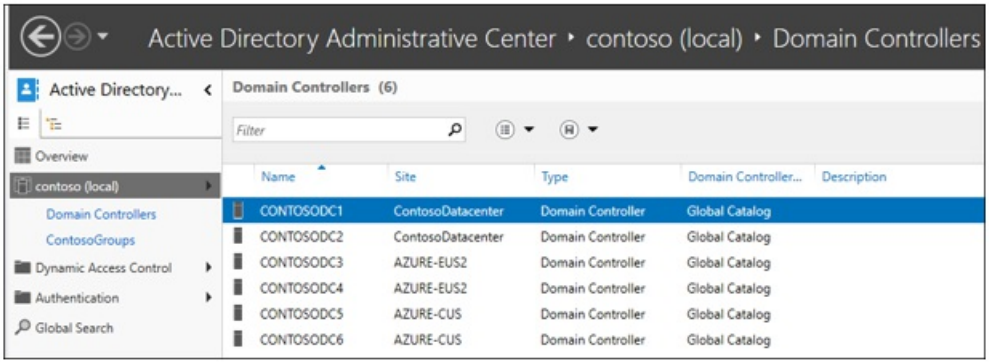


Figure 37: The Active Directory Administrative Center.

Step 5: Plan for governance

Azure provides a range of governance controls across services and the Azure platform. For more information, see the Azure governance options.

As it configures identity and access control, Contoso has already begun to put some aspects of governance and security in place. Broadly, it needs to consider three areas:

- **Policy:** Azure Policy applies and enforces rules and effects over your resources, so the resources comply with corporate requirements and SLAs.
- **Locks:** Azure allows you to lock subscriptions, resource groups, and other resources so that they can be modified only by those with permissions.
- **Tags:** Resources can be controlled, audited, and managed with tags. Tags attach metadata to resources, providing information about resources or owners.

Set up policies

The Azure Policy service evaluates your resources by scanning for those not compliant with policy definitions. For example, you might have a policy that only allows certain types of VMs or requires resources to have a specific tag.

Policies specify a policy definition, and a policy assignment specifies the scope in which a policy should be applied. The scope can range from a management group to a resource group. Learn how to create and manage policies.

Contoso wants to begin two policies. It wants a policy to ensure that resources can be deployed in the `East US 2` and `Central US` regions only. It also wants a policy to limit VM SKUs to approved SKUs only. The intention is to ensure that expensive VM SKUs aren't used.

Limit resources to regions

Contoso uses the built-in policy definition **Allowed locations** to limit resource regions.

1. In the Azure portal, select **All services**, and search for **Policy**.
2. Select **Assignments** > **Assign policy**.
3. In the policy list, select **Allowed locations**.
4. Set **Scope** to the name of the Azure subscription, and select the two regions in the allowlist.

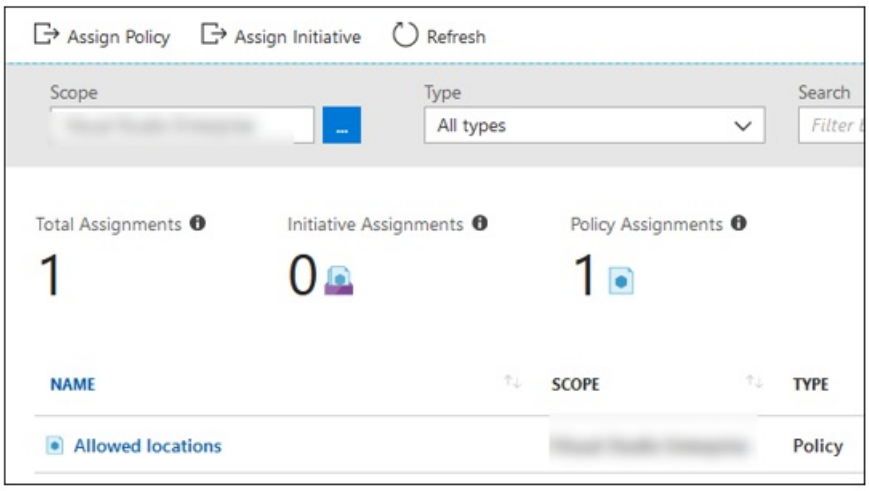


Figure 38: Allowed locations defined via policy.

5. By default, the policy is set with **Deny**. This setting means that if someone starts a deployment in the subscription that isn't in either the `East US 2` or `Central US` region, the deployment will fail. Here's what happens if someone in the Contoso subscription tries to set up a deployment in `West US`.

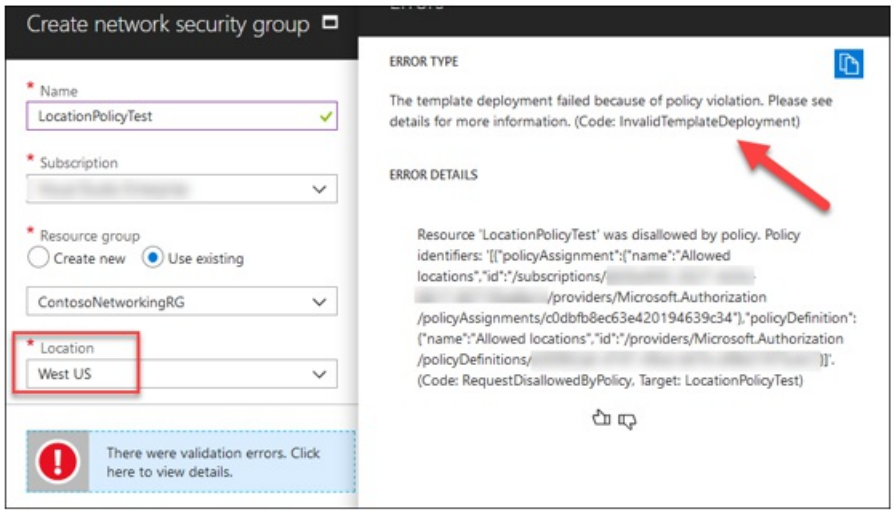


Figure 39: A failed policy.

Allow specific VM SKUs

Contoso will use the built-in policy definition `Allow virtual machine SKUs` to limit the types of VMs that can be created in the subscription.

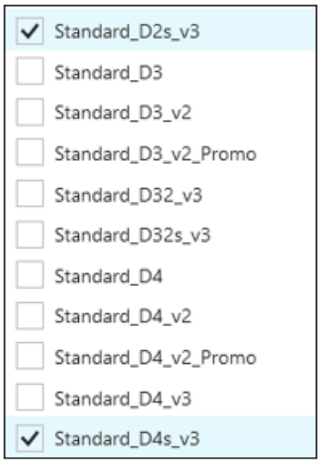


Figure 40: A policy SKU.

Check policy compliance

Policies go into effect immediately, and Contoso can check resources for compliance. In the Azure portal, select the **Compliance** link. The compliance dashboard appears. You can drill down for more details.

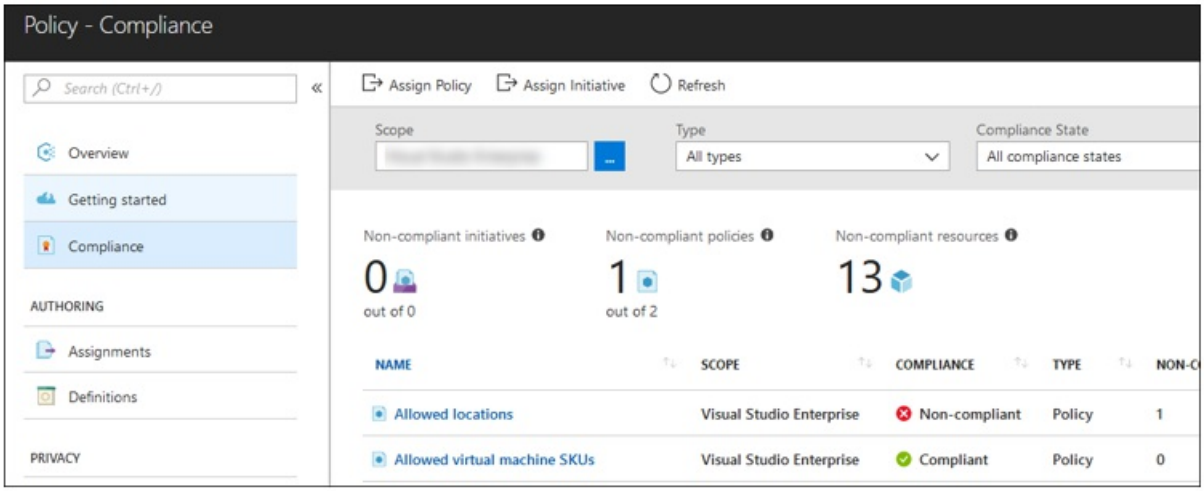


Figure 41: Policy compliance.

Set up locks

Contoso has long been using the ITIL framework for the management of its systems. One of the most important aspects of the framework is change control, and Contoso wants to make sure that change control is implemented in the Azure deployment.

Contoso will lock resources. Any production or failover component must be in a resource group that has a read-only lock. This means that to modify or delete production items, authorized users must remove the lock. Nonproduction resource groups will have `CanNotDelete` locks. This means that authorized users can read or modify a resource but can't delete it.

Set up tagging

To track resources as they're added, it will be increasingly important for Contoso to associate resources with an appropriate department, customer, and environment. In addition to providing information about resources and owners, tags will enable Contoso to aggregate and group resources and to use that data for chargeback purposes.

Contoso needs to visualize its Azure assets in a way that makes sense for the business, such as by role or department. Note that resources don't need to reside in the same resource group to share a tag. Contoso will create a tag taxonomy so that everyone uses the same tags.

TAG NAME	VALUE
`CostCenter`	12345: It must be a valid cost center from SAP.
`BusinessUnit`	Name of the business unit (from SAP). Matches `CostCenter`.
`ApplicationTeam`	Email alias of the team that owns support for the application.
`CatalogName`	Name of the application or `SharedServices`, according to the service catalog that the resource supports.
`ServiceManager`	Email alias of the ITIL Service Manager for the resource.
`COBPriority`	Priority set by the business for BCDR. Values of 1-5.
`ENV`	`DEV`, `STG`, and `PROD` are the allowed values, representing development, staging, and production.

For example:

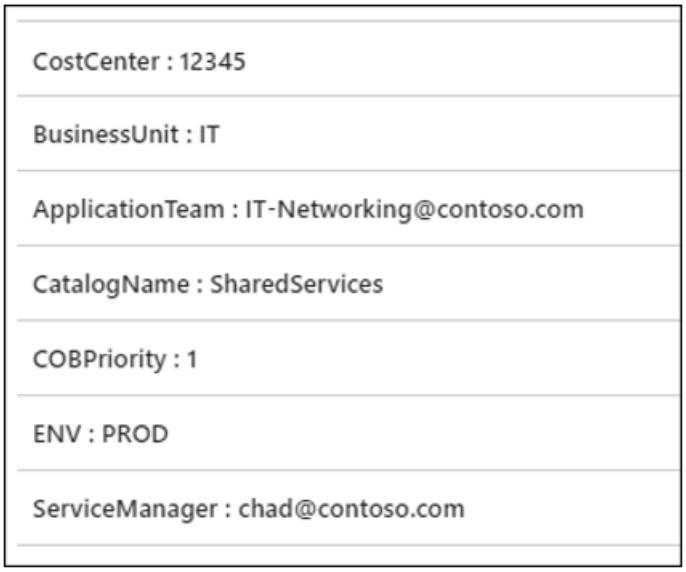


Figure 42: Azure tags.

After creating the tag, Contoso will go back and create new policy definitions and assignments to enforce the use of the required tags across the organization.

Step 6: Consider security

Security is crucial in the cloud, and Azure provides a wide array of security tools and capabilities. These help you to create secure solutions on the secure Azure platform. See Trust your cloud to learn more about Azure security.

There are a few aspects for Contoso to consider:

- Azure Security Center provides unified security management and Microsoft Defender for Identity across hybrid cloud workloads. Use it to apply security policies across your workloads, limit your exposure to threats, and detect and respond to attacks.
- A network security group (NSG) filters network traffic based on a list of security rules that allow or deny network traffic to resources connected to virtual networks in Azure.
- Azure Disk Encryption is a capability that helps you encrypt your Windows and Linux IaaS VM disks.

Work with the Azure Security Center

Contoso is looking for a quick view into the security posture of its new hybrid cloud, and specifically, its Azure workloads. As a result, Contoso has decided to implement Azure Security Center starting with the following features:

- Centralized policy management
- Continuous assessment
- Actionable recommendations

Centralize policy management

With centralized policy management, Contoso will ensure compliance with security requirements by centrally managing security policies across the entire environment. It can simply and quickly implement a policy that applies to all of its Azure resources.

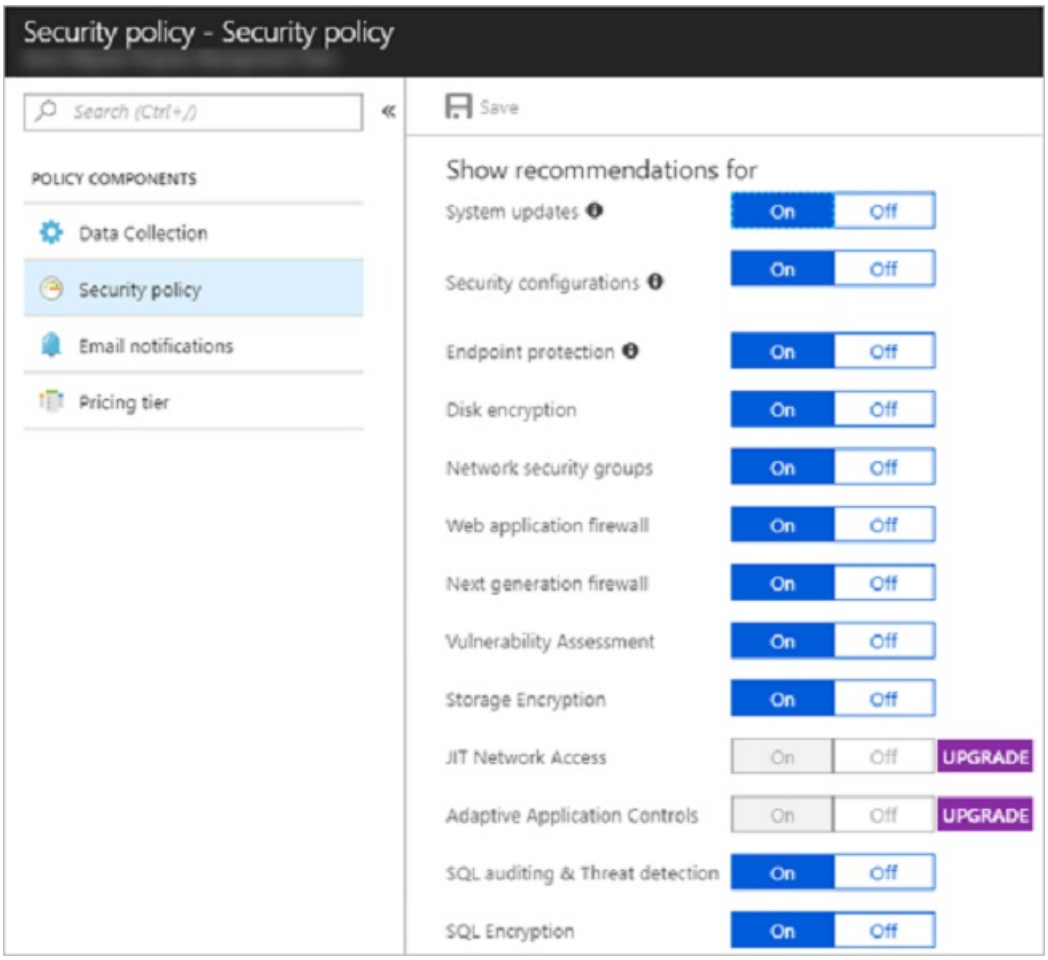


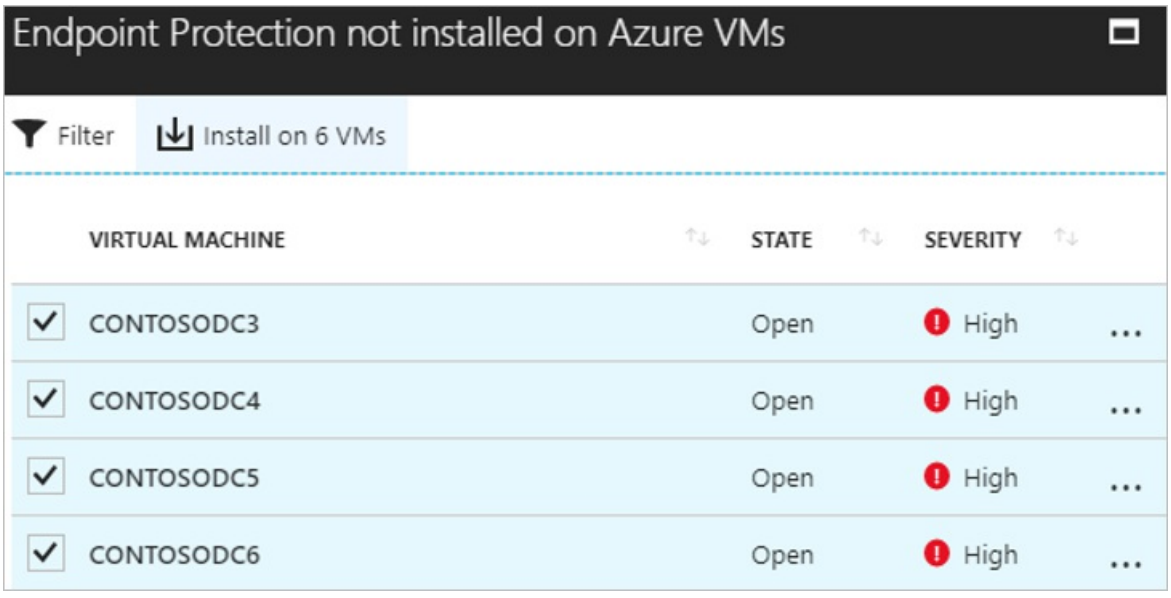
Figure 43: A security policy.

Assess security

Contoso will take advantage of the continuous security assessment that monitors the security of machines, networks, storage, data, and applications to discover potential security issues.

Security Center analyzes the security state of the Contoso compute, infrastructure, and data resources. It also analyzes the security state of Azure apps and services. Continuous assessment helps the Contoso operations team to discover potential security issues, such as systems with missing security updates or exposed network ports.

Contoso wants to make sure all of the VMs are protected. Security Center helps with this. It verifies VM health, and it makes prioritized and actionable recommendations to remediate security vulnerabilities before they're exploited.



Figure

44: Monitoring.

Work with NSGs

Contoso can limit network traffic to resources in a virtual network by using network security groups.

A network security group contains a list of security rules that allow or deny inbound or outbound network traffic based on source or destination IP address, port, and protocol. When applied to a subnet, rules are applied to all resources in the subnet. In addition to network interfaces, this includes instances of Azure services deployed in the subnet.

Application security groups (ASGs) enable you to configure network security as a natural extension of an application structure. You can then group VMs and define network security policies based on those groups.

Contoso can use ASGs to reuse the security policy at scale without manual maintenance of explicit IP addresses. The platform handles the complexity of explicit IP addresses and multiple rule sets, so the organization can focus on business logic. Contoso can specify an ASG as the source and destination in a security rule. After a security policy is defined, Contoso can create VMs and assign the VM NICs to a group.

Contoso will implement a mix of NSGs and ASGs. Contoso is concerned about NSG management. It's also worried about the overuse of NSGs and the added complexity for operations staff. Here's what Contoso will do:

- All traffic into and out of all subnets (north/south) will be subject to an NSG rule, except for the gateway subnets in the hub networks.
- Any firewalls or domain controllers will be protected by both subnet NSGs and NIC NSGs.
- All production applications will have ASGs applied.

Contoso has built a model of how this security configuration will look for its applications.

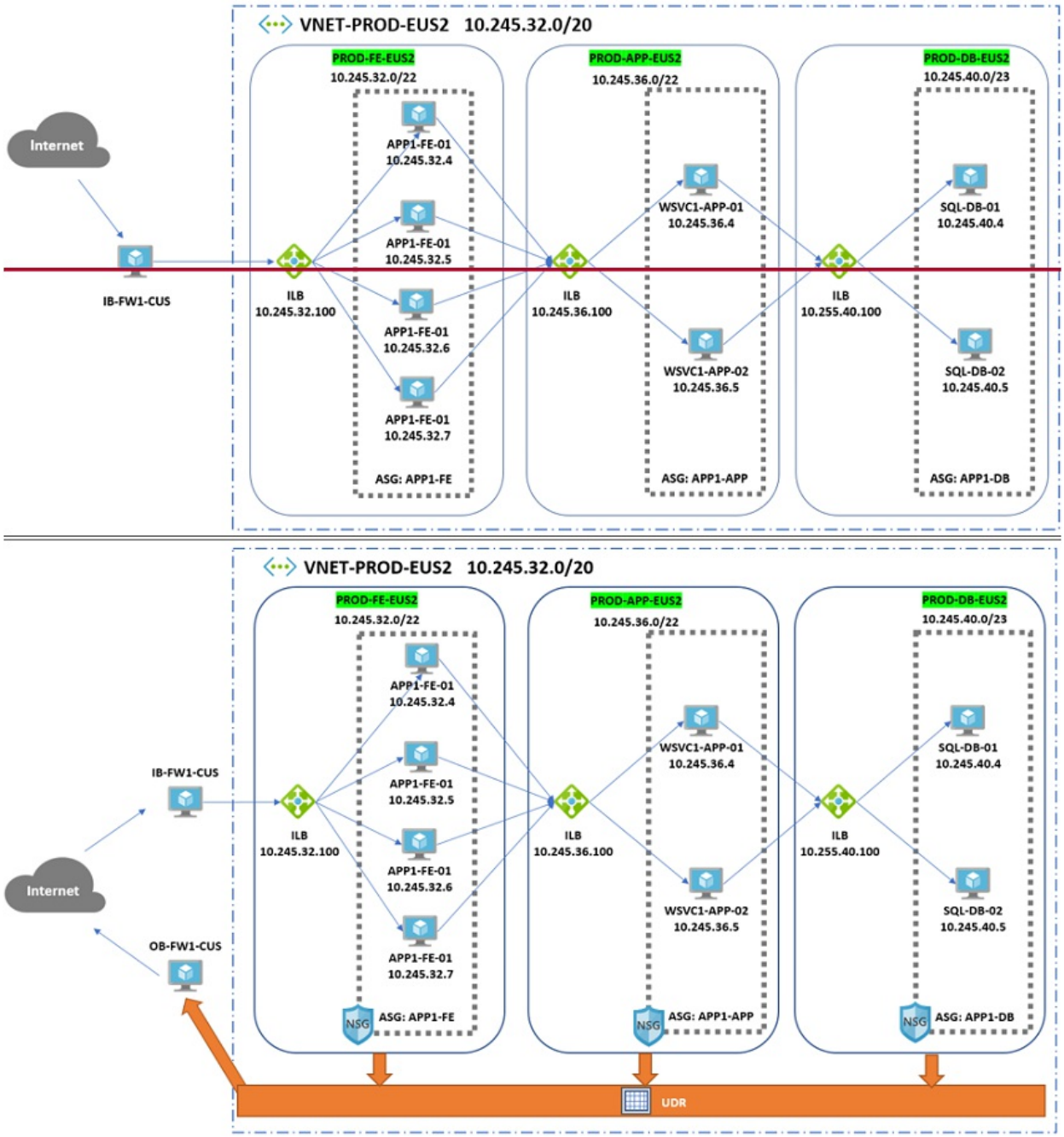


Figure 45: Security model.

The NSGs associated with the ASGs will be configured with least privilege to ensure that only allowed packets can flow from one part of the network to its destination.

ACTION	NAME	SOURCE	TARGET	PORT
`Allow`	`AllowInternetToFE`	`VNET-HUB-EUS1` / `IB-TrustZone`	`APP1-FE`	80, 443
`Allow`	`AllowWebToApp`	`APP1-FE`	`APP1-APP`	80, 443
`Allow`	`AllowAppToDB`	`APP1-APP`	`APP1-DB`	1433
`Deny`	`DenyAllInbound`	Any	Any	Any

Encrypt data

Azure Disk Encryption integrates with Azure Key Vault to help control and manage the disk-encryption keys and secrets for a subscription. It ensures that all data on VM disks is encrypted at rest in Azure Storage.

Contoso has determined that specific VMs require encryption. Contoso will apply encryption to VMs with customer, confidential, or personal data.

Conclusion

In this article, Contoso set up an Azure infrastructure and policy for Azure subscription, hybrid identify, disaster recovery, network, governance, and security.

Not every step taken here is required for a cloud migration. In this case, Contoso planned a network infrastructure that can handle all types of migrations while being secure, resilient, and scalable.

Next steps

After setting up its Azure infrastructure, Contoso is ready to begin migrating workloads to the cloud. See the migration patterns and examples overview for a selection of scenarios that use this sample infrastructure as a migration target.

Best practices to cost and size workloads migrated to Azure

5/26/2021 • 17 minutes to read • Edit Online

As you plan and design for an Azure migration, focusing on costs can help to ensure its long-term success. During a migration project, it's critical that all teams (such as finance, management, and application development teams) understand the costs associated with this process.

- Before migration, it's important to have a baseline for monthly, quarterly, and yearly budget targets in order to estimate the amount you'd spend on your migration and ensure its success.
- After migration, you should optimize costs, continually monitor workloads, and plan for future usage patterns. Migrated resources might start out as one type of workload but because of usage, costs, and shifting business requirements, shift to another type over time.

This article describes best practices for preparing for and managing cost and size, both before and after migration.

> **IMPORTANT**
> The best practices and opinions described in this article are based on Azure platform and service features available at the time of writing. Features and capabilities change over time. Not all recommendations might be applicable for your deployment, so select what works for you.

Before migration

Before you move your workloads to the cloud, estimate the monthly cost of running them in Azure. Proactively managing cloud costs helps you adhere to your operating expense budget. If budget is limited, take this into account before migration. Consider converting workloads to Azure serverless technologies, where appropriate, to reduce costs.

The best practices in this section help you to:

- Estimate costs.
- Perform right-sizing for virtual machines (VMs) and storage.
- Use Azure Hybrid Benefit.
- Use Azure Reserved VM Instances.
- Estimate cloud spending across subscriptions.

Best practice: Estimate monthly workload costs

You can use several tools to forecast your monthly bill for migrated workloads.

- **Azure pricing calculator:** Select the products you want to estimate, such as VMs and storage. Then, input costs into the calculator to build an estimate.

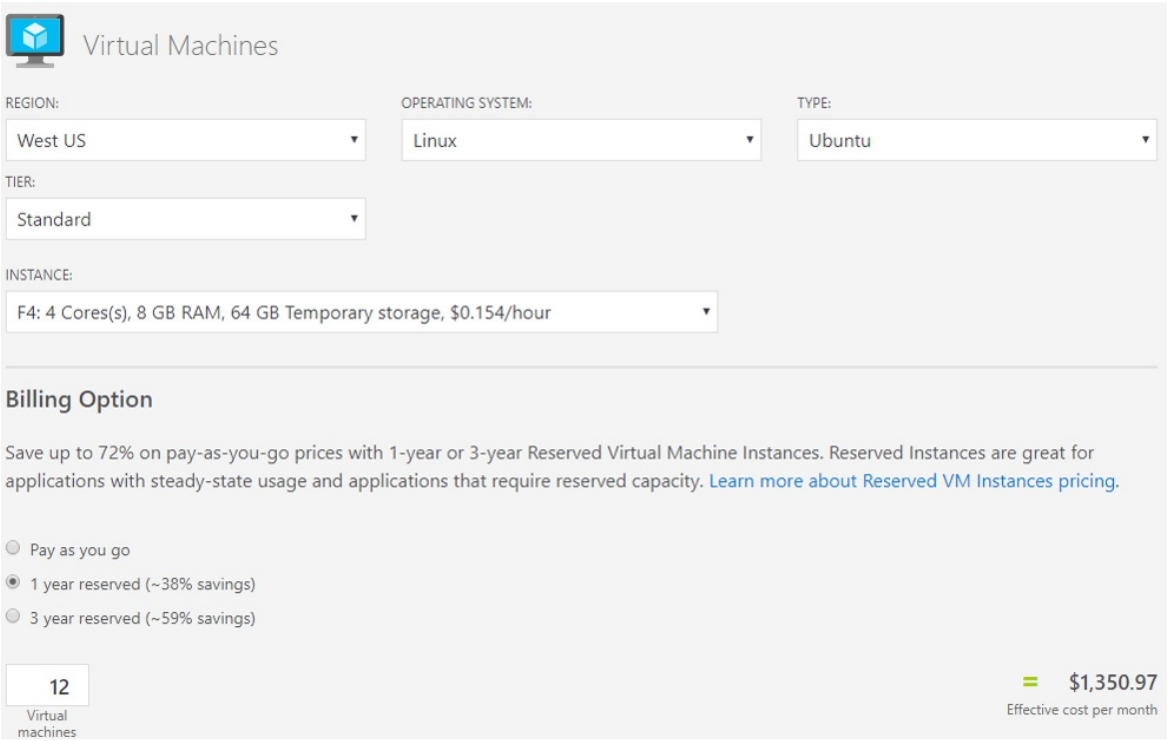


Figure 1: Azure pricing calculator.

- **Azure Migrate:** To estimate costs, you need to review and account for all the resources required to run your workloads in Azure. To acquire this data, you create inventory of your assets, including servers, VMs, databases, and storage. You can use Azure Migrate to collect this information.
  - Azure Migrate discovers and assesses your on-premises environment to provide an inventory.
  - Azure Migrate can map and show you dependencies between VMs, so that you have a complete picture.
  - An Azure Migrate assessment contains estimated cost.
    - **Compute costs:** Using the Azure VM size recommended when you create an assessment, Azure Migrate uses the Azure Billing APIs to calculate estimated monthly VM costs. The estimation considers the operating system, Software Assurance, Azure Reserved VM Instances, VM uptime, location, and currency settings. It aggregates the cost across all VMs in the assessment, and calculates a total monthly compute cost.
    - **Storage cost:** Azure Migrate calculates total monthly storage costs by aggregating the storage costs of all VMs in an assessment. You can calculate the monthly storage cost for a specific machine by aggregating the monthly cost of all disks attached to it.

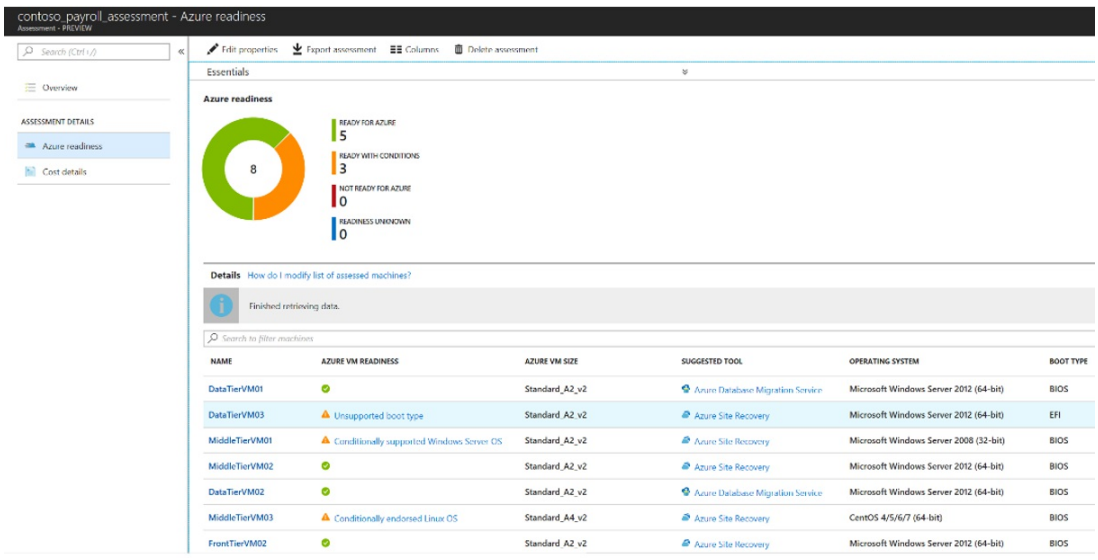


Figure 2: Azure Migrate assessment.

Learn more:

- Use the Azure pricing calculator.
- Read the Azure Migrate overview.
- Read about Azure Migrate assessments.
- Learn more about Azure Database Migration Service.

Best practice: Right-size VMs

You can choose various options when you deploy Azure VMs to support workloads. Each VM type has specific features and different combinations of CPU, memory, and disks. VMs are grouped as shown in the following table:

TYPE	DETAILS	USAGE
General-purpose	Balanced CPU-to-memory.	Good for testing and development, small- to medium-sized databases, and low- to medium-volume traffic web servers.
Compute-optimized	High CPU-to-memory.	Good for medium-volume traffic web server, network appliances, batch processes, and application servers.
Memory-optimized	High memory-to-CPU.	Good for relational databases, medium- to large-size cache, and in-memory analytics.
Storage optimized	High disk throughput and I/O.	Good for big data, and SQL and NoSQL databases.
GPU optimized	Specialized VMs. Single or multiple GPUs.	Heavy graphics and video editing.
High performance	Fastest and most powerful CPU. VMs with optional high-throughput network interfaces (RDMA).	Critical high-performance applications.

- It's important to understand the pricing differences between these VMs, and the long-term budget effects.
- Each type has several VM series within it.
- Additionally, when you select a VM within a series, you can only scale the VM up and down within that series. For example, a `DS2_v2` instance can scale up to `DS4_v2`, but it can't be changed to an instance of a different series, such as a `F2s_v2` instance.

Learn more:

- Learn more about VM types and sizing, and map sizes to types.
- Plan sizes for VM instances.
- Review a sample assessment for the fictional Contoso company.

Best practice: Select the right storage

Tuning and maintaining on-premises storage (SAN or NAS), and the networks to support them, can be costly and time-consuming. File (storage) data is commonly migrated to the cloud to help alleviate operational and

management headaches. Microsoft provides several options for moving data to Azure, and you need to make decisions about those options. Picking the right storage type for data can save your organization several thousands of dollars every month. Here are a few considerations:

- Data that isn't accessed much and isn't business-critical doesn't need to be placed on the most expensive storage.
- Conversely, important business-critical data should be located on higher tier storage options.
- During migration planning, take an inventory of data and classify it by importance, in order to map it to the most suitable storage. Consider budget and costs, as well as performance. Cost shouldn't necessarily be the main factor. Picking the least expensive option might expose the workload to performance and availability risks.

Storage data types

Azure provides different types of storage data.

DATA TYPE	DETAILS	USAGE
Blobs	Optimized to store massive amounts of unstructured objects, such as text or binary data.	Access data from everywhere over HTTP/HTTPS. Use for streaming and random access scenarios. For example, to serve images and documents directly to a browser, stream video and audio, and store backup and disaster recovery data.
Files	Managed file shares accessed over SMB 3.0.	Use when migrating on-premises file shares and to provide multiple access/connections to file data.
Disks	Based on page blobs. Disk type: standard (HDD or SSD) or premium (SSD). Disk management: unmanaged (you manage disk settings and storage) or managed (you select the disk type and Azure manages the disk for you).	Use premium disks for VMs. Use managed disks for simple management and scaling.
Queues	Store and retrieve large numbers of messages accessed via authenticated calls (HTTP or HTTPS).	Connect application components with asynchronous message queueing.
Tables	Store tables.	This data type is part of Azure Cosmos DB Table API.

Access tiers

Azure Storage provides different options for accessing block blob data. Selecting the right access tier helps ensure that you store block blob data in the most cost-effective manner.

ACCESS TIER	DETAILS	USAGE

ACCESS TIER	DETAILS	USAGE
Hot	Higher storage cost than cool. Lower access charges than cool. This is the default tier.	Use for data in active use, that's accessed frequently.
Cool	Lower storage cost than hot. Higher access charges than hot. Store for minimum of 30 days.	Store short-term. Data is available but accessed infrequently.
Archive	Used for individual block blobs. Most cost-effective option for storage. Data access is more expensive than hot and cold.	Use for data that can tolerate several hours of retrieval latency and will remain in the tier for at least 180 days.

Storage account types

Azure provides different types of storage accounts and performance tiers.

ACCOUNT TYPE	DETAILS	USAGE
General-purpose v2 standard	Supports blobs (block, page, and append), files, disks, queues, and tables. Supports hot, cool, and archive access tiers. Zone-redundant storage (ZRS) is supported.	Use for most scenarios and most types of data. Standard storage accounts can be HDD- or SSD-based.
General-purpose v2 premium	Supports Blob Storage data (page blobs). Supports hot, cool, and archive access tiers. ZRS is supported. Stored on SSD.	Microsoft recommends using for all VMs.
General-purpose v1	Access tiering isn't supported. Doesn't support ZRS.	Use if applications need the Azure classic deployment model.
Blob	Specialized storage account for storing unstructured objects. Provides block blobs and append blobs only (no file, queue, table, or disk storage services). Provides the same durability, availability, scalability and performance as general-purpose v2.	You can't store page blobs in these accounts, and therefore can't store VHD files. You can set an access tier to hot or cool.

Storage redundancy options

Storage accounts can use different types of redundancy for resilience and high availability.

TYPE	DETAILS	USAGE

TYPE	DETAILS	USAGE
Locally redundant storage (LRS)	Protects against a local outage by replicating within a single storage unit to a separate fault domain and update domain. Keeps multiple copies of your data in one datacenter. Provides at least 99.999999999 percent (eleven nines) durability of objects over a particular year.	Consider whether your application stores data that can be easily reconstructed.
Zone-redundant storage (ZRS)	Protects against a datacenter outage by replicating across three storage clusters in a single region. Each storage cluster is physically separated and located in its own Availability Zone. Provides at least 99.9999999999 percent (twelve nines) durability of objects over a particular year, by keeping multiple copies of your data across multiple datacenters or regions.	Consider whether you need consistency, durability, and high availability. Might not protect against a regional disaster, when multiple zones are permanently affected.
Geo-redundant storage (GRS)	Protects against an entire region outage, by replicating data to a secondary region hundreds of miles away from the primary. Provides at least 99.99999999999999 percent (sixteen nines) durability of objects over a particular year.	Replica data isn't available unless Microsoft initiates a failover to the secondary region. If failover occurs, read and write access is available.
Read-access geo-redundant storage (RA-GRS)	Similar to GRS. Provides at least 99.99999999999999 percent (sixteen nines) durability of objects over a particular year.	Provides 99.99 percent read availability, by allowing read access from the second region used for GRS.

Learn more:

- Review Azure Storage pricing.
- Learn about Azure Import/Export.
- Compare blobs, files, and disk storage data types.
- Learn more about access tiers.
- Review different types of storage accounts.
- Learn about Azure Storage redundancy, including LRS, ZRS, GRS, and read-access GRS.
- Learn more about Azure Files.

Best practice: Take advantage of Azure Hybrid Benefit

A portfolio that integrates on-premises Microsoft software with Azure can provide you with competitive and cost advantages. If you currently have an operating system or other software licensing through Software Assurance, you can take those licenses with you to the cloud, with Azure Hybrid Benefit.

Learn more:

- Take a look at the Azure Hybrid Benefit savings calculator.
- Learn more about Azure Hybrid Benefit for Windows Server.
- Review pricing guidance for SQL Server Azure VMs.

Best practice: Use reserved VM instances

Most cloud platforms use a pay-as-you-go payment model. This model presents disadvantages, because you don't necessarily know how dynamic your workloads will be. When you specify clear intentions for a workload, you contribute to infrastructure planning.

When you use Azure Reserved VM Instances, you prepay for a one-year or three-year term for VM instances.

- Prepayment provides a discount on the resources you use.
- You can significantly reduce VM, Azure SQL Database compute, Azure Cosmos DB, or other resource costs that compare to pay-as-you-go prices.
- Reservations provide a billing discount, and don't affect the runtime state of your resources.
- You can cancel reserved instances.

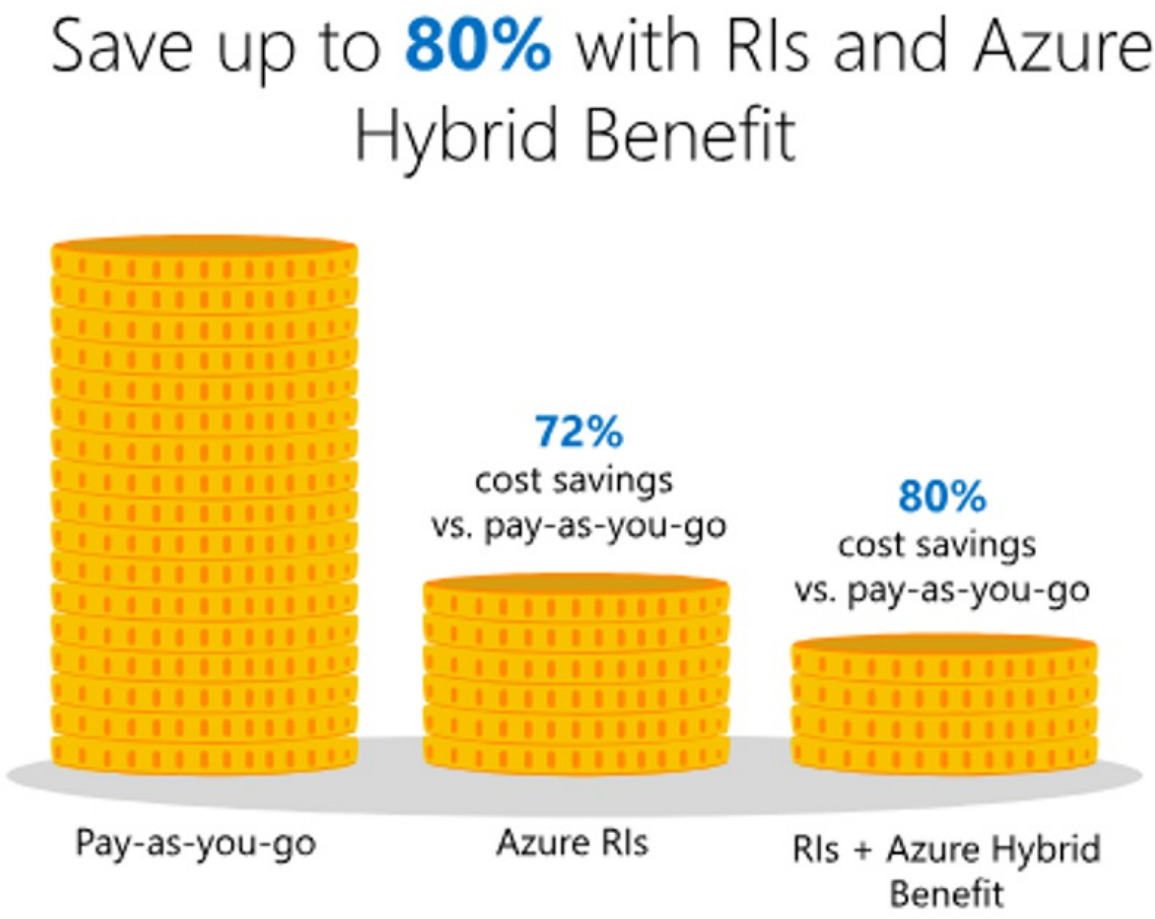


Figure 3: Azure Reserved VM Instances.

Learn more:

- Learn about Azure Reservations.
- Read the Azure Reserved VM Instances FAQ.
- Review pricing guidance for SQL Server on Azure VMs.

Best practice: Aggregate cloud spending across subscriptions

Eventually, it's very likely that you'll have more than one Azure subscription. For example, you might need an additional subscription to separate development and production boundaries, or you might have a platform that requires a separate subscription for each client. Having the ability to aggregate data reporting across all the subscriptions into a single platform is a valuable feature.

To do this, you can use Azure Cost Management + Billing APIs. Then, after aggregating data into a single source

such as Azure SQL Database, you can use tools like Power BI to surface the aggregated data. You can create aggregated subscription reports, and granular reports. For example, for users who need proactive insights into cost management, you can create specific views of costs, based on department, resource group, or other information. You don't need to provide them with full access to Azure billing data.

Learn more:

- Read the Azure Consumption APIs overview.
- Learn how to connect to Azure Consumption Insights in Power BI Desktop.
- Learn how to manage access to billing information for Azure by using Azure role-based access control (Azure RBAC).

After migration

After a successful migration of your workloads and a few weeks of collecting consumption data, you'll have a clear idea of resources costs. As you analyze data, you can start to generate a budget baseline for Azure resource groups and resources. Then, as you understand where your cloud budget is being spent, you can analyze how to further reduce your costs.

Best practice: Use Azure Cost Management + Billing

Microsoft provides Azure Cost Management + Billing to help you track spending. This service:

- Helps you to monitor and control Azure spending, and optimize use of resources.
- Reviews your entire subscription and all of its resources, and makes recommendations.
- Provides you with a full API to integrate external tools and financial systems for reporting.
- Tracks resource usage and helps you manage cloud costs with a single, unified view.
- Provides rich operational and financial insights to help you make informed decisions.

With Azure Cost Management + Billing, you can:

- Create a budget for financial accountability.
    - You can account for the services you consume or subscribe to for a specific period (monthly, quarterly, or annually), and a scope (subscriptions or resource groups). For example, you can create an Azure subscription budget for a monthly, quarterly, or annual period.
        - After you create a budget, it's shown in a cost analysis. Viewing your budget against current spending is important when you're analyzing your costs and spending.
    - You can choose to have email notifications sent when your budget thresholds are reached.
    - You can export costs management data to Azure Storage to analyze it.

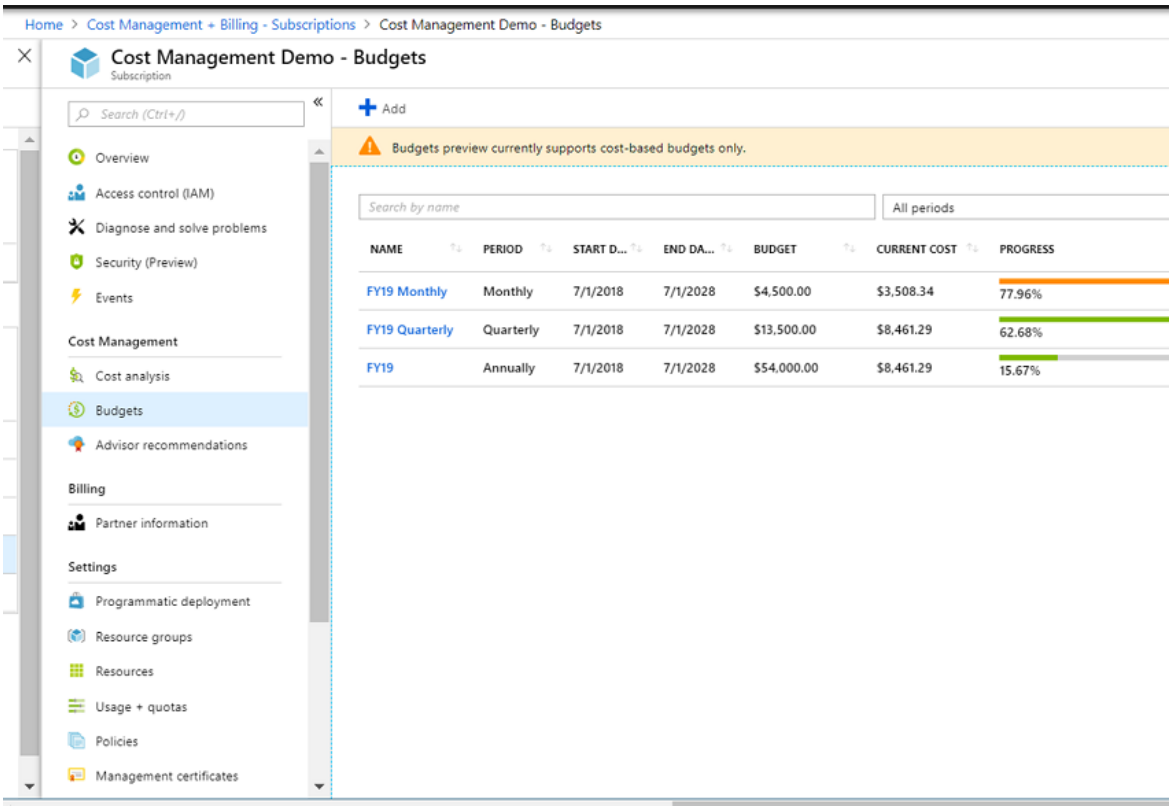


Figure 4: An Azure Cost Management + Billing budget.

- Perform a cost analysis to explore and analyze your organizational costs, to understand how costs are accrued, and identify spending trends.
    - Cost analysis is available to Enterprise Agreement users.
    - You can view cost analysis data for various scopes, including by department, account, subscription, or resource group.
    - You can get a cost analysis that shows total costs for the current month, and accumulated daily costs.

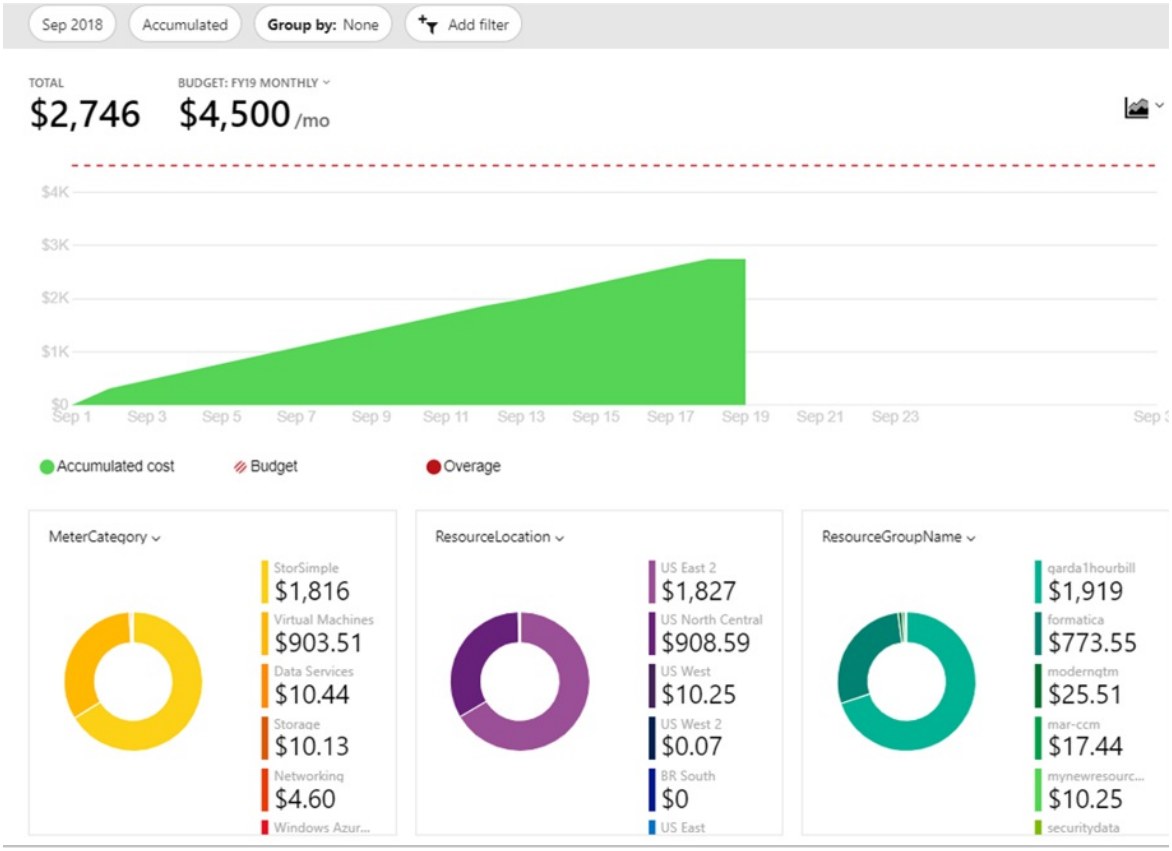


Figure 5: Azure Cost Management + Billing analysis.

- Get Azure Advisor recommendations that show you how you can optimize and improve efficiency.

Learn more:

- Read the Azure Cost Management + Billing overview.
- Learn to optimize your cloud investment with Azure Cost Management + Billing.
- Learn about Azure Cost Management + Billing reports.
- Get a tutorial on optimizing costs from recommendations.
- Review the Azure Consumption APIs.

Best practice: Monitor resource utilization

In Azure you pay for what you use, when resources are consumed, and you don't pay when they aren't. For VMs, billing occurs when a VM is allocated, and you aren't charged after a VM is deallocated. With this in mind, you should monitor VMs in use, and verify VM sizing.

Continually evaluate your VM workloads to determine baselines. For example, if your workload is used heavily Monday through Friday, 8 AM to 6 PM, but hardly used outside those hours, you can downgrade VMs outside peak times. This might mean changing VM sizes, or using virtual machine scale sets to autoscale VMs up or down. Some companies "snooze" VMs via a calendar that specifies when they should be available and when they're not needed.

You can monitor VM usage by using Microsoft tools, such as Azure Cost Management + Billing, Azure Monitor, and Azure Advisor. Third-party tools are also available.

> **NOTE**
> In addition to VM monitoring, you should monitor other networking resources, such as Azure ExpressRoute and virtual network gateways, for underuse and overuse.

Learn more:

- Read overviews of Azure Monitor and Azure Advisor.
- Get Azure Advisor cost recommendations.
- Learn how to optimize costs from recommendations, and prevent unexpected charges.
- Learn about the Azure Resource Optimization (ARO) toolkit.

Best practice: Implement resource group budgets

Often, you might find it useful to represent cost boundaries with resource groups. A resource group budget helps you track the costs associated with a resource group. You can trigger alerts and run a wide variety of playbooks when you reach or exceed your budget.

Learn more:

- Learn how to manage costs with Azure budgets.
- Review a tutorial on creating and managing an Azure budget.

Best practice: Optimize Azure Monitor retention

As you move resources into Azure and enable diagnostic logging for them, you generate a lot of log data. Typically, this log data is sent to a storage account that's mapped to a Log Analytics workspace. Here are a few tips for optimizing Azure Monitor retention:

- The longer the log data retention period, the more data you'll have.

- Not all log data is equal, and some resources will generate more log data than others.
- Due to regulations and compliance, it's likely that you'll need to retain log data for some resources longer than for others.
- You should walk a careful line between optimizing your log storage costs, and keeping the log data you need.
- We recommend evaluating and setting up the logging immediately after completing a migration so that you don't spend money on retaining nonessential logs.

Learn more:

- Learn about monitoring usage and estimated costs.

Best practice: Optimize storage

If you followed best practices for selecting storage before migration, you're probably reaping some benefits. But there can be additional storage costs that you can still optimize. Over time, blobs and files become stale. Data might not be used anymore, but regulatory requirements might mean that you need to keep it for a certain period. As such, you might not need to store it on the high-performance storage that you used for the original migration.

Identifying and moving stale data to cheaper storage areas can have a huge impact on your monthly storage budget and cost savings. Azure provides many ways to help you identify and then store this stale data.

- Take advantage of access tiers for general-purpose v2 storage, moving less important data from the hot tier to the cool or archived tiers.
- Use StorSimple to help move stale data that's based on customized policies.

Learn more:

- Learn more about access tiers.
- Read the StorSimple overview.
- Review StorSimple pricing.

Best practice: Automate VM optimization

The ultimate goal of running a VM in the cloud is to maximize the CPU, memory, and disk that it uses. If you discover VMs that aren't optimized or have frequent periods when VMs aren't used, it makes sense to either shut them down or downscale them by using virtual machine scale sets.

You can optimize a VM with Azure Automation, virtual machine scale sets, auto-shutdown, and scripted or third-party solutions.

Learn more:

- Learn about vertical autoscaling.
- Schedule a VM autostart.
- Learn how to start or stop VMs off hours in Azure Automation.
- Get more information about Azure Advisor, and the Azure Resource Optimization (ARO) toolkit.

Best practices: Use Azure Logic Apps and runbooks with Budgets API

Azure provides a REST API that has access to your tenant billing information. You can use the Budgets API to integrate external systems and workflows that are triggered by metrics that you build from the API data. You can pull usage and resource data into your preferred data analysis tools. You can integrate the Budgets API with Azure Logic Apps and runbooks.

The Azure Resource Usage and RateCard APIs can help you accurately predict and manage your costs. The APIs

are implemented as a resource provider and are included in the APIs exposed by the Azure Resource Manager.

Learn more:

- Review the Azure Budgets API.
- Get insights into usage with the Azure Billing APIs.

Best practice: Implement serverless technologies

VM workloads are often migrated "as-is" to avoid downtime. Often, VMs can host tasks that are intermittent, run over a short period, or take up many hours. Examples include VMs that run scheduled tasks, such as Windows task scheduler or PowerShell scripts. When these tasks aren't running, you're absorbing VM and disk storage costs.

After migrating and thoroughly reviewing these types of tasks, you might consider migrating them to serverless technologies like Azure Functions or Azure Batch jobs. These solutions can cut costs, and you'd no longer need to manage and maintain the VMs.

Learn more:

- Learn about Azure Functions.
- Learn about Azure Batch.

Next steps

Review other best practices:

- Explore best practices for security and management after migrating workloads to Azure.
- Explore best practices for networking after migrating workloads to Azure.

Scale a migration to Azure

5/18/2021 • 20 minutes to read • Edit Online

This article demonstrates how the fictional company Contoso performs a migration at scale to Azure. The company considers how to plan and perform a migration of more than 3,000 workloads, 8,000 databases, and 10,000 virtual machines (VMs).

Business drivers

The IT leadership team has worked closely with business partners to understand what they want to achieve with this migration:

- **Address business growth.** Contoso is growing, causing pressure on on-premises systems and infrastructure.
- **Increase efficiency.** Contoso needs to remove unnecessary procedures, and streamline processes for developers and users. The business needs IT to be fast and not waste time or money, thus delivering faster on customer requirements.
- **Increase agility.** Contoso IT needs to be more responsive to the needs of the business. It must be able to react faster than the changes in the marketplace, to enable success in a global economy. It mustn't get in the way or become a business blocker.
- **Scale.** As the business grows successfully, the Contoso IT team must provide systems that are able to grow at the same pace.
- **Improve cost models.** Contoso wants to lessen capital requirements in the IT budget. Contoso wants to use cloud abilities to scale and reduce the need for expensive hardware.
- **Lower licensing costs.** Contoso wants to minimize cloud costs.

Migration goals

The Contoso cloud team has pinned down goals for this migration. It used these goals to determine the best migration method.

REQUIREMENTS	DETAILS
Move to Azure quickly	Contoso wants to start moving applications and VMs to Azure as quickly as possible.
Compile a full inventory	Contoso wants a complete inventory of all applications, databases, and VMs in the organization.
Assess and classify applications	Contoso wants to take full advantage of the cloud. As a default, Contoso assumes that all services will run as platform as a service (PaaS). Infrastructure as a service (IaaS) will be used where PaaS isn't appropriate.
Train and move to DevOps	Contoso wants to move to a DevOps model. Contoso will provide Azure and DevOps training and will reorganize teams as necessary.

After establishing goals and requirements, Contoso reviews the IT footprint and identifies the migration process.

Current deployment

Contoso has planned and set up an Azure infrastructure and tried out different proof-of-concept (POC) migration combinations as detailed in the preceding table. It's now ready to embark on a full migration to Azure at scale. Here's what Contoso wants to migrate.

ITEM	VOLUME	DETAILS
Workloads	> 3,000 applications	• Applications run on VMs. • Application platforms include Windows, SQL Server, and LAMP.
Databases	Approximately 8,500 databases	Databases include SQL Server, MySQL, and PostgreSQL.
VMs	> 35,000 VMs	VMs run on VMware hosts and are managed by vCenter servers.

Migration process

Now that Contoso has established business drivers and migration goals, it can align to the Migrate methodology. It can build on the application of migration waves and migration sprints to iteratively plan and execute migration efforts.

Plan

Contoso kicks off the planning process by discovering and assessing on-premises applications, data, and infrastructure. Here's what Contoso will do:

- Contoso needs to discover applications, map dependencies across applications, and decide on migration order and priority.
- As Contoso assesses, it will build out a comprehensive inventory of applications and resources. Along with the new inventory, Contoso will use and update these existing items:
  - The configuration management database (CMDB). It holds technical configurations for Contoso applications.
  - The service catalog. It documents the operational details of applications, including associated business partners and service-level agreements.

Discover applications

Contoso runs thousands of applications across a range of servers. In addition to the CMDB and service catalog, Contoso needs discovery and assessment tools.

The tools must provide a mechanism that can feed assessment data into the migration process. Assessment tools must provide data that helps build up an intelligent inventory of Contoso's physical and virtual resources. Data should include profile information and performance metrics.

When discovery is complete, Contoso should have a full inventory of assets and the metadata associated with them. The company will use this inventory to define the migration plan.

Identify classifications

Contoso identifies some common categories to classify assets in the inventory. These classifications are critical to Contoso's decision making for migration. The classification list helps to establish migration priorities and identify complex issues.

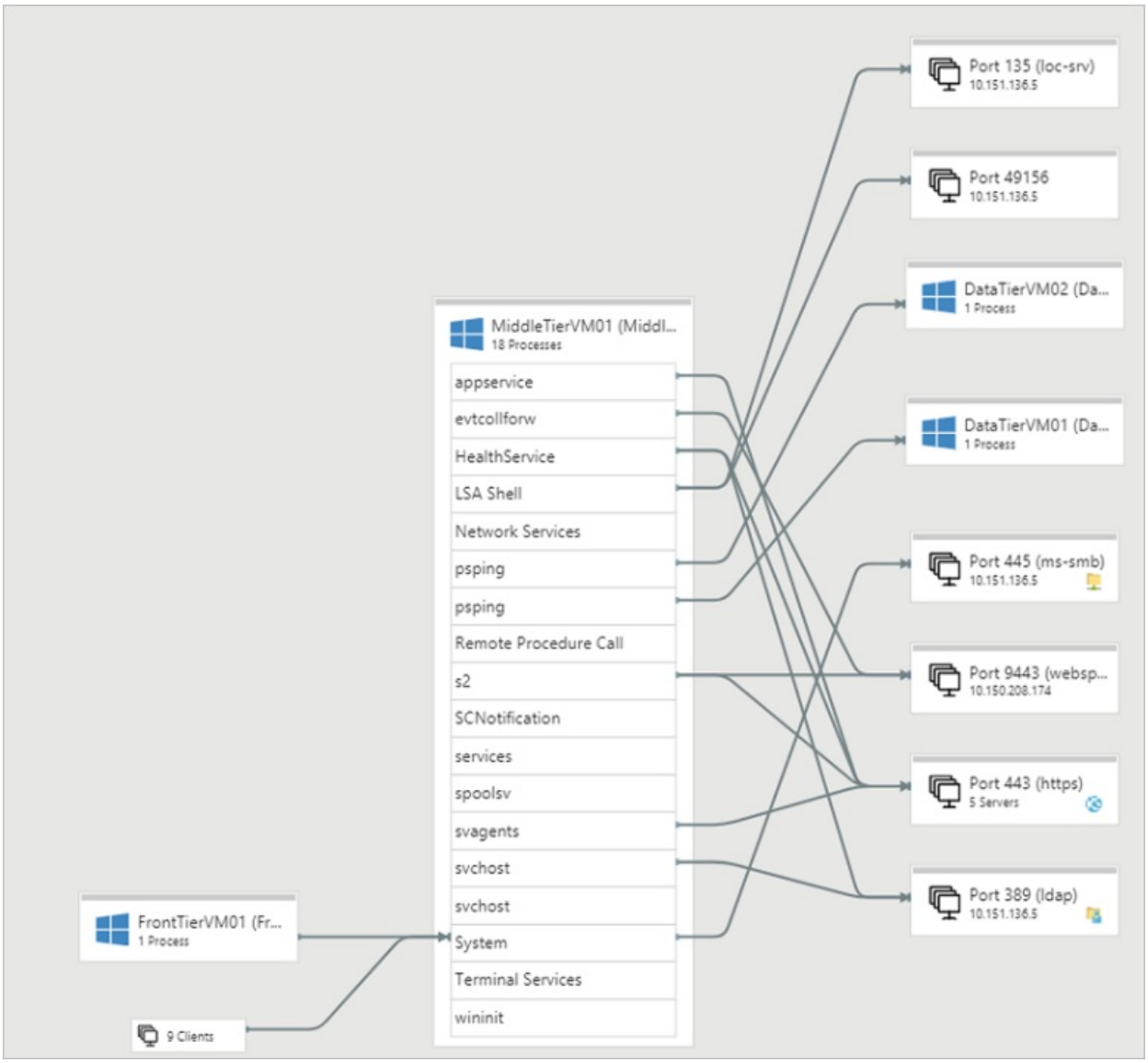
CATEGORY	ASSIGNED VALUE	DETAILS
Business group	List of business group names	Which group is responsible for the inventory item?
POC candidate	Y/N	Can the application be used as a POC or early adopter for cloud migration?
Technical debt	None/Some/Severe	Is the inventory item running or using an out-of-support product, platform, or operating system?
Firewall implications	Y/N	Does the application communicate with the internet or outside traffic? Does it integrate with a firewall?
Security issues	Y/N	Are there known security issues with the application? Does the application use unencrypted data or out-of-date platforms?

Discover application dependencies

As part of the assessment process, Contoso needs to identify where applications are running. It also needs to figure out the dependencies and connections between application servers. Contoso maps the environment in steps:

1. Contoso discovers how servers and machines map to individual applications, network locations, and groups.
2. Contoso clearly identifies applications that have few dependencies and are suitable for a quick migration.
3. Contoso can use mapping to help identify more complex dependencies and communications between application servers. Contoso can then group these servers logically to represent applications and plan a migration strategy based on these groups.

With mapping completed, Contoso can ensure that all application components are identified and accounted for when building the migration plan.



Evaluate applications

As the last step in the discovery and assessment process, Contoso can evaluate assessment and mapping results to figure out how to migrate each application in the service catalog.

To capture this evaluation process, Contoso adds a couple of classifications to the inventory.

CATEGORY	ASSIGNED VALUE	DETAILS
Business group	List of business group names	Which group is responsible for the inventory item?
POC candidate	Y/N	Can the application be used as a POC or early adopter for cloud migration?
Technical debt	None/Some/Severe	Is the inventory item running or using an out-of-support product, platform, or operating system?
Firewall implications	Y/N	Does the application communicate with the internet or outside traffic? Does it integrate with a firewall?

CATEGORY	ASSIGNED VALUE	DETAILS
Security issues	Y/N	Are there known security issues with the application? Does the application use unencrypted data or out-of-date platforms?
Migration strategy	Rehost/Refactor/Rearchitect/Rebuild	What kind of migration is needed for the application? How will the application be deployed in Azure? Learn more.
Technical complexity	1-5	How complex is the migration? This value should be defined by Contoso DevOps and relevant partners.
Business criticality	1-5	How important is the application for the business? For example, a small workgroup application might be assigned a score of one, while a critical application used across the organization might be assigned a score of five. This score will affect the migration priority level.
Migration priority	1/2/3	What's the migration priority for the application?
Migration risk	1-5	What's the risk level for migrating the application? Contoso DevOps and relevant partners should agree on this value.

Determine costs

To determine costs and the potential savings of Azure migration, Contoso can use the total cost of ownership (TCO) calculator to calculate and compare the TCO for Azure to a comparable on-premises deployment.

Identify assessment tools

Contoso decides which tool to use for discovery, assessment, and building the inventory. Contoso identifies a mix of Azure tools and services, native application tools and scripts, and partner tools. In particular, Contoso is interested in how Azure Migrate can be used to assess at scale.

Azure Migrate

The Azure Migrate service helps you to discover and assess on-premises VMware VMs, in preparation for migration to Azure. Here's what Azure Migrate does:

1. **Discover**: Discover on-premises VMware VMs.

   Azure Migrate supports discovery from multiple vCenter servers (serially) and can run discoveries in separate Azure Migrate projects.

   Azure Migrate performs discovery via a VMware VM running the Azure Migrate Collector. The same collector can discover VMs on different vCenter servers and send data to different projects.

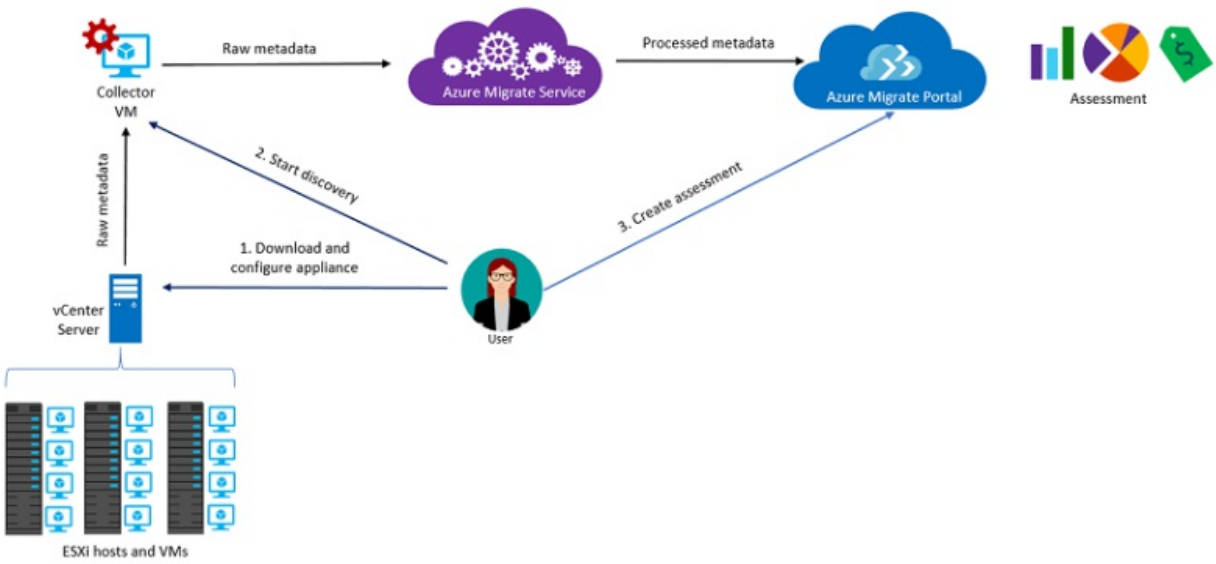
2. **Assess readiness**: Assess whether on-premises machines are suitable for running in Azure. An assessment includes:

   - **Size recommendations**: Get size recommendations for Azure VMs, based on the performance

history of on-premises VMs.

- **Estimated monthly costs:** Get estimated costs for running on-premises machines in Azure.

3. **Identify dependencies:** Visualize dependencies of on-premises machines to create optimal machine groups for assessment and migration.



Contoso needs to use Azure Migrate correctly, given the scale of this migration:

- Contoso assesses each application via Azure Migrate. This assessment ensures that Azure Migrate returns timely data to the Azure portal.
- Contoso admins learn how to deploy Azure Migrate at scale.
- Contoso notes the Azure Migrate limits summarized in the following table.

ACTION	LIMIT
Create Azure Migrate project	10,000 VMs
Discovery	10,000 VMs
Assessment	10,000 VMs

Contoso will use Azure Migrate as follows:

- In vCenter, organize VMs into folders. This will make it easy for the admins to focus as they run an assessment against VMs in a specific folder.
- Assess dependencies between machines. This requires agents to be installed on VMs to be assessed.

Contoso will use automated scripts to install the required Windows or Linux agents. By scripting, Contoso can push the installation to VMs within a vCenter folder.

Database tools

In addition to Azure Migrate, Contoso will focus on using tools specifically for database assessment. Tools such as Data Migration Assistant will help assess SQL Server databases for migration.

Data Migration Assistant can help Contoso to figure out whether on-premises databases are compatible with a range of Azure database solutions. These solutions include Azure SQL Database, SQL Server running on an Azure IaaS VM, and Azure SQL Managed Instance.

In addition to Database Migration Service, Contoso has some scripts that it uses to discover and document the SQL Server databases. These scripts are located in the GitHub repo.

Partner assessment tools

There are several other partner tools that can help Contoso assess the on-premises environment for migration to Azure. Learn more about Azure migration partners.

Step 2: Migrate

With the assessment complete, Contoso needs to identify tools to move its applications, data, and infrastructure to Azure.

Migration strategies

Contoso can consider four broad migration strategies.

STRATEGY	DETAILS	USAGE
Rehost	• Often called a *lift-and-shift* migration, this is a no-code option for migrating existing applications to Azure quickly. • An application is migrated as is with the benefits of the cloud and without the risks or costs associated with code changes.	• Contoso can rehost less-strategic applications that require no code changes.
Refactor	• Also called *repackaging*, this strategy requires minimal application code or configuration changes to connect the application to Azure PaaS, and take better advantage of cloud capabilities.	• Contoso can refactor strategic applications to retain the same basic functionality, but move them to run on an Azure platform such as Azure App Service. • This requires minimum code changes. • On the other hand, Contoso will have to maintain a VM platform because Microsoft won't manage this.
Rearchitect	• This strategy modifies or extends an application code base to optimize the application architecture for cloud capabilities and scale. • It modernizes an application into a resilient, highly scalable, independently deployable architecture. • Azure services can accelerate the process, scale applications with confidence, and manage applications with ease.	
Rebuild	• This approach rebuilds an application from scratch by using cloud-native technologies. • Azure PaaS provides a complete development and deployment environment in the cloud. It eliminates some expense and complexity of software licenses. It also removes the need for an underlying application infrastructure, middleware, and other resources.	• Contoso can rewrite critical applications to take advantage of cloud technologies such as serverless compute or microservices. • Contoso will manage the application and services that it develops, and Azure will manage everything else.

Data must also be considered, especially with the volume of databases that Contoso has. Contoso's default approach is to use PaaS services such as Azure SQL Database to take full advantage of cloud features. By

moving to a PaaS service for databases, Contoso will only have to maintain data. It will leave the underlying platform to Microsoft.

Evaluate migration tools

Contoso is primarily using these Azure services and tools for the migration:

- Azure Migrate: Service for migrating on-premises virtual machines and other resources to Azure.
- Azure Database Migration Service: Migrates on-premises databases such as SQL Server, MySQL, and Oracle to Azure.

Azure Migrate

Azure Migrate is the primary Azure service for orchestrating migration from within Azure and from on-premises sites to Azure.

Azure Migrate orchestrates replication from on-premises locations to Azure. When replication is set up and running, on-premises machines can be failed over to Azure, completing the migration.

Contoso already completed a POC to see how Azure Migrate can help it to migrate to the cloud.

Use Azure Migrate at scale

Contoso plans to perform multiple lift-and-shift migrations. To ensure that this works, Azure Migrate will replicate batches of around 100 VMs at a time. To determine how this will work, Contoso must perform capacity planning for the proposed migration.

Contoso needs to gather information about their traffic volumes. In particular:

- It needs to determine the rate of change for VMs that it wants to replicate.
- It needs to take network connectivity from the on-premises site to Azure into account.

In response to capacity and volume requirements, Contoso will need to allocate sufficient bandwidth based on the daily data change rate for the required VMs, to meet its recovery point objective (RPO). Last, it must determine how many servers are needed to run the Azure Migrate components for the deployment.

Gather on-premises information

Contoso can use Azure Migrate:

- To remotely profile VMs without an impact on the production environment. This helps pinpoint bandwidth and storage requirements for replication and failover.
- Without installing any Site Recovery components on-premises.

The tool gathers information about compatible and incompatible VMs, disks per VM, and data churn per disk. It also identifies network bandwidth requirements and the Azure infrastructure needed for successful replication and failover.

Contoso needs to ensure that it runs the planner tool on a Windows Server machine that matches the minimum requirements for the Site Recovery configuration server. The configuration server is a Site Recovery machine that's needed to replicate on-premises VMware VMs.

Identify Site Recovery requirements

In addition to the VMs being replicated, Site Recovery requires several components for VMware migration.

COMPONENT	DETAILS
Configuration server	• Usually a VMware VM configured through an OVF template. • The configuration server component coordinates communications between on-premises and Azure, and it manages data replication.

COMPONENT	DETAILS
Process server	• Installed by default on the configuration server. • The process server component receives replication data; optimizes it with caching, compression, and encryption; and sends it to Azure Storage. • The process server also installs the Azure Site Recovery Mobility service on VMs that you want to replicate, and performs automatic discovery of on-premises machines. • Scaled deployments need additional, standalone process servers to handle large volumes of replication traffic.
Mobility service	• The Mobility service agent is installed on each VMware VM that will be migrated with Azure Site Recovery.

Contoso needs to figure out how to deploy these components, based on capacity considerations.

COMPONENT	CAPACITY REQUIREMENTS
Maximum daily change rate	• A single process server can handle a daily change rate up to 2 terabytes (TB). Because a VM can only use one process server, the maximum daily data change rate that's supported for a replicated VM is 2 TB.
Maximum throughput	• A standard Azure Storage account can handle a maximum of 20,000 requests per second. I/O operations per second (IOPS) across a replicating VM should be within this limit. For example, if a VM has 5 disks, and each disk generates 120 IOPS (8K size) on the VM, then it will be within the Azure per-disk IOPS limit of 500. • The number of storage accounts needed equals the total source machine IOPS divided by 20,000. A replicated machine can belong to only a single storage account in Azure.
Configuration server	Based on Contoso's estimate of replicating 100 to 200 VMs together and the configuration server sizing requirements, Contoso estimates that it needs the following type of server machine configuration: • CPU: 16 vCPUs (2 sockets × 8 cores @ 2.5 GHz) • Memory: 32 GB • Cache disk: 1 TB • Data change rate: 1 to 2 TB In addition to sizing requirements, Contoso must ensure that the configuration server is optimally located on the same network and LAN segment as the VMs to be migrated.
Process server	Contoso will deploy a standalone dedicated process server with the ability to replicate 100 to 200 VMs: • CPU: 16 vCPUs (2 sockets × 8 cores @ 2.5 GHz) • Memory: 32 GB • Cache disk: 1 TB • Data change rate: 1 to 2 TB The process server will be working hard, so it's located on an ESXi host that can handle the disk I/O, network traffic, and CPU required for the replication. Contoso will consider a dedicated host for this purpose.

COMPONENT	CAPACITY REQUIREMENTS
Networking	• Contoso has reviewed the current Site-to-Site VPN infrastructure and decided to implement Azure ExpressRoute. The implementation is critical because it will lower latency and improve bandwidth to Contoso's primary Azure region (`East US 2`). • Contoso will need to carefully monitor data flowing from the process server. If the data overloads the network bandwidth, Contoso will consider throttling the process server bandwidth.
Azure Storage	• For migration, Contoso must identify the right type and number of target Azure Storage accounts. Site Recovery replicates VM data to Azure Storage. • Site Recovery can replicate to standard or premium SSD storage accounts. • To decide about storage, Contoso must review storage limits and consider expected growth and future increased usage. Given the speed and priority of migrations, Contoso has decided to use premium SSDs. • Contoso has decided to use managed disks for all VMs deployed to Azure. The IOPS required will help determine whether the disks will be standard HDD, standard SSD, or premium SSD.

Azure Database Migration Service

Azure Database Migration Service is a fully managed service that enables seamless migrations from multiple database sources to Azure data platforms with minimal downtime. Here are some details about the service:

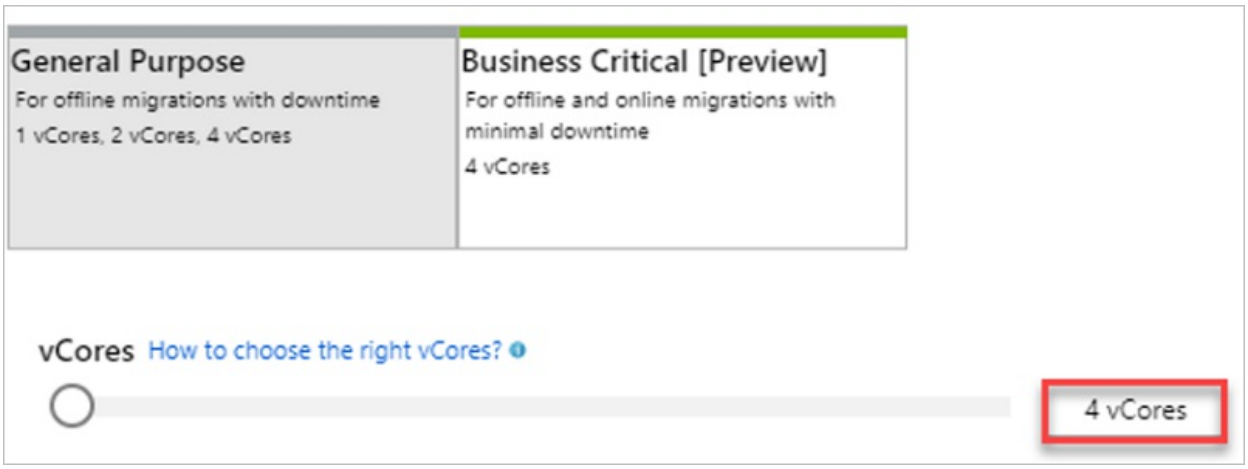
- It integrates functionality of existing tools and services. It uses Data Migration Assistant to generate assessment reports that pinpoint recommendations about database compatibility and any required modifications.
- It uses a simple, self-guided migration process with intelligent assessment that helps address potential issues before the migration.
- It can migrate at scale from multiple sources to the target Azure database.
- It provides support from SQL Server 2005 to SQL Server 2017.

Database Migration Service isn't the only Microsoft database migration tool. Get a comparison of tools and services.

Use Database Migration Service at scale

Contoso will use Database Migration Service when migrating from SQL Server.

When provisioning Database Migration Service, Contoso needs to size it correctly and set it to optimize performance for data migrations. Contoso will select the Business Critical service tier with four vCores. This option allows the service to take advantage of multiple vCPUs for parallelization and faster data transfer.



Another scaling tactic that Contoso can use is to temporarily scale up the Azure SQL Database or Azure Database for MySQL target instance to the Premium pricing tier during data migration. This minimizes database throttling that might affect data transfer activities when an organization is using lower tiers.

Use other tools

In addition to Database Migration Service, Contoso can use other tools and services to identify VM information:

- Scripts to help with manual migrations. These are available in the GitHub repo.
- Various partner tools for migration.

<-- docutune:ignore "cost management tools" -->

Ready for production

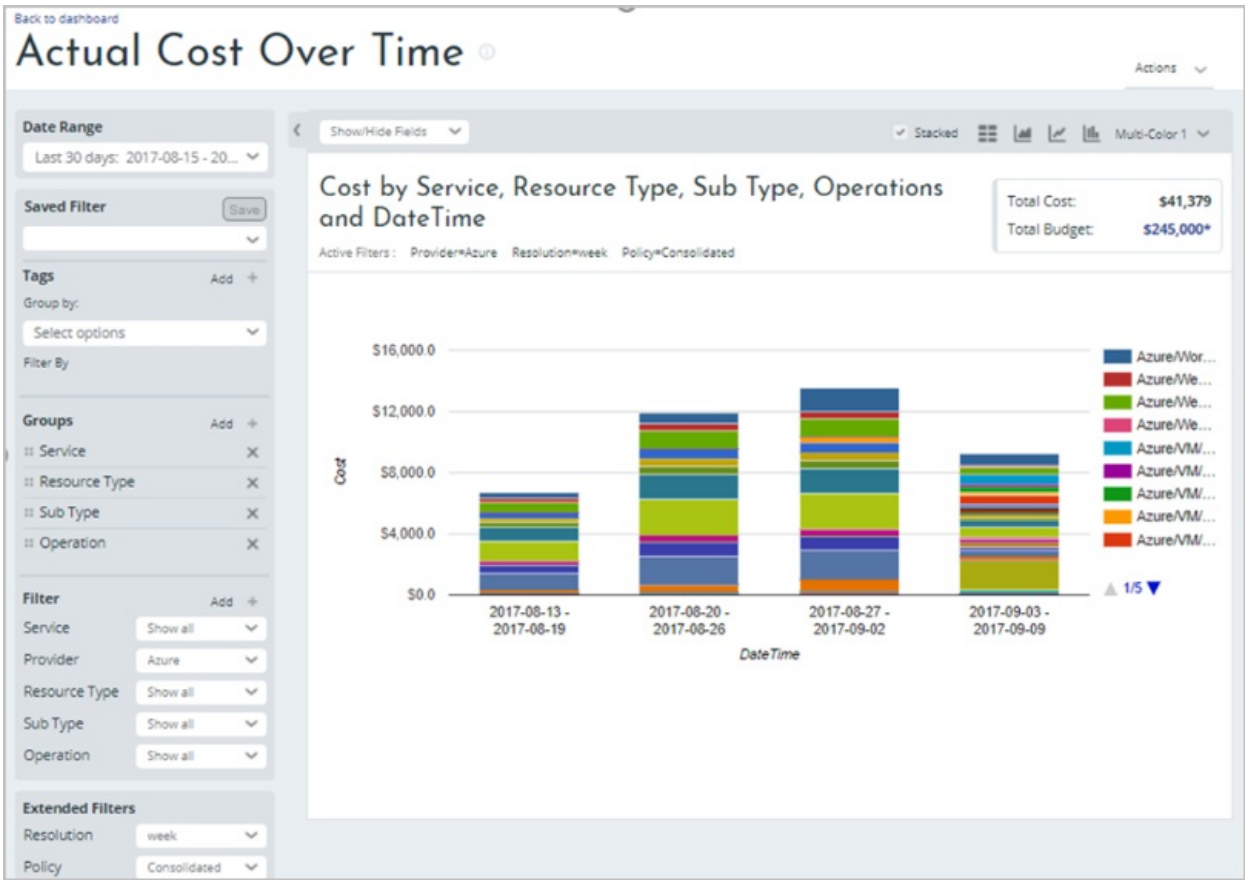
After Contoso moves resources to Azure, it needs to streamline them to improve performance, and maximize ROI with cost management tools. Because Azure is a pay-for-use service, it's critical for Contoso to understand how systems are performing and to ensure they're sized properly.

Azure Cost Management + Billing

To make the most of their cloud investment, Contoso will take advantage of the free Azure Cost Management + Billing tool. This solution allows Contoso to manage cloud spending with transparency and accuracy. It provides tools to monitor, allocate, and trim cloud costs.

Azure Cost Management + Billing provides simple dashboard reports to help with cost allocation, showbacks, and chargebacks. The tool can help optimize cloud spending by identifying underutilized resources that Contoso can then manage and adjust.

You can learn more in an overview of Azure Cost Management + Billing.



Native tools

Contoso will also use scripts to locate unused resources.

During large migrations, there are often leftover pieces of data such as virtual hard drives, which incur a charge but provide no value to the company. Scripts are available in the GitHub repo.

Contoso will take advantage of work done by Microsoft's IT department and consider implementing the Azure Resource Optimization (ARO) toolkit. The toolkit is also in the GitHub repo.

Contoso can deploy an Azure Automation account with preconfigured runbooks and schedules to its subscription, and start saving money. Resource optimization happens automatically on a subscription after a schedule is enabled or created, including optimization on new resources. This provides decentralized automation capabilities to reduce costs. Features include:

- Autosnooze Azure VMs based on low CPU utilization.
- Schedule Azure VMs to snooze and unsnooze.
- Schedule Azure VMs to snooze or unsnooze in ascending and descending order by using Azure tags.
- Bulk-delete resource groups on demand.

Partner optimization tools

Contoso can use partner tools such as Hanu and Scalr.

Step 4: Secure and manage

Contoso uses Azure security and management resources during this step to govern, secure, and monitor cloud applications in Azure. These resources help an organization run a secure and well-managed environment while using products available in the Azure portal.

Contoso begins using these services during migration. With Azure hybrid support, Contoso continues using many of them for a consistent experience across the hybrid cloud.

Security

Contoso will rely on Azure Security Center for unified security management and Microsoft Defender for Identity across hybrid cloud workloads.

Security Center provides full visibility into and control over the security of cloud applications in Azure. Contoso can quickly detect and take action in response to threats, and reduce security exposure by enabling adaptive threat protection.

Learn more about Security Center.

Monitoring

Contoso needs visibility into the health and performance of the newly migrated applications, infrastructure, and data now running in Azure. Contoso will use built-in Azure cloud monitoring tools such as Azure Monitor, a Log Analytics workspace, and Application Insights.

By using these tools, Contoso can easily collect data from sources and gain insights. For example, Contoso can gauge CPU disk and memory utilization for VMs, view applications and network dependencies across multiple VMs, and track application performance. Contoso will use these cloud monitoring tools to take action and integrate with service solutions.

Learn more about Azure monitoring.

Business continuity and disaster recovery

Contoso will need a business continuity and disaster recovery (BCDR) strategy for its Azure resources.

Azure provides built-in BCDR features to help protect data and keep applications and services running.

In addition to built-in features, Contoso wants to ensure that it can recover from failures, avoid costly business disruptions, meet compliance goals, and protect data against ransomware and human errors. To do this:

- Contoso will deploy Azure Backup as a cost-efficient solution for backup of Azure resources. Because it's built in, Contoso can set up cloud backups in a few simple steps.
- Contoso will set up disaster recovery for Azure VMs by using Azure Site Recovery for replication, failover, and failback between Azure regions that it specifies. This ensures that applications running on Azure VMs remain available in a secondary region of Contoso's choosing if an outage occurs in the primary region. Learn more.

Conclusion

In this article, Contoso planned for an Azure migration at scale. It divided the migration process into four stages. The stages ran from assessment and migration, through to optimization, security, and management after migration was complete.

It's important for an organization to plan a migration project as a whole process and to migrate its systems by breaking down sets into classifications and numbers that make sense for the business. By assessing data and applying classifications, projects can be broken down into a series of smaller migrations, which can run safely and rapidly. The sum of these smaller migrations quickly turns into a large successful migration to Azure.

Schema migration and DDL commands

7/15/2021 • 23 minutes to read • Edit Online

This article describes design considerations and performance options for data definition language (DDL) commands when you're migrating schemas to Azure Synapse Analytics.

Design considerations

Review these design considerations as you plan your schema migration.

Preparation for migration

When you're preparing to migrate existing data to Azure Synapse Analytics, it's important to clearly define the scope of the exercise (especially for an initial migration project). The time spent up front to understand how database objects and related processes will migrate can reduce both effort and risk later in the project.

Create an inventory of database objects to be migrated. Depending on the source platform, this inventory will include some or all of the following objects:

- Tables
- Views
- Indexes
- Functions
- Stored procedures
- Data distribution and partitioning

The basic information for these objects should include metrics such as row counts, physical size, data compression ratios, and object dependencies. This information should be available via queries against system catalog tables in the source system. The system metadata is the best source for this information. External documentation might be stale and not in sync with changes that have been applied to the data structure since the initial implementation.

You might also be able to analyze actual object usage from query logs or use tooling from Microsoft partners like Attunity Visibility to help. It's possible that some tables don't need to be migrated because they're no longer used in production queries.

Data size and workload information is important for Azure Synapse Analytics because it helps to define appropriate configurations. One example is the required levels of concurrency. Understanding the expected growth of data and workloads might affect a recommended target configuration, and it's a good practice to also harness this information.

When you're using data volumes to estimate the storage required for the new target platform, it's important to understand the data compression ratio, if any, on the source database. Simply taking the amount of storage used on the source system is likely to be a false basis for sizing. Monitoring and metadata information can help you determine uncompressed raw data size and overheads for indexing, data replication, logging, or other processes in the current system.

The uncompressed raw data size of the tables to be migrated is a good starting point for estimating the storage required in the new target Azure Synapse Analytics environment.

The new target platform will also include a compression factor and indexing overhead, but these will probably be different from the source system. Azure Synapse Analytics storage pricing also includes seven days of snapshot backups. When compared to the existing environment, this can have an impact on the overall cost of

storage required.

You can delay performance tuning for the data model until late in the migration process and time this with when real data volumes are in the data warehouse. However, we recommend that you implement some performance tuning options earlier on.

For example, in Azure Synapse Analytics, it makes sense to define small dimension tables as replicated tables and to define large fact tables as clustered columnstore indexes. Similarly, indexes defined in the source environment provide a good indication of which columns might benefit from indexing in the new environment. Using this information when you're initially defining the tables before loading will save time later in the process.

It's good practice to measure the compression ratio and index overhead for your own data in Azure Synapse Analytics as the migration project progresses. This measure enables future capacity planning.

It might be possible to simplify your existing data warehouse before migration by reducing complexity to ease migration. This effort might include:

- Removing or archiving unused tables before migrating to avoid migrating data that's not used. Archiving to Azure Blob Storage and defining the data as an external table might keep the data available for a lower cost.
- Converting physical data marts to virtual data marts by using data virtualization software to reduce what you have to migrate. This conversion also improves agility and reduces total cost of ownership. You might consider it as modernization during migration.

One objective of the migration exercise might also be to modernize the warehouse by changing the underlying data model. One example is moving from an Inmon-style data model to a data vault approach. You should decide this as part of the preparation phase and incorporate a strategy for the transition into the migration plan.

The recommended approach in this scenario is to first migrate the data model as is to the new platform and then transition to the new model in Azure Synapse Analytics. Use the platform's scalability and performance characteristics to execute the transformation without affecting the source system.

Data model migration

Depending on the platform and the origins of the source system, the data model of some or all parts may already be in a star or snowflake schema form. If so, you can directly migrate it to Azure Synapse Analytics as is. This scenario is the easiest and lowest-risk migration to achieve. An as-is migration can also be the first stage of a more complex migration that includes a transition to a new underlying data model such as a data vault, as described earlier.

Any set of relational tables and views can be migrated to Azure Synapse Analytics. For analytical query workloads against a large data set, a star or snowflake data model generally gives the best overall performance. If the source data model is not already in this form, it might be worth using the migration process to reengineer the model.

If the migration project includes any changes to the data model, the best practice is to perform these changes in the new target environment. That is, migrate the existing model first, and then use the power and flexibility of Azure Synapse Analytics to transform the data to the new model. This approach minimizes the impact on the existing system and uses the performance and scalability of Azure Synapse Analytics to make any changes quickly and cost-effectively.

You can migrate the existing system as several layers; for example, the data ingest/staging layer, data warehouse layer, and reporting or data mart layer. Each layer consists of relational tables and views. Although you can migrate these to Azure Synapse Analytics as they are, it might be more cost-effective and reliable to use some of the features and capabilities of the Azure ecosystem. For example:

- **Data ingest and staging:** Instead of using relational tables for fast parallel data loading, you can use Azure Blob storage with PolyBase during part of the ETL (extract, transform, load) or ELT (extract, load, transform) process.

- **Data ingest and staging:** You can use Azure Blob Storage in conjunction with PolyBase for fast parallel data loading for part of the ETL (extract, transform, load) or ELT (extract, load, transform) process, rather than relational tables.
- **Reporting layer and data marts:** The performance characteristics of Azure Synapse Analytics might eliminate the need to physically instantiate aggregated tables for reporting purposes or data marts. It might be possible to implement these as views onto the core data warehouse or via a third-party data virtualization layer. At the basic level, you can achieve the process for data migration of historical data and possibly also incremental updates as shown in this diagram:

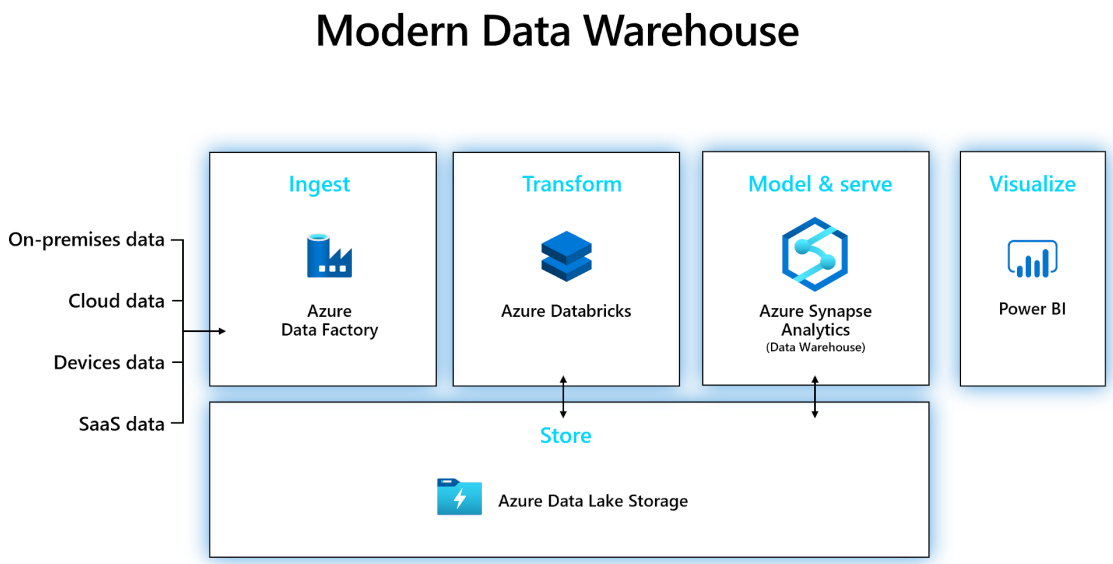


Figure 1: A modern data warehouse.

If you can use these or similar approaches, the number of tables to be migrated is reduced. Some processes might be simplified or eliminated, again reducing the migration workload. The applicability of these approaches depends on the individual use case. But, the general principle is to consider using the features and facilities of the Azure ecosystem, where possible, to reduce the migration workload and build a cost-effective target environment. This also holds true for other functions, such as backup/restore and workflow management and monitoring.

Products and services available from Microsoft partners can help with data warehouse migrations and automate parts of the process for some cases. If the existing system incorporates a third-party ETL product, it might already support Azure Synapse Analytics as a target environment. The existing ETL workflows can be redirected to the new target data warehouse.

Data marts: Physical or virtual

It's a common practice for organizations with older data warehouse environments to create data marts that provide their departments or business functions with good ad hoc self-service query and report performance. A data mart typically consists of a subset of the data warehouse that contains aggregated versions of the original data. Its form, typically a dimensional data model, supports users to easily query the data and receive fast response times from user-friendly tools like Tableau, MicroStrategy, or Microsoft Power BI.

One use of data marts is to expose the data in a usable form, even if the underlying warehouse data model is something different (such as a data vault). This approach is also known as a three-tier model.

You can use separate data marts for individual business units within an organization to implement robust data security regimes. For example, you can allow user access to specific data marts relevant to them and eliminate, obfuscate, or anonymize sensitive data.

If these data marts are implemented as physical tables, they require additional storage resources to house them

and additional processing to build and refresh them regularly. Physical tables show that the data in the mart is only as current as the last refresh operation, so they may not be suitable for highly volatile data dashboards.

With the advent of low-cost, scalable massively parallel processing (MPP) architectures, you might be able to provide data mart functionality without needing to instantiate the mart as a set of physical tables. You can achieve this with Azure Synapse Analytics by using one of these methods to virtualize the data marts:

- SQL views on the main data warehouse.
- A virtualization layer that uses features such as views in Azure Synapse Analytics or third-party virtualization products such as Denodo.

This approach simplifies or eliminates the need for additional storage and aggregation processing. It reduces the overall number of database objects to be migrated.

Another benefit of the data warehouse approach is the capacity to run operations such as joins and aggregations on large data volumes. For example, implementing the aggregation and join logic within a virtualization layer and displaying external reporting in a virtualized view push the robust processing required to create these views into the data warehouse.

The primary drivers for choosing to implement physical or virtual data mart implementation are:

- More agility. A virtual data mart is easier to change than physical tables and the associated ETL processes.
- Lower total cost of ownership because of fewer data stores and copies of data in a virtualized implementation.
- Elimination of ETL jobs to migrate and simplified data warehouse architecture in a virtualized environment.
- Performance. Historically, physical data marts have been more reliable. Virtualization products are now implementing intelligent caching techniques to mitigate this.

You can also use data virtualization to display data to users consistently during a migration project.

Data mapping

Key and integrity constraints in Azure Synapse Analytics

Primary key and foreign key constraints aren't currently enforced within Azure Synapse Analytics. However, you can include the definition for `PRIMARY KEY` in the `CREATE TABLE` statement with the `NOT ENFORCED` clause. This means that third-party reporting products can use the metadata for the table to understand the keys within the data model and therefore generate the most efficient queries.

Data type support in Azure Synapse Analytics

Some older database systems include support for data types that aren't directly supported within Azure Synapse Analytics. You can handle these data types by using a supported data type to store the data as is or by transforming the data to a supported data type.

Here's an alphabetical list of supported data types:

- `bigint`
- `binary [ (n) ]`
- `bit`
- `char [ (n) ]`
- `date`
- `datetime`
- `datetime2 [ (n) ]`
- `datetimeoffset [ (n) ]`
- `decimal [ (precision [, scale ]) ]`
- `float [ (n) ]`

- `int`
- `money`
- `nchar [ (n) ]`
- `numeric [ (precision [ , scale ]) ]`
- `nvarchar [ (n | MAX) ]`
- `real [ (n) ]`
- `smalldatetime`
- `smallint`
- `smallmoney`
- `time [ (n) ]`
- `tinyint`
- `uniqueidentifier`
- `varbinary [ (n | MAX) ]`
- `varchar [ (n | MAX) ]`

The following table lists common data types that aren't currently supported, together with the recommended approach for storing them in Azure Synapse Analytics.

UNSUPPORTED DATA TYPE	WORKAROUND
`geometry`	`varbinary`
`geography`	`varbinary`
`hierarchyid`	`nvarchar(4000)`
`image`	`varbinary`
`text`	`varchar`
`ntext`	`nvarchar`
`sql_variant`	Split column into several strongly typed columns
`table`	Convert to temporary tables
`timestamp`	Rework code to use `datetime2` and the `CURRENT_TIMESTAMP` function
`xml`	`varchar`
User-defined type	Convert back to the native data type when possible

Potential data issues

Depending on the source environment, some issues can cause problems when you're migrating data:

- There can be subtle differences in the way that `NULL` data is handled in different database products. Examples include collation sequence and handling of empty character strings.
- `DATE`, `TIME`, `INTERVAL`, and `TIME ZONE` data and associated functions can vary widely from product to product.

Test these thoroughly to determine whether the desired results are achieved in the target environment. The migration exercise can uncover bugs or incorrect results that are currently part of the existing source system, and the migration process is a good opportunity to correct anomalies.

Best practices for defining columns in Azure Synapse Analytics

It's common for older systems to contain columns with inefficient data types. For example, you might find a field defined as `VARCHAR(20)` when the actual data values would fit into a `CHAR(5)` field. Or, you might find the use of `INTEGER` fields when all values would fit within a `SMALLINT` field. Insufficient data types can lead to inefficiencies in both storage and query performance, especially in large fact tables.

It's a good time to check and rationalize current data definitions during a migration exercise. You can automate these tasks by using SQL queries to find the maximum numeric value or character length within a data field and comparing the result to the data type.

In general, it's a good practice to minimize the total defined row length for a table. For the best query performance, you can use the smallest data type for each column, as described earlier. The recommended approach to load data from external tables in Azure Synapse Analytics is to use the PolyBase utility, which supports a maximum defined row length of 1 megabyte (MB). PolyBase won't load tables with rows longer than 1 MB, and you must use the `bcp` utility instead.

For the most efficient join execution, define the columns on both sides of the join as the same data type. If the key of a dimension table is defined as `SMALLINT`, then the corresponding reference columns in fact tables using that dimension should also be defined as `SMALLINT`.

Avoid defining character fields with a large default size. If the maximum size of data within a field is 50 characters, use `VARCHAR(50)`. Similarly, don't use `NVARCHAR` if `VARCHAR` will suffice. `NVARCHAR` stores Unicode data to allow for different language character sets. `VARCHAR` stores ASCII data and takes less space.

Summary of design recommendations

Don't migrate unnecessary objects or processes. Use built-in features and functions in the target Azure environment where appropriate to reduce the actual number of objects and processes to migrate. Consider using a virtualization layer to reduce or eliminate the number of physical data marts that you'll migrate and to push down processing into the data warehouse.

Automate wherever possible, and use metadata from system catalogs in the source system to generate DDL commands for the target environment. If possible, also automate generating documents. Microsoft partners such as WhereScape can provide specialized tools and services to assist with automation.

Perform any required data model changes or data-mapping optimizations on the target platform. You can make these changes more efficiently in Azure Synapse Analytics. This approach reduces the impact on source systems that might already be running close to full capacity.

Performance options

This section describes the features available within Azure Synapse Analytics that you can use to improve performance for a data model.

General approach

The platform's features run performance tuning on the database that will be migrated. Indexes, data partitioning, and data distribution are examples of such performance tuning. When you're preparing for migration, documenting the tuning can capture and reveal optimizations that you can apply in the Azure Synapse Analytics target environment.

For example, the presence of a non-unique index on a table can indicate that fields used in the index are used frequently for filtering, grouping, or joining. This will still be the case in the new environment, so keep it in mind

when you're choosing which fields to index there. For more detailed information about Teradata or Netezza environments, see the Azure Synapse Analytics articles about solutions and migration for Teradata and solutions and migration for Netezza.

Use the performance and scalability of the target Azure Synapse Analytics environment to experiment with different performance options like data distribution. Determine the best choice of alternative approaches (for example, replicated versus hash-distributed for a large dimension table). This doesn't mean that data must be reloaded from external sources. It's relatively quick and easy to test alternative approaches in Azure Synapse Analytics by creating copies of any table with different partitioning or distribution options via a `CREATE TABLE AS SELECT` statement.

Use the monitoring tools provided by the Azure environment to understand how queries are executed and where bottlenecks might be occurring. Tools are also available from third-party Microsoft partners to provide monitoring dashboards and automated resource management and alerting.

Each SQL operation in Azure Synapse Analytics and resource, such as memory or the CPU used by that query, is logged into system tables. A series of dynamic management views simplifies access to this information.

The following sections explain the key options within Azure SQL Data Warehouse for tuning query performance. Existing environments will contain information about potential optimization in the target environment.

Temporary tables

Azure Synapse Analytics supports temporary tables that are visible only to the session in which they were created. They exist for the duration of a user session and are automatically dropped at the end of the session.

To create a temporary table, prefix the table name with the hash character (`#`). You can use all the usual indexing and distribution options with temporary tables, as described in the next section.

Temporary tables have some restrictions:

- Renaming them isn't allowed.
- Viewing or partitioning them isn't allowed.
- Changing permissions isn't allowed.

Temporary tables are commonly used within ETL/ELT processing, where transient intermediate results are used as part of a transformation process.

Table distribution options

Azure Synapse Analytics is an MPP database system that achieves performance and scalability by running in parallel across multiple processing nodes.

The ideal processing scenario for running an SQL query in a multi-node environment is to balance the workload and give all nodes an equal amount of data to process. This approach also allows you to minimize or eliminate the amount of data that has to be moved between nodes to satisfy the query.

It can be challenging to achieve the ideal scenario because there are often aggregations in typical analytics queries and multiple joins between several tables, as between fact and dimension tables.

One way to influence how queries are processed is to use the distribution options within Azure Synapse Analytics to specify where each table's individual data rows are stored. For example, assume that two large tables are joined on the data column, `CUSTOMER_ID`. By distributing the two tables through the `CUSTOMER_ID` columns whenever that join is performed, you can ensure that the data from each side of the join will already be co-located on the same processing node. This method eliminates the need to move data between nodes. The distribution specification for a table is defined in the `CREATE TABLE` statement.

The following sections describe the available distribution options and recommendations for when to use them. It's possible to change the distribution of a table after the initial load, if necessary: re-create the table with the

new distribution by using the `CREATE TABLE AS SELECT` statement.

Round robin

Round-robin table distribution is the default option and spreads the data evenly across the nodes in the system. This method is good for fast data loading and for data that's relatively low in volume and doesn't have an obvious candidate for hashing. It's frequently used for staging tables as part of an ETL or ELT process.

Hashed

The system assigns the row to a hash bucket, a task based on a hashing algorithm applied to a user-defined key like `CUSTOMER_ID` in the preceding example. The bucket is then assigned to a specific node, and all data rows hash-distributed on the same value end up on the same processing node.

This method is useful for large tables that are frequently joined or aggregated on a key. Other large tables to be joined should be hashed on the same key if possible. If there are multiple candidates for the hash key, choose the most frequently joined one.

The hash column shouldn't contain nulls and isn't typically a date because many queries filter on date. Hashing is typically more efficient if the key to hash is an integer value instead `CHAR` or `VARCHAR`. Avoid choosing keys with a highly skewed range of values, like when a small number of key values represent a large percentage of the data rows.

Replicated

Choosing replicated as the distribution option for a table will cause a complete copy of that table to be replicated on each compute node for query processing purposes.

This approach is useful for relatively small tables (typically less than 2 GB compressed) that are relatively static and frequently joined to larger tables via an equi-join. These tables are often dimensional tables in a star schema.

Indexing

Azure Synapse Analytics includes options for indexing data in large tables to reduce the resources and time required to retrieve records:

- Clustered columnstore index
- Clustered index
- Non-clustered index

A non-indexed option, `HEAP`, exists for tables that don't benefit from any of the index options. Using indexes is a trade-off between improved query times versus longer load times and usage of more storage space. Indexes often speed up `SELECT`, `UPDATE`, `DELETE`, and `MERGE` operations on large tables that affect a small percentage of the data rows, and they can minimize full table scans.

Indexes are automatically created when `UNIQUE` or `PRIMARY KEY` constraints are defined on columns.

Clustered columnstore index

Clustered columnstore index is the default indexing option within Azure Synapse Analytics. It provides the best compression and query performance for large tables. For smaller tables of fewer than 60 million rows, these indexes aren't efficient, so you should use the `HEAP` option. Similarly, a heap or a temporary table might be more efficient if the data in a table is transient and part of an ETL/ELT process.

Clustered index

If there's a requirement to regularly retrieve a single row or small number of rows from a large table based on a strong filter condition, a clustered index might be more efficient than a clustered columnstore index. Only one clustered index is allowed per table.

Non-clustered index

Non-clustered indexes are similar to clustered indexes in that they can speed up retrieval of single rows or a small number of rows based on a filter condition. Internally, non-clustered indexes are stored separately from

the data, and multiple non-clustered indexes can be defined on a table. However, each additional index will require more storage and will reduce the throughput of data insert or loading.

Heap

Heap tables incur none of the overhead associated with the creation and maintenance of indexes at data load time. They can help to quickly load transient data during processes, including ELT processes. Caching can also assist when the data is read immediately afterward. Because clustered columnstore indexes are inefficient below 60 million rows, heap tables can also help to store tables with rows less than this amount.

Data partitioning

In an enterprise data warehouse, fact tables can contain many billions of rows. Partitioning is a way to optimize the maintenance and querying of these tables by splitting them into separate parts to reduce the amount of data processed when running queries. The partitioning specification for a table is defined in the `CREATE TABLE` statement.

You can use only one field per table for partitioning. It's frequently a date field because many queries are filtered by a date or date range. You can change the partitioning of a table after initial load, if necessary, by re-creating the table with the new distribution through the `CREATE TABLE AS SELECT` statement.

Partitioning for query optimization: If queries against a large fact table are frequently filtered by a certain data column, then partitioning on that column can significantly reduce the amount of data that needs to be processed to perform the queries. A common example is to use a date field to split the table into smaller groups. Each group contains data for a single day. When a query contains a `WHERE` clause that filters on the date, only partitions that match the date filter need to be accessed.

Partitioning for optimization of table maintenance: It's common for data warehouse environments to maintain a rolling window of detailed fact data. An example is sales transactions that go back five years. By partitioning on the sales date, the removal of old data beyond the rolling window becomes much more efficient. Dropping the oldest partition is quicker and uses fewer resources than deleting each individual row.

Statistics

When a query is submitted to Azure Synapse Analytics, it's first processed by the query optimizer. The optimizer determines the best internal methods to execute the query efficiently.

The optimizer compares the various query-execution plans that are available based on a cost-based algorithm. The accuracy of the cost estimates is dependent on the statistics available. It's a good practice to ensure that statistics are up to date.

In Azure Synapse Analytics, if the `AUTO_CREATE_STATISTICS` option is turned on, it will trigger an automatic update of statistics. You can also create or update statistics manually via the `CREATE STATISTICS` command.

Refresh statistics when the contents have changed substantially, such as in a daily update. This refresh can be incorporated into an ETL process.

All tables in the database should have statistics collected on at least one column. It ensures that basic information such as row count and table size is available to the optimizer. Other columns that should have statistics collected are columns specified in `JOIN`, `DISTINCT`, `ORDER BY`, and `GROUP BY` processing.

Workload management

Azure Synapse Analytics incorporates comprehensive features for managing resource utilization across mixed workloads. Creating resource classes for different workload types, such as queries versus data load, helps you manage your workload. It sets limits on the number of queries that run concurrently and on the compute resources assigned to each query. There's a trade-off between memory and concurrency:

- Smaller resource classes reduce the maximum memory per query but increase concurrency.
- Larger resource classes increase the maximum memory per query but reduce concurrency.

Performance recommendations

Use performance improvement methods like indexes or data distribution to gauge candidates for similar methods in the new target environment, but benchmark to confirm that they're necessary in Azure Synapse Analytics. Build `COLLECT STATISTICS` steps into ETL/ELT processes to ensure that statistics are up to date, or select to automatically create statistics.

Understand the tuning options available in Azure Synapse Analytics and the performance characteristics of associated utilities, such as PolyBase for fast parallel data loading. Use these options to build an efficient end-to-end implementation.

Use the flexibility, scalability, and performance of the Azure environment to implement any data model changes or performance tuning options in place. This effort will reduce the impact on existing source systems.

Understand the dynamic management views available in Azure Synapse Analytics. These views provide both system-wide resource utilization information and detailed execution information for individual queries.

Understand Azure resource classes and allocate them appropriately to ensure efficient management of mixed workloads and concurrency.

Consider using a virtualization layer as part of the Azure Synapse Analytics environment. It can shield changes in the warehouse implementation from business users and reporting tools.

Research partner-provided migration tools and services such as Qlik Replicate for Microsoft migrations, WhereScape, and Datometry. These services can automate parts of the migration process and reduce the elapsed time and risk involved in a migration project.

Database restore points for Azure Synapse Analytics

7/1/2021 • 2 minutes to read • Edit Online

One of the key benefits of a modern cloud-based infrastructure such as Microsoft Azure is that features for database restore points and disaster recovery (DR) are built in and simple to implement and customize. These facilities are often lower in cost than the equivalent functionality within an on-premises environment. Using these built-in functions also means that the backup and recovery mechanisms in the existing data warehouse don't need to be migrated.

The following sections describe the standard Azure Synapse Analytics features that address requirements for database restore points and disaster recovery.

Database restore points

Azure Synapse Analytics uses database snapshots to provide database restore points for the warehouse. A data warehouse snapshot creates a restore point that can be used to recover or copy a data warehouse to a previous state. Because Azure Synapse Analytics is a distributed system, a data warehouse snapshot consists of many files that are located in Azure Storage. Snapshots capture incremental changes from the data stored in your data warehouse.

Azure Synapse Analytics automatically takes snapshots throughout the day to create restore points that are available for seven days. This retention period can't be changed. Azure Synapse Analytics supports an eight-hour recovery point objective (RPO). You can restore a data warehouse in the primary region from any one of the snapshots taken in the past seven days.

The service also supports user-defined restore points. Manually triggering snapshots can create restore points of a data warehouse before and after large modifications. This capability ensures that restore points are logically consistent. Logical consistency provides additional data protection against workload interruptions or user errors for quick recovery time.

Disaster recovery

In addition to the snapshots described earlier, Azure Synapse Analytics performs a standard geo-backup once per day to a paired datacenter. The RPO for a geo-restore is 24 hours. The dedicated SQL pool is backed up. You can restore the geo-backup to a server in any other region where Azure Synapse Analytics is supported. A geo-backup ensures that a data warehouse can be restored in case the restore points in the primary region are not available.

For more details about database backup and recovery, see Backup and restore in Azure Synapse Dedicated SQL pool.

Governance or compliance strategy

5/25/2021 • 4 minutes to read • Edit Online

When governance or compliance is required throughout a migration effort, you need to broaden your scope to account for these requirements. The following guidance expands the scope of the Azure migration guide to address different approaches to addressing governance or compliance requirements.

General scope expansion

Prerequisite activities are affected the most when governance or compliance are required. Additional adjustments can also be required during assessment, migration, and optimization.

Suggested prerequisites

Configuration of the base Azure environment can change significantly when you're integrating governance or compliance requirements. To understand how prerequisites change, it's important to understand the nature of the requirements. Prior to beginning any migration that requires governance or compliance, you should choose and implement an approach in the cloud environment. The following are a few high-level approaches commonly seen during migrations:

Common governance approach: For most organizations, the Cloud Adoption Framework governance model is a sufficient approach. It consists of a minimum viable product (MVP) implementation, followed by targeted iterations of governance maturity to address tangible risks identified in the adoption plan. This approach provides the minimum tooling needed to establish consistent governance, so the team can understand the tools. It then expands on those tools to address common governance concerns.

International Organization for Standardization (ISO) 27001 compliance blueprints: If your organization is required to adhere to ISO compliance standards, the ISO 27001 Shared Services blueprint samples can serve as a more effective MVP. The blueprint can produce richer governance constraints, earlier in the iterative process. The ISO 27001 App Service Environment/SQL Database workload blueprint sample expands on the ISO 27001 Shared Services blueprint, to map controls and deploy a common architecture for an application environment.

Cloud Adoption Framework enterprise-scale landing zone: You might require a more robust governance starting point. If so, consider the Cloud Adoption Framework enterprise-scale landing zone. The Cloud Adoption Framework enterprise-scale landing zone approach focuses on adoption teams who have a mid-term objective (within 24 months) to host more than 1,000 assets (applications, infrastructure, or data assets) in the cloud. The Cloud Adoption Framework enterprise-scale landing zone is the de facto choice for complex governance scenarios for these larger cloud adoption efforts.

Partnership option to complete prerequisites

Microsoft Services: Microsoft Services provides solution offerings that can align to the Cloud Adoption Framework governance model, compliance blueprints, or Cloud Adoption Framework enterprise-scale landing zone options. This option helps you to ensure that you're using the most appropriate governance or compliance model. Use the Secure Cloud Insights solution to establish a data-driven picture of a customer deployment in Azure. This solution also validates the customer´s Azure implementation maturity while identifying optimization of existing deployment architectures. Secure Cloud Insights also helps you reduce risk pertaining to governance, security, and availability. Based on customer insights, you should lead with the following approaches:

- **Cloud foundation:** Establish the customer's core Azure designs, patterns, and governance architecture with the hybrid cloud foundation solution. Map the customer's requirements to the most appropriate reference

architecture. Implement a minimum viable product consisting of shared services and IaaS workloads.

- **Cloud modernization:** Use the cloud modernization solution as a comprehensive approach to move applications, data, and infrastructure to an enterprise-ready cloud. You can also optimize and modernize after cloud deployment.
- **Innovate with cloud:** Engage the customer through the cloud center of excellence (CCoE) solution. It implements an agile approach to capture business requirements, and to reuse deployment packages aligned with security, compliance, and service management policies. It also maintains the alignment of the Azure platform with operational procedures.

Assess process changes

During assessment, you must make additional decisions to align to the required governance approach. The cloud governance team provides all members of the cloud adoption team with any policy statements, architectural guidance, or governance or compliance requirements prior to the assessment of a workload.

Suggested action during the assessment process

Governance and compliance assessment requirements are too customer-specific to provide general guidance on the actual steps taken during assessment. The process should include tasks and time for aligning to compliance and governance requirements.

For a deeper understanding of governance, read the overview of the Disciplines of Cloud Governance. This section of the Cloud Adoption Framework includes templates to document the policies, guidance, and requirements for each of the following sections:

- Cost Management discipline
- Security Baseline discipline
- Resource Consistency discipline
- Identity Baseline discipline
- Deployment Acceleration discipline

For information about developing governance guidance based on the Cloud Adoption Framework governance model, see Implement a cloud governance strategy.

Optimize and promote process changes

During the optimization and promotion processes, the cloud governance team should invest time to test and validate adherence to governance and compliance standards. Additionally, this step is a good time for the cloud governance team to curate templates that provide additional guidance for future projects, particularly in the Deployment Acceleration discipline.

Suggested action during the optimize and promote process

During this process, the project plan should include time allocations for the cloud governance team to execute a compliance review, for each workload planned for production promotion.

Next steps

Return to the checklist to reevaluate any additional scope requirements for the migration effort.

Migration best practices checklist

Cloud migration antipatterns

3/29/2021 • 5 minutes to read • Edit Online

Customers often experience antipatterns during the migration phase of cloud adoption. The following measures help to avoid migration antipatterns:

- Making sure security and compliance guardrails are in place.
- Understanding potential application and server dependencies.
- Choosing an architecture based on a thorough assessment.

Antipattern: Migrate, modernize, or innovate without guardrails

When customers deploy their first workloads to the cloud, they consider it as a platform to test innovative solutions. They enjoy the flexibility that's available within the cloud. But, whenever these workloads become productive, need to hold company data, or need access to company systems, progress slows down as they need to adhere to the compliance, regulatory, and security standards.

Example: Omit security guardrails

A company wants to modernize its online shop to improve its user experience. The modernization should be done by moving the online shop website and the underlying inventory database to Azure. Because dependencies exist between the inventory database and the company's SAP system, those systems need to communicate. So, the company needs to build up a hybrid cloud.

The online shop team is innovative, so it begins modernizing the application, but it doesn't consider security requirements because of the hybrid connection. When it tests the application, it finds that the IT security team doesn't allow communication within Azure and on-premises systems, because security and compliance requirements aren't met.

Preferred outcome: Establish security and compliance guardrails

Before you shift workloads to the cloud, put security and compliance guardrails in place. These guardrails make sure that workloads follow security and compliance requirements. Have the cloud governance and cloud security teams deliver the guardrails within an Azure landing zone. Check the guardrails with IT, especially for hybrid workloads. See Cloud Adoption Framework enterprise-scale landing zone architecture for help with defining guardrails that support workload teams, so that they work in a fast, consistent, compliant, and secure manner.

Antipattern: Migrate, modernize, or innovate without an assessment

When a company considers a migration or modernization project, it needs to understand potential application and server dependencies so that it can plan more accurately. In application innovation scenarios, a company experiences more success by using architectural design sessions and reference architectures instead of aimless engineering efforts.

Example: Cause downtime by migrating without planning thoroughly

A team member plans to migrate applications to the cloud to reduce the company's carbon footprint. The migration plan, which identifies the first asset to migrate, is based on configuration management database (CMDB) entries and a single application owner interview. After the team member migrates one of the application's database servers, several other application owners call IT to complain that their applications aren't working properly. The dependencies depicted in the CMDB aren't accurate anymore, causing unexpected downtime in other applications.

Preferred outcome: Assess infrastructure before migrating or modernizing

For a large-scale migration or modernization project, perform an infrastructure assessment before migration starts. This assessment helps you identify dependencies and compatibility issues. See Azure migration guide for detailed information that the Microsoft Cloud Adoption Framework for Azure provides on migration best practices.

In modernization projects, use additional application assessments to identify coding antipatterns, compatibility issues, and technical debt. For more information about modernization aspects, see Overview of application migration examples for Azure.

For innovation projects, see Azure innovative solutions guide overview for help with identifying the correct way to plan and develop an innovative cloud solution.

For mission-critical workloads or workloads which require architectural change, use the Azure Well-Architected Framework or an architectural design session (ADS) to help design, build, and deploy a high-quality, robust architecture that scales within an enterprise. Use ADS whiteboards to discover, envision, and plan the solution.

Antipattern: Dictate an architecture

A company might pursue a microservice-first strategy when developing in the cloud, assuming that a microservice architecture always outperforms a traditional, monolithic architecture. If the company doesn't perform a proper application assessment and due diligence for its application, this strategy can fail. Other architectural approaches might be more suitable for the application. Choosing or dictating a microservice architecture or an architecture for all situations often results in failed projects.

Example: Use a microservice architecture for all applications

A company's chief information officer (CIO) establishes a policy of using a microservice architecture when building up new applications in the cloud. The company developers have never worked with the microservice architecture. They need to develop a simple web app. After working on the application for a few months, the developers realize that they probably would have already finished development if they had started with a monolithic architecture. The company hasn't achieved a faster time to market, among other benefits.

Preferred outcome: Base architectural decisions on assessments

Instead of fixating on a specific architecture style, make an architectural decision based on an assessment and due diligence of the use case or an architecture. Don't limit what architectures can be used, because freedom of choice is one of the major benefits of the cloud. Picking an architecture just because it's in fashion is an antipattern to avoid. For more information, see the Azure application architecture guide and cloud design patterns.

Antipattern: Use a single subscription

Companies often decide to use only one subscription to host all their workloads. They usually make this choice when implementing rapid migrations that require speed over everything else. This decision leads to poorly governed and designed landscapes. These companies can quickly run into subscription limits, which means they need to redesign the architecture.

Example: Migrate under one subscription

A conglomerate decides to spin off its hotel division into a separate company. The hotel division needs to move or migrate its IT assets to a new place. The new hotel company chooses a cloud-first approach and migrates all IT assets to the cloud. Because of time constraints, the new company migrates everything into one subscription and uses a huge virtual network, where there are few possibilities for separating duties and the security model properly. Three months after the spin-off is complete, the hotel company determines that its assets are less secure and governed than before, and that it's running into subscription limits.

Preferred outcome: Use a segmentation strategy

Separate various duties and plan for a different environment before you migrate to Azure. You can reach subscription limits quickly when you combine different stages into one subscription. Establish a segmentation strategy to make it easier to implement governance and compliance.

Next steps

- Azure migration guide overview
- Azure cloud migration best practices checklist
- Management group and subscription organization

Cloud Adoption Framework migration model

3/6/2021 • 3 minutes to read • Edit Online

This section of the Cloud Adoption Framework explains the principles behind its migration model. Wherever possible, this content attempts to maintain a vendor-neutral position while guiding you through the processes and activities that can be applied to any cloud migration, regardless of your chosen cloud vendor.

Understand migration motivations

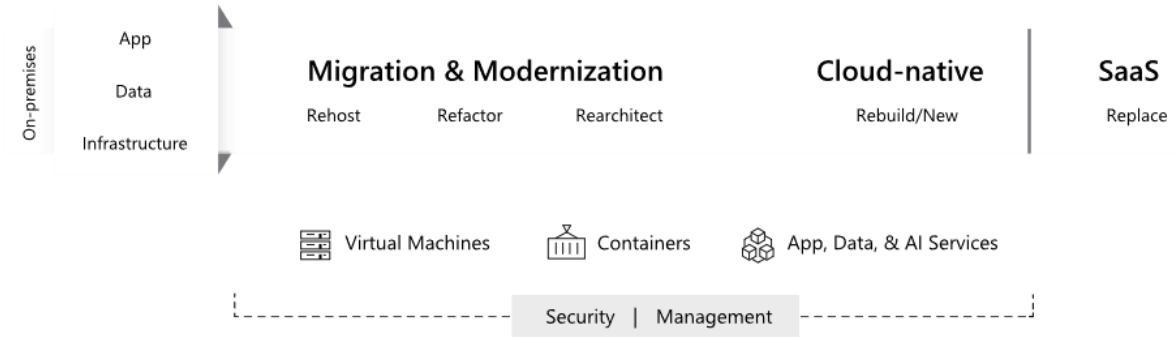
Cloud migration is a portfolio management effort, cleverly disguised as a technical implementation. During the migration process, you will decide to move some assets, invest in others, and retire obsolete or unused assets. Some assets will be optimized, refactored, or replaced entirely as part of this process. Each of these decisions should align with the motivations behind your cloud migration. The most successful migrations also go a step further and align these decisions with desired business outcomes.

The Cloud Adoption Framework migration model depends on your organization having completed a process of business readiness for cloud adoption. Make sure you have reviewed the Plan methodology and the Ready methodology guidance in the Cloud Adoption Framework to determine the business drivers or other justification for a cloud migration, as well as any required organizational planning or training required before executing a migration process at scale.

> **NOTE**
> While business planning is important, a growth mindset is equally important. In parallel with broader business planning efforts by the cloud strategy team, it's suggested that the cloud adoption team begin migrating a first workload as a precursor to wider scale migration efforts. This initial migration will allow the team to gain practical experience with the business and technical issues involved in a migration.

Envision an end state

It's important to establish a rough vision of your end state before starting your migration efforts. The following diagram shows an on-premises starting point of infrastructure, applications, and data, which defines your *digital estate*. During the migration process, those assets are transitioned using one of the five migration strategies described in the five Rs of rationalization.



Migration and modernization of workloads range from simple *rehost* migrations (also called *lift and shift* migrations) using infrastructure as a service (IaaS) capabilities that don't require code and application changes, through *refactoring* with minimal changes, to *rearchitecting* to modify and extend code and application functionality to take advantage of cloud technologies.

Cloud-native strategies and platform as a service (PaaS) strategies *rebuild* on-premises workloads using Azure

platform offerings and managed services. Workloads that have equivalent fully managed software as a service (SaaS) cloud-based offerings can often be fully *replaced* by these services as part of the migration process.

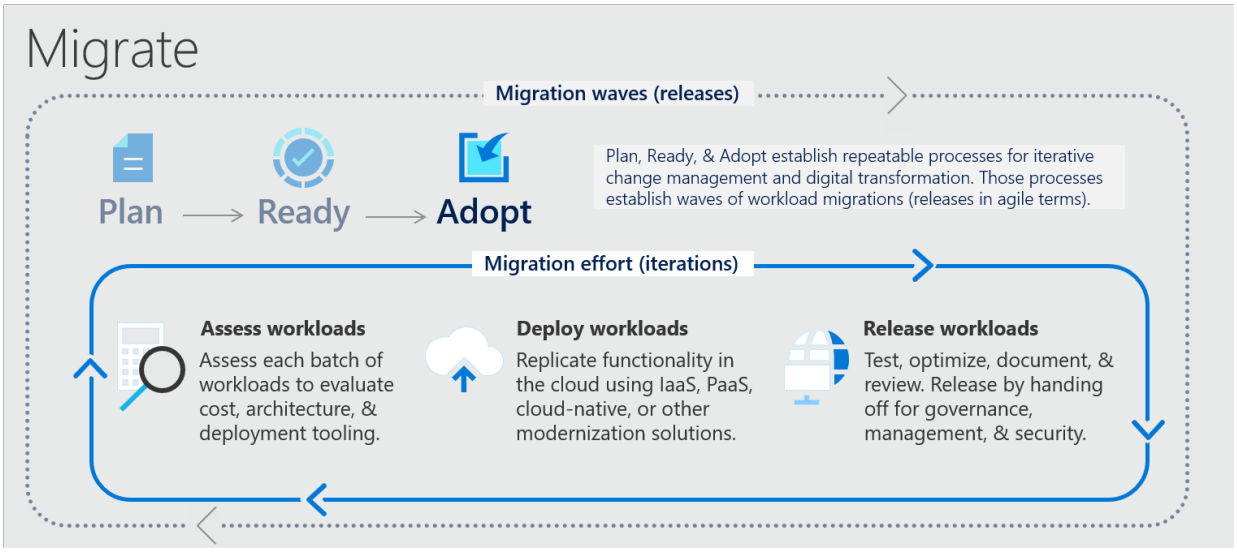
> **NOTE**
>
> During the public preview of the Cloud Adoption Framework, this section of the framework emphasizes a rehost migration strategy. Although PaaS and SaaS solutions are discussed as alternatives when appropriate, the migration of virtual machine-based workloads using IaaS capabilities is the primary focus.
>
> Other sections and future iterations of this content will expand on other approaches. For a high-level discussion on expanding the scope of your migration to include more complicated migration strategies, see the article balancing the portfolio.

Incremental migration

The Cloud Adoption Framework migration model is based on an incremental cloud transformation process. It assumes that your organization will start with an initial, limited-scope, cloud migration effort, which we refer to commonly as the first workload. This effort will expand iteratively to include more workloads as your operations teams refine and improve your migration processes.

Cloud migrations tools like Azure Site Recovery can migrate entire datacenters consisting of tens of thousands of VMs. However, the business and existing IT operations can seldom handle such a high pace of change. As such many organizations break up a migration effort into multiple iterations, moving one workload (or a collection of workloads) per iteration.

The principles behind this incremental model are based on the execution of processes and prerequisites referenced in the following infographic.



The consistent application of these principles represents an end goal for your cloud migration processes and should not be viewed as a required starting point. As your migration efforts mature, refer to the guidance in this section to help define the best process to support your organizational needs.

Next steps

Begin learning about this model by investigating the prerequisites to migration.

Prerequisites to migration

Prerequisites for migration

3/6/2021 • 3 minutes to read • Edit Online

Prior to beginning any migrations, your migration target environment must be prepared for the coming changes. In this case, *environment* refers to the technical foundation in the cloud. Environment also means the business environment and mindset driving the migration. Likewise, the environment includes the culture of the teams executing the changes and those receiving the output. Lack of preparation for these changes is the most common reason for failure of migrations. This series of articles walks you through suggested prerequisites to prepare the environment.

Objective

Ensure business, culture, and technical readiness prior to beginning an iterative migration plan.

Review business drivers

Before beginning any cloud migration, review the Plan methodology and the Ready methodology in the Cloud Adoption Framework to ensure your organization is prepared for cloud adoption and migration processes. In particular, review the business requirements and expected outcomes driving the migration:

- Get started: Accelerate migration
- Why are we moving to the cloud?

Definition of done

Prerequisites are completed when the following are true:

- **Business readiness.** The cloud strategy team has defined and prioritized a high-level migration backlog representing the portion of the digital estate to be migrated in the next two or three releases. The cloud strategy team and the cloud adoption team have agreed to an initial strategy for managing change.
- **Culture readiness.** The roles, responsibilities, and expectations of the cloud adoption team, cloud strategy team, and affected users have been agreed on regarding the workloads to be migrated in the next two or three releases.
- **Technical readiness.** The landing zone (or allocated hosting space in the cloud) that will receive the migrated assets meets minimum requirements to host the first migrated workload.

Caution

Preparation is key to the success of a migration. However, too much preparation can lead to *analysis paralysis*, where too much time spent on planning can seriously delay a migration effort. The processes and prerequisites defined in this section are meant to help you make decisions, but don't let them block you from making meaningful progress.

Choose a relatively simple workload for your initial migration. Use the processes discussed in this section as you plan and implement this first migration. This first migration effort will quickly demonstrate cloud principles to your team and force them to learn about how the cloud works. As your team gains experience, integrate these learnings as you take on larger and more complex migrations.

Accountability during prerequisites

Two teams are accountable for readiness when completing prerequisites:

- **Cloud strategy team:** This team is responsible for identifying and prioritizing the first two or three workloads to serve as migration candidates.
- **Cloud adoption team:** This team is responsible for validating readiness of the technical environment and the feasibility of migrating the proposed workloads.

A single member of each team should be identified as accountable for each of the three definitions of done statements in the prior section.

Responsibilities during prerequisites

In addition to the high-level accountability, there are actions that an individual or group needs to be directly responsible for. The following are a few such responsibilities that affect these activities:

- **Business prioritization.** Make business decisions regarding the workloads to be migrated and general timing constraints. For more information, see Cloud migration business motivations.
- **Change management readiness.** Establish and communicate the plan for tracking technical change during migration.
- **Business user alignment.** Establish a plan for readying the business user community for migration execution.
- **Digital estate inventory and analysis.** Execution of the tools required to inventory and analyze the digital estate. See the Cloud Adoption Framework discussion of the digital estate for more information.
- **Cloud readiness.** Evaluate the target deployment environment to ensure that it complies with requirements of the first few workload candidates. See the Azure setup guide for more information.

The remaining articles in this series help with the execution of each.

Next steps

With a general understanding of the prerequisites, you're ready to address the first prerequisite early migration decisions.

Early migration decisions

Decisions that affect migration

3/6/2021 • 6 minutes to read • Edit Online

During migration, several factors affect decisions and execution activities. This article explains the central theme of those decisions and explores a few questions that carry through the discussions of migration principles in this section of the Cloud Adoption Framework guidance.

Business outcomes

The objective or goal of any adoption effort can have a significant impact on the suggested approach to execution.

- **Migration.** Urgent business drivers, speed of adoption, or cost savings are examples of operational outcomes. These outcomes are central to efforts that drive business value from transitive change in IT or operations models. The Migrate methodology of the Cloud Adoption Framework focuses heavily on migration focused business outcomes.
- **Application innovation.** Improving customer experience and growing market share are examples of incremental outcomes. The outcomes result from a collection of incremental changes focused on the needs and desires of current customers.
- **Data-driven innovation.** New products or services, especially ones that come from the power of data, are examples of disruptive outcomes. These outcomes are the result of experimentation and predictions that use data to disrupt status quo in the market.

No business would pursue just one of these outcomes. Without operations, there are no customers, and vice versa. Cloud adoption is no different. Companies commonly work to achieve each of these outcomes, but trying to focus on all of them simultaneously can spread your efforts too thin and slow progress on work that could most benefit your business needs.

This prerequisite isn't a demand for you to pick one of these three goals, but instead to help your cloud strategy team and your cloud adoption team establish a set of operational priorities that will guide execution for the next three to six months. These priorities are set by ranking each of the three itemized options from most significant to least significant, as they relate to the efforts this team can contribute to in the next one or two quarters.

Act on migration outcomes

If operational outcomes rank highest in the list, this section of the Cloud Adoption Framework will suit your team well. In this section, it is assumed that you need to prioritize speed and cost savings as primary key performance indicators (KPIs), in which case a migration model to adoption would be well aligned with the outcomes. A migration-focused model is heavily predicated on lift and shift migration of infrastructure as a service (IaaS) assets to deplete a datacenter and to produce cost savings. In such a model, modernization may occur but is a secondary focus until the primary migration mission is realized.

Act on application innovations

If market share and customer experience are your primary drivers, this section may not be the best section of the Cloud Adoption Framework to guide your teams' efforts. Application innovation requires a plan that focuses on the modernization and transition of workloads, regardless of the underlying infrastructure. In such a case, the guidance in this section can be informative but may not be the best approach to guide core decisions.

Act on data innovations

If data, experimentation, research and development (R&D), or new products are your priority for the next six months or so, this section may not be the best section of the Cloud Adoption Framework to guide your teams'

efforts. Any data innovation effort could benefit from guidance regarding the migration of existing source data. However, the broader focus of that effort would be on the ingress and integration of additional data sources. Extending that guidance with predictions and new experiences is much more important than the migration of IaaS assets.

Effort

Migration effort can vary widely depending on the size and complexities of the workloads involved. A smaller workload migration involving a few hundred virtual machines (VMs) is a tactical process, potentially being implemented using automated tools such as Azure Migrate. Conversely, a large enterprise migration of tens of thousands of workloads requires a highly strategic process and can involve extensive refactoring, rebuilding, and replacing of existing applications integrating platform as a service (PaaS) and software as a service (SaaS) capabilities. Identifying and balancing the scope of your planned migrations is critical.

Before making any decisions that could have a long-term impact on the current migration program, it is vital that you create consensus on the following decisions.

Effort type

In any migration of significant scale (more than 250 VMs), assets are migrated using a variety of transition options, discussed in the five Rs of rationalization: *rehost*, *refactor*, *rearchitect*, *rebuild*, and *replace*.

Some workloads are modernized through a *rebuild* or *rearchitect* process, creating more modern applications with new features and technical capabilities. Other assets go through a *refactor* process, for instance a move to containers or other more modern hosting and operational approaches that don't necessarily affect the solutions code base. Commonly, virtual machines and other assets that are more well established go through a *rehost* process, transitioning those assets from the datacenter to the cloud. Some workloads could potentially be migrated to the cloud, but instead should be *replaced* using service-based (SaaS-based) cloud services that meet the same business need—for example, by using Microsoft 365 as an alternative to migrating Exchange Server instances.

In the majority of scenarios, some business event creates a forcing function that causes a high percentage of assets to temporarily migrate using the *rehost* process, followed by a more significant secondary transition using one of the other migration strategies after they're in the cloud. This process is commonly known as a *cloud transition*.

During the process of rationalizing the digital estate, these types of decisions are applied to each asset to migrate. However, the prerequisite needed at this time is to make a baseline assumption. Of the five migration strategies, which best aligns with the business objectives or business outcomes driving this migration effort? This decision serves as a guiding assumption throughout the migration effort.

Effort scale

Scale of the migration is the next important prerequisite decision. The processes needed to migrate 1,000 assets are different from the processes required to move 10,000 assets. Before beginning any migration effort, it is important to answer the following questions:

- **How many assets support the migrating workloads today?** Assets include data structures, applications, VMs, and necessary IT appliances. Choose a relatively small workload for your first migration candidate.
- **Of those assets, how many are planned for migration?** It's common for some assets to be terminated during a migration process, due to lack of sustained end-user dependency.
- **What are the top-down estimates of the scale of migrateable assets?** For the workloads included for migration, estimate the number of supporting assets such as applications, virtual machines, data sources, and IT appliances. See the Digital estate section of the Cloud Adoption Framework for guidance on identifying relevant assets.

Effort timing

Often, migrations are driven by a compelling business event that is time sensitive. For instance, one common driver is the termination or renewal of a third-party hosting contract. Although there are many potential business events necessitating a migration, they share a common factor: an end date. It is important to understand the timing of any approaching business events, so activities and velocity can be planned and validated properly.

Recap

Before proceeding, document the following assumptions and share them with the cloud strategy team and the cloud adoption teams:

- Business outcomes.
- Roles, documented and refined for the *Assess*, *Migrate*, *Optimize*, and *Secure and manage* migration processes.
- Definition of done, documented and refined separately for these migration processes.
- Effort type.
- Effort scale.
- Effort timing.

Next steps

After the process is understood among the team, it's time to review technical prerequisites. The migration environment planning checklist helps to ensure that the technical foundation is ready for migration.

Once the process is understood among the team, its time to review technical prerequisites the migration planning checklist will help ensure the technical foundation is ready for migration.

Review the migration planning checklist

Migration environment planning checklist: Validate environmental readiness prior to migration

3/6/2021 • 3 minutes to read • Edit Online

As an initial step in the migration process, you need to create the right environment in the cloud to receive, host, and support migrating assets. This article provides a list of things to validate in the current environment prior to migration.

The following checklist aligns with the guidance in the Ready methodology of the Cloud Adoption Framework. Review that section for guidance regarding execution of any of the following.

Effort type assumption

This article and checklist assume a *rehost* or *cloud transition* approach to cloud migration.

Governance alignment

The first and most important decision regarding any migration-ready environment is the choice of governance alignment. Has a consensus been achieved regarding alignment of governance with the migration foundation? At a minimum, the cloud adoption team should understand whether this migration is landing in a single environment with limited governance, a fully governed environment factory, or some variant in between. For additional guidance on governance alignment, see the Govern methodology.

Operations management alignment

Before migrating assets into the cloud, it is important to understand any requirements or constraints regarding operations management. At a minimum, the migration environment should include any implementations required to meet the operations baseline. For additional guidance on operations alignment, see the Manage methodology.

Cloud readiness implementation

Whether you choose to align with a broader cloud governance strategy or not for your initial migration, you will need to ensure your cloud deployment environment is configured to support your workloads.

If you're planning to align your migration with a cloud governance strategy from the start, you'll need to apply the Five Disciplines of Cloud Governance to help inform decisions on policies, toolchains, and enforcement mechanisms that will align your cloud environment with overall corporate requirements. Consult the Cloud Adoption Framework actionable governance design guides for examples of how to implement this model using Azure services.

If your initial migrations are not closely aligned with a broader cloud governance strategy, the general issues of organization, access, and infrastructure planning still need to be managed. Consult the Azure setup guide for help with these cloud readiness decisions.

Caution

We highly recommend that you develop a governance strategy for anything beyond your initial workload migration.

Regardless of your level of governance alignment, you will need to make decisions related to the following topics.

Resource organization

Based on the governance alignment decision, an approach to the organization and deployment of resources should be established prior to migration.

Nomenclature

A consistent approach for naming resources, along with consistent naming schemas, should be established prior to migration.

Resource governance

A decision regarding the tools to govern resources should be made prior to migration. The tools do not need to be fully implemented, but a direction should be selected and tested. The cloud governance team should define and require the implementation of a minimum viable product (MVP) for governance tooling prior to migration.

Network

Your cloud-based workloads will require the provisioning of virtual networks to support end-user and administrative access. Based on resource organization and resource governance decisions, you should select a network approach align it to IT security requirements. Further, your networking decisions should be aligned with any hybrid network constraints required to operate the workloads in the migration backlog and support any access to resources hosted on-premises.

Identity

Cloud-based identity services are a prerequisite for offering identity and access management (IAM) for your cloud resources. Align your identity management strategy with your cloud adoption plans before proceeding. For example, when migrating existing on-premises assets, consider supporting a hybrid identity approach using directory synchronization to allow a consistent set of user credentials across you on-premises and cloud environments during and after the migration.

Next steps

If the environment meets the minimum requirements, it may be deemed approved for migration readiness. Cultural complexity and change management helps to align roles and responsibilities to ensure proper expectations during execution of the plan.

Cultural complexity and change management

Prepare for cultural complexity: Aligning roles and responsibilities

3/6/2021 • 3 minutes to read • Edit Online

An understanding of the culture required to operate the existing datacenters is important to the success of any migration. In some organizations, datacenter management is contained within centralized IT operations teams. In these centralized teams, roles and responsibilities tend to be well defined and well understood throughout the team. For larger enterprises, especially those bound by third-party compliance requirements, the culture tends to be more nuanced and complex. Cultural complexity can lead to roadblocks that are difficult to understand and time consuming to overcome.

In either scenario, it's wise to invest in the documentation of roles and responsibilities required to complete a migration. This article outlines some of the roles and responsibilities seen in a datacenter migration, to serve as a template for documentation that can drive clarity throughout execution.

Business functions

In any migration, there are a few key functions that are best executed by the business, whenever possible. Often, IT is capable of completing the following tasks. However, engaging members of the business could aid in reducing barriers later in the adoption process. It also ensures mutual investment from key stakeholders throughout the migration process.

PROCESS	ACTIVITY	DESCRIPTION
Assess	Business goals	Define the desired business outcomes of the migration effort.
Assess	Priorities	Ensure alignment with changing business priorities and market conditions.
Assess	Justification	Validate assumptions that drive changing business justifications.
Assess	Risk	Help the cloud adoption team understand the impact of tangible business risks.
Assess	Approve	Review and approve the business impact of proposed architecture changes.
Optimize	Change plan	Define a plan for consumption of change within the business, including periods of low activities and change freezes.
Optimize	Testing	Align power users capable of validating performance and functionality.

PROCESS	ACTIVITY	DESCRIPTION
Secure and manage	Interruption impact	Aid the cloud adoption team in quantifying the impact of a business process interruption.
Secure and manage	Service-level agreement (SLA) validation	Aid the cloud adoption team in defining service-level agreements and acceptable tolerances for business outages.

Ultimately, the cloud adoption team is accountable for each of these activities. However, establishing responsibilities and a regular cadence with the business for the completion of these activities on an established rhythm can improve stakeholder alignment and cohesiveness with the business.

Common roles and responsibilities

Each process within the discussion of the Cloud Adoption Framework migration principles includes a process article outlining specific activities to align roles and responsibilities. For clarity during execution, a single accountable party should be assigned for each activity, along with any responsible parties required to support those activities. However, the following list contains a series of common roles and responsibilities that have a higher degree of impact on migration execution. These roles should be identified early in the migration effort.

> **NOTE**
> In the following table, an accountable party should start the alignment of roles. That column should be customized to fit existing processes for efficient execution. Ideally, a single person should be assigned as the accountable party.

PROCESS	ACTIVITY	DESCRIPTION	ACCOUNTABLE PARTY
Prerequisite	Digital estate	Align the existing inventory to basic assumptions, based on business outcomes.	Cloud strategy team
Prerequisite	Migration backlog	Prioritize the sequence of workloads to be migrated.	Cloud strategy team
Assess	Architecture	Challenge initial assumptions to define the target architecture based on usage metrics.	Cloud adoption team
Assess	Approval	Approve the proposed architecture.	Cloud strategy team
Migrate	Replication access	Access to existing on-premises hosts and assets to establish replication processes.	Cloud adoption team
Optimize	Ready	Validate that the system meets performance and cost requirements prior to promotion.	Cloud adoption team

PROCESS	ACTIVITY	DESCRIPTION	ACCOUNTABLE PARTY
Optimize	Promote	Permissions to promote a workload to production and redirect production traffic.	Cloud adoption team
Secure and manage	Ops transition	Document production systems prior to production operations.	Cloud adoption team

Caution

For these activities, permissions and authorization heavily influence the accountable party, who must have direct access to production systems in the existing environment or must have means of securing access through other responsible actors. Determining this accountable party directly affects the promotion strategy during the migrate and optimize processes.

Next steps

When the team has a general understanding of roles and responsibilities, it's time to begin preparing the technical details of the migration. Understanding technical complexity and change management can help prepare the cloud adoption team for the technical complexity of migration by aligning to an incremental change management process.

Technical complexity and change management

Prepare for technical complexity: Agile change management

5/18/2021 • 12 minutes to read • Edit Online

When an entire datacenter can be deprovisioned and re-created with a single line of code, traditional processes struggle to keep up. The guidance throughout the Cloud Adoption Framework is built on practices like IT service management (ITSM), the open group architecture framework (TOGAF), and others. However, to ensure agility and responsiveness to business change, this framework molds those practices to fit agile methodologies and DevOps approaches.

When shifting to an agile model where flexibility and iteration are emphasized, technical complexity and change management are handled differently than they're in a traditional waterfall model focusing on a linear series of migration steps. This article outlines a high-level approach to change management in an agile-based migration effort. At the end of this article, you should have a general understanding of the levels of change management and documentation involved in an incremental migration approach. Additional training and decisions are required to select and implement agile practices based on that understanding. The intention of this article is to prepare cloud architects for a facilitated conversation with project management to explain the general concept of change management in this approach.

Address technical complexity

When changing any technical system, complexity and interdependency inject risk into project plans. Cloud migrations are no exception. When moving thousands (or tens of thousands) of assets to the cloud, these risks are amplified. Detecting and mapping all dependencies across a large digital estate could take years. Few businesses can tolerate such a long analysis cycle. To balance the need for architectural analysis and business acceleration, the Cloud Adoption Framework focuses on an INVEST model for product backlog management. The following sections summarize this type of model.

INVEST in workloads

The term *workload* appears throughout the Cloud Adoption Framework. A workload is a unit of application functionality that can be migrated to the cloud. It could be a single application, a layer of an application, or a collection of an application. The definition is flexible and may change at various phrases of migration. The Cloud Adoption Framework uses the term *INVEST* to define a workload.

INVEST is a common acronym in many agile methodologies for writing user stories or product backlog items, both of which are units of output in agile project management tools. The measurable unit of output in a migration is a migrated workload. The Cloud Adoption Framework modifies the INVEST acronym a bit to create a construct for defining workloads:

- **Independent:** A workload should not have any inaccessible dependencies. For a workload to be considered migrated, all dependencies should be accessible and included in the migration effort.
- **Negotiable:** As additional discovery is performed, the definition of a workload changes. The architects planning the migration could negotiate factors regarding dependencies. Examples of negotiation points could include prerelease of features, making features accessible over a hybrid network, or packaging all dependencies in a single release.
- **Valuable:** Value in a workload is measured by the ability to provide users with access to a production workload.
- **Estimable:** Dependencies, assets, migration time, performance, and cloud costs should all be estimable and

should be estimated prior to migration.

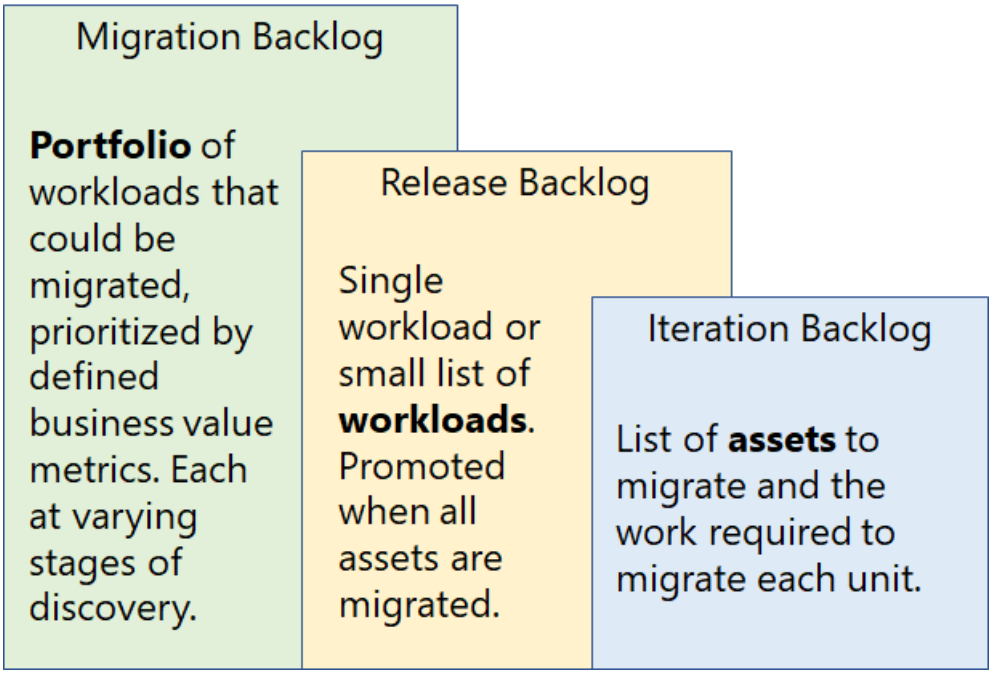
- **Small:** The goal is to package workloads in a single sprint. However, this may not always be feasible. Instead, teams are encouraged to plan sprints and releases to minimize the time required to move a workload to production.
- **Testable:** There should always be a defined means of testing or validating completion of the migration of a workload.

This acronym is not intended as a basis for rigid adherence but should help guide the definition of the term *workload*.

Migration backlog: Aligning business priorities and timing

The migration backlog allows you to track your top-level portfolio of workloads that can be migrated. Prior to migration, the cloud strategy team and the cloud adoption team are encouraged to perform a review of the current digital estate, and agree to a prioritized list of workloads to be migrated. This list forms the basis of the initial migration backlog.

Initially, workloads on the migration backlog are unlikely to meet the INVEST criteria outlined in the previous section. Instead, they serve as a logical grouping of assets from an initial inventory as a placeholder for future work. Those placeholders may not be technically accurate, but they serve as the basis for coordination with the business.



The migration, release, and iteration backlogs track different levels of activity during migration processes.

In any migration backlog, the change management team should strive to obtain the following information for any workload in the plan. At a minimum, this data should be available for any workloads prioritized for migration in the next two or three releases.

Migration backlog data points

- **Business impact.** Understanding of the impact to the business of missing the expected timeline or reducing functionality during freeze windows.
- **Relative business priority.** A ranked list of workloads based on business priorities.
- **Business owner.** Document the one individual responsible for making business decisions regarding this workload.
- **Technical owner.** Document the one individual responsible for technical decisions related to this workload.
- **Expected timelines.** When the migration is scheduled for completion.

- **Workload freezes.** Time frames in which the workload should be ineligible for change.
- **Workload name.**
- **Initial inventory.** Any assets required to provide the functionality of the workload, including VMs, IT appliances, data, applications, deployment pipelines, and others. This information is likely to be inaccurate.

Release backlog: Aligning business change and technical coordination

In the context of a migration, a *release* is an activity that deploys one or more workloads into production. A release generally covers several iterations or technical work. However, it represents a single iteration of business change. After one or more workloads have been prepared for production promotion, a release occurs. The decision to package a release is made when the workloads migrated represent enough business value to justify injecting change into a business environment. Releases are executed in conjunction with a business change plan, after business testing has been completed. The cloud strategy team is responsible for planning and overseeing the execution of a release to ensure that the desired business change is released.

A *release backlog* is the future state plan that defines a coming release. Release backlog is the pivot point between business change management (*migration backlog*) and technical change management (*sprint backlog*). A release backlog consists of a list of workloads from the migration backlog that align to a specific subset of business outcome realization. Definition and submission of a release backlog to the cloud adoption team serve as a trigger for deeper analysis and migration planning. After the cloud adoption team has verified the technical details associated with a release, it can choose to commit to the release, establishing a release timeline based on current knowledge.

Given the degree of analysis required to validate a release, the cloud strategy team should maintain a running list of the next two to four releases. The team should also attempt to validate as much of the following information as possible, before defining and submitting a release. A disciplined cloud strategy team capable of maintaining the next four releases can significantly increase the consistency and accuracy of release timeline estimates.

Release backlog data points

A partnership between the cloud strategy team and the cloud adoption team collaborates to add the following data points for any workloads in the release backlog:

- **Refined inventory.** Validation of required assets to be migrated. Often validated through log or monitoring data at the host, network, or OS level to ensure an accurate understanding of network and hardware dependencies of each asset under standard load.
- **Usage patterns.** An understanding of the patterns of usage from end users. These patterns often include an analysis of end-user geographical distribution, network routes, seasonal usage spikes, daily/hourly usage spikes, and end-user composition (interval versus external).
- **Performance expectations.** Analysis of available log data capturing throughput, page views, network routes, and other performance data required to replicate the end-user experience.
- **Dependencies.** Analysis of network traffic and application usage patterns to identify any additional workload dependencies, which should be factored into sequencing and environmental readiness. Don't include a workload in a release until one of the following criteria can be met:
  - All dependent workloads have been migrated.
  - Network and security configurations have been implemented to allow the workload to access all dependencies in alignment with existing performance expectations.
- **Desired migration approach.** At the migration backlog level, the assumed migration effort is the only consideration used in analysis. For instance, if the business outcome is an exit from an existing datacenter, all migrations are assumed to be a rehost scenario in the migration backlog. In the release backlog, the cloud strategy team and the cloud adoption team should evaluate the long-term value of additional features, modernization, and continued development investments to evaluate whether a more modern approach should be involved.

- **Business testing criteria.** After a workload is added to the migration backlog, testing criteria should be mutually agreed on. In some cases, testing criteria can be limited to a performance test with a defined power user group. However, for statistical validation, an automated performance test is desired and should be included. The existing instance of the application often has no automated testing capabilities. Should this prove accurate, it is not uncommon for the cloud architects to work with power users to create a baseline load test against the existing solution to establish a benchmark to be used during migration.

Release backlog cadence

In mature migrations, releases come in a regular cadence. The velocity of the cloud adoption team often normalizes, producing a release every two to four iterations (approximately every one or two months). However, this should be an organic outcome. Creating artificial release cadences can negatively affect the cloud adoption team's ability to achieve consistent throughput.

To stabilize business impact, the cloud strategy team should establish a monthly release process with the business to maintain regular dialogue but should also establish the expectation that it will be several months before a regular release cadence can be predicted.

Sprint or iteration backlog: Aligning technical change and effort

A *sprint*, or *iteration*, is a consistent, time-bound unit of work. In the migration process, this is often measured in two-week increments. However, it's not unheard of to have one-week or four-week iterations. Creating time-bound iterations forces consistent intervals of effort completion and allows for more frequent adjustment to plans, based on new learnings. During any given sprint, there are usually tasks for the assessment, migration, and optimization of workloads defined in the migration backlog. Those units of work should be tracked and managed in the same project-management tool as the migration and release backlog, to drive consistency across each level of change management.

A *sprint backlog*, or *iteration backlog*, consists of the technical work to be completed in a single sprint or iteration, dealing with migrating individual assets. That work should be derived from the list of workloads being migrated. When using tools like Azure DevOps (previously Visual Studio online) for project management, the work items in a sprint would be children of the product backlog items in a release backlog and the epics in a migration backlog. Such a parent-child relationship allows for clarity at all levels of change management.

Within a single sprint or iteration, the cloud adoption team would work to deliver the committed amount of technical work, driving toward the migration of a defined workload. This is the end result of the change management strategy. When complete, these efforts can be tested by validating production readiness of a workload staged in the cloud.

Large or complex sprint structures

For a small migration with a self-contained migration team, a single sprint could include all four phases of a migration for a single workload (*Assess*, *Migrate*, *Optimize*, and *Secure and manage*). More commonly, each of these processes shared by multiple teams in distinct work items across numerous sprints. Depending on the effort type, effort scale, and roles, these sprints can take a few different shapes.

- **Migration factory.** Large-scale migrations sometimes require an approach that resembles a factory in the execution model. In this model, various teams are allocated to the execution of a specific migration process (or subset of the process). After completion, the output of one team's sprint populates the backlog for the next team. This is an efficient approach for large-scale rehost migrations of many potential workloads involving thousands of virtual machines moving through phases of assessment, architecture, remediation, and migration. However, for this approach to work, a new homogeneous environment with streamlined change management and approval processes is a must.
- **Migration waves.** Another approach that works well for large migrations is a wave model. In this model, division of labor isn't nearly as clear. Teams dedicate themselves to the migration process execution of individual workloads. However, the nature of each sprint changes. In one sprint, the team may complete

assessment and architecture work. In another sprint, it may complete the migration work. In yet another sprint, the focus would be on optimization and production release. This approach allows a core team to stay aligned to workloads, seeing them through the process in its entirety. When using this approach, the diversity of skills and context switching could reduce the potential velocity of the team, slowing the migration effort. Additionally, roadblocks during approval cycles can cause significant delays. It is important to maintain options in the release backlog to keep the team moving during blocked periods, with this model. It is also important to cross-train team members and to ensure that skill sets align with the theme of each sprint.

Sprint backlog data points

The outcome of a sprint captures and documents the changes made to a workload, thus closing the change-management loop. When completed, at a minimum, the following should be documented. Throughout the execution of a sprint, this documentation should be completed in tandem with the technical work items.

- **Assets deployed.** Any assets deployed to the cloud to host the workload.
- **Remediation.** Any changes to the assets to prepare for cloud migration.
- **Configuration.** Chosen configuration of any assets deployed, including any references to configuration scripts.
- **Deployment model.** Approach used to deploy the asset to the cloud, including references to any deployment scripts or tools.
- **Architecture.** Documentation of the architecture deployed to the cloud.
- **Performance metrics.** Output of automated testing or business testing performed to validate performance at the time of deployment.
- **Unique requirements or configuration.** Any unique aspects of the deployment, configuration, or technical requirements necessary to operate the workload.
- **Operational approval.** Sign-off of validating operational readiness from the application owner and the IT operations staff responsible for managing the workload after deployment.
- **Architecture approval.** Sign-off from the workload owner and the cloud adoption team to validate any architecture changes required to host each asset.

Next steps

After change management approaches have been established, its time to address the final prerequisite, migration backlog review

Migration backlog review

Migration backlog review

3/6/2021 • 2 minutes to read • Edit Online

The actionable output during the planning phase is a migration backlog, which influences all of the prerequisites discussed so far. Development of the migration backlog should be completed as a first prerequisite. This article serves as a milestone to complete prerequisite activities. The cloud strategy team is accountable for the care and maintenance of the digital estate. However, the realization of the resultant backlog is the responsibility of every member of the migration effort. As a final prerequisite, the cloud strategy team and the cloud adoption team should review and understand the migration backlog. During that review, the members of both teams must gain sufficient knowledge to articulate the following key points in the migration backlog.

Business outcomes and metrics

Every member of the team should understand the desired business outcomes. Migrations take time. It's easy for team members to become distracted by urgent but less important activities during migration. Establishing and reinforcing the desired outcomes helps the team understand the priority and relative importance of the migration, enabling better decision-making over time.

Tracking migration progress is equally important to the motivation of the team and to continued stakeholder support. Progress can be tracked through migration KPIs and learning metrics. Regardless of how the effort is tracked, it is important that the team is aware of these metrics so that they can evaluate performance during subsequent iterations.

Business priorities

Sometimes, prioritizing one workload over another may seem illogical to the cloud adoption team. Understanding the business priorities that drove those decisions can help maintain the team's motivation. It also allows the team to make a stronger contribution to the prioritization process.

Core assumptions

The article on digital estate rationalization discusses the agility and time-saving impact of basic assumptions when evaluating a digital estate. To fully realize those values, the cloud adoption team needs to understand the assumptions and the reasons that they were established. That knowledge better equips the cloud adoption team to challenge those assumptions.

Next steps

With a general understanding of the digital estate and migration backlog, the team is ready to move beyond prerequisites and begin assessing workloads.

Assess workloads

Assess workloads and validate assumptions before migration

3/6/2021 • 3 minutes to read • Edit Online

Many of your existing workloads are ideal candidates for cloud migration, but not every asset is compatible with cloud platforms and not all workloads can benefit from hosting in the cloud. Digital estate planning allows you to generate an overall migration backlog of potential workloads to migrate. However, this planning effort is high-level. It relies on assumptions made by the cloud strategy team and does not dig deeply into technical considerations.

As a result, before migrating a workload to the cloud it's critical to assess the individual assets associated with that workload for their migration suitability. During this assessment, your cloud adoption team should evaluate technical compatibility, required architecture, performance/sizing expectations, and dependencies to ensure that the migrated workload can be deployed to the cloud effectively.

The *assess* process is the first of four incremental activities that occur within an iteration. As discussed in the prerequisite article regarding technical complexity and change management, a decision should be made in advance to determine how this phase is executed. In particular, will assessments be completed by the cloud adoption team during the same sprint as the actual migration effort? Alternatively, will a wave or factory model be used to complete assessments in a separate iteration? If the answer to this basic process question can't be answered by every member of the team, it may be wise to revisit the Prerequisites section.

Objective

Assess a migration candidate, evaluating the workload, associated assets, and dependencies prior to migration.

Definition of done

This process is complete when the following are known about a single migration candidate:

- The path from on-premises to cloud, including production promotion approach decision, has been defined.
- Any required approvals, changes, cost estimates, or validation processes have been completed to allow the cloud adoption team to execute the migration.

Accountability during assessment

The cloud adoption team is accountable for the entire assessment process. However, members of the cloud strategy team has a few responsibilities, as listed in the following section.

Responsibilities during assessment

In addition to the high-level accountability, there are actions that an individual or group needs to be directly responsible for. The following are a few activities that require assignments to responsible parties:

- **Business priority.** The team understands the purpose for migrating this workload, including any intended impact to the business.
  - A member of the cloud strategy team should carry final responsibility for this activity, under the direction of the cloud adoption team.
- **Stakeholder alignment.** The team aligns expectations and priorities with internal stakeholders, identifying success criteria for the migration. What does success look like post-migration?

- **Refined rationalization.** Evaluate the initial assumptions regarding rationalization. Should a different rationalization approach be used to migrate this specific workload?
- **Modernization decisions.** Regardless of the rationalization decision, should various assets in the workload be modernized to use PaaS-based solutions?
- **Cost.** The cost of the target architecture has been estimated, and the overall budget has been adjusted.
- **Migration support.** The team has decided how the technical work of the migration will be completed, including decisions regarding partner or Microsoft support.
- **Evaluation.** The workload is evaluated for compatibility and dependencies.
  - This activity should be assigned to a subject matter expert who is familiar with the architecture and operations of the candidate workload.
- **Architect.** The team has agreed on the final state architecture for the migrated workload.
- **Migration tooling.** Depending on modernization and architecture approaches, a variety of migration tools could be used to automate the migration. Based on the proposed architecture, will this migration use the best migration tools?
- **Backlog alignment.** The cloud adoption team reviews requirements and commits to the migration of the candidate workload. After commitment, the release backlog and iteration backlog are to be updated accordingly.
- **Work breakdown structure or work-back schedule.** The team establishes a schedule of major milestones identifying goals for when planning, implementation, and review processes are completed.
- **Final approval.** Any necessary approvers have reviewed the plan and have signed off on the approach to migrate the asset.
  - To avoid surprises later in the process, at least one representative of the business should be involved in the approval process.

Caution

This full list of responsibilities and actions can support large and complex migrations involving multiple roles with varying levels of responsibility, and requiring a detailed approval process. Smaller and simpler migration efforts may not require all of roles and actions described here. To determine which of these activities add value and which are unnecessary, your cloud adoption team and the cloud strategy team should use this complete process as part of your first workload migration. After the workload has been verified and tested, the team can evaluate this process and choose which actions to use moving forward.

Next steps

With a general understanding of the assessment process, you're ready to begin the process by classifying workloads.

Classify workloads

Workload classification before migration

8/11/2021 • 2 minutes to read • Edit Online

During each iteration of any migration process, one or more workloads will be migrated and promoted to production. Prior to either of those migration activities, it is important to classify each workload. Classification helps clarify governance, security, operations, and data management and analytics requirements.

The following guidance builds on the suggested tagging requirements outlined in Define your tagging strategy by adding important operations and governance elements.

In this article, we specifically suggest adding criticality and data sensitivity to your existing tagging standards. Each of these data points will help other teams understand which workloads may require additional attention or support.

Data sensitivity

As outlined in the article on data classification, data classification measures the impact that a data leak would have on the business or customers. The governance and security teams use data sensitivity or data classification as an indicator of security risks. During assessment, the cloud adoption team should evaluate the data classification for each workload targeted for migration and share that classification with supporting teams. Workloads that deal strictly in "public data" may not have any impact on supporting teams. However, as data moves further towards the "highly confidential" end of the spectrum, both governance and security teams will likely have a vested interest in participating in the assessment of the workload.

Work with your security and governance teams as early as possible to define the following items:

- A clear process for sharing any workloads on the backlog with sensitive data.
- An understanding of the governance requirements and security baseline required for various different levels of data sensitivity.
- Any impact data sensitivity may have on subscription design, management group hierarchies, or landing zone requirements.
- Any requirements for testing data classification, which may include specific tooling or defined scope of classification.

Mission criticality

As outlined in the article on workload criticality, the criticality of a workload is a measure of how significantly the business will be affected during an outage. This data point helps operations management and security teams evaluate risks regarding outages and breaches. During assessment, the cloud adoption team should evaluate mission criticality for each workload targeted for migration and share that classification with supporting teams. "Low" or "unsupported" workloads are likely to have little impact on the supporting teams. However, as workloads approach "mission critical" or "unit critical" classifications, their operational dependencies become more apparent.

Work with your security and operations teams as early as possible to define the following items:

- A clear process for sharing any workloads on the backlog with support requirements.
- An understanding of the operations management and resource consistency requirements for various different levels of criticality.
- Any impact criticality may have on subscription design, management group hierarchies, or landing zone requirements.

- Any requirements for documenting criticality, which might include specific traffic or usage reports, financial analyses, or other tools.

Next steps

Once workloads are properly classified, it's much easier to align business priorities.

Align business priorities

Business priorities: Maintaining alignment

3/6/2021 • 3 minutes to read • Edit Online

Transformation is often defined as a dramatic or spontaneous change. At the board level, change can look like a dramatic transformation. However, for those who work through the process of change in an organization, transformation is a bit misleading. Under the surface, transformation is better described as a series of properly executed transitions from one state to another.

The amount of time required to rationalize or transition a workload will vary, depending on the technical complexity involved. However, even when this process can be applied to a single workload or group of applications quickly, it takes time to produce substantial changes among a user base. It takes longer for changes to propagate through various layers of existing business processes. If transformation is expected to shape behavior patterns in consumers, the results can take longer to produce significant results.

Unfortunately, the market doesn't wait for businesses to transition. Consumer behavior patterns change on their own, often unexpectedly. The market's perception of a company and its products can be swayed by social media or a competitor's positioning. Fast and unexpected market changes require companies to be nimble and responsive.

The ability to execute processes and technical transitions requires a consistent, stable effort. Quick decisions and nimble actions are needed to respond to market conditions. These two are at odds, making it easy for priorities to fall out of alignment. This article describes approaches to maintaining transitional alignment during migration efforts.

How can business and technical priorities stay aligned during a migration?

The cloud adoption team and the cloud governance team focus on the execution of the current iteration and current release. Iterations provide stable increments of technical work, thus avoiding costly disruptions that would otherwise slow the progress of migration efforts. Releases ensure that the technical effort and energy stay focused on the business objectives of the workload migration. A migration project could require many releases over an extended period. By the time it is completed, market conditions have likely changed significantly.

In parallel, the cloud strategy team focuses on executing the business change plan and preparing for the next release. The cloud strategy team generally looks at least one release ahead, and it monitors for changing market conditions and adjusts the migration backlog accordingly. This focus of managing transformation and adjusting the plan creates natural pivots around the technical work. When business priorities change, adoption is only one release behind, creating technical and business agility.

Business alignment questions

The following questions can help the cloud strategy team shape and prioritize the migration backlog to help ensure that the transformation effort best aligns with current business needs.

- Has the cloud adoption team identified a list of workloads ready for migration?
- Has the cloud adoption team selected a single candidate for an initial migration from that list of workloads?
- Do the cloud adoption team and the cloud governance team have all of the necessary data regarding the workload and cloud environment to be successful?
- Does the candidate workload deliver the most relevant impact for the business in the next release?

- Are there other workloads that are better candidates for migration?

Tangible actions

During the execution of the business change plan, the cloud strategy team monitors for positive and negative results. When those observations require technical change, the adjustments are added as work items to the release backlog to be prioritized in the next iteration.

When the market changes, the cloud strategy team works with the business to understand how to best respond to the changes. When that response requires a change in migration priorities, the migration backlog is adjusted. This moves up workloads that were previously lower in priority.

Next steps

With properly aligned business priorities, the cloud adoption team can confidently begin to evaluate workloads to develop architecture and migration plans.

Evaluate workloads

Evaluate workload readiness

3/6/2021 • 3 minutes to read • Edit Online

This activity focuses on evaluating readiness of a workload to migrate to the cloud. During this activity, the cloud adoption team validates that all assets and associated dependencies are compatible with the chosen deployment model and cloud provider. During the process, the team documents any efforts required to remediate compatibility issues.

Evaluation assumptions

Most of the content discussing principles in the Cloud Adoption Framework is cloud agnostic. However, the readiness evaluation process must be largely specific to each specific cloud platform. The following guidance assumes an intention to migrate to Azure. It also assumes use of Azure Migrate (also known as Azure Site Recovery) for replication activities. For alternative tools, see Replication options.

This article doesn't capture all possible evaluation activities. It is assumed that each environment and business outcome will dictate specific requirements. To help accelerate the creation of those requirements, the remainder of this article shares a few common evaluation activities related to infrastructure, database, and network evaluation.

Common infrastructure evaluation activities

- VMware requirements: review the Azure Site Recovery requirements for VMware.
- Hyper-V requirements: review the Azure Site Recovery requirements for Hyper-V.

Be sure to document any discrepancies in host configuration, replicated VM configuration, storage requirements, or network configuration.

Common database evaluation activities

- Document the recovery point objectives (RPOs) and recovery time objectives (RTOs) of the current database deployment. These are used during architecture activities to aid in decision-making.
- Document any requirements for high-availability configuration. For assistance understanding SQL Server requirements, see the SQL Server high availability solutions guide.
- Evaluate PaaS compatibility. The Azure data migration guide maps on-premises databases to compatible Azure PaaS solutions, like Azure Cosmos DB, Azure SQL Database Azure Database for MySQL, Azure Database for PostgreSQL, or Azure Database for MariaDB.
- When PaaS compatibility is an option without the need for any remediation, consult the team responsible for architecture activities. PaaS migrations can produce significant time savings and reductions in the total cost of ownership (TCO) of most cloud solutions.
- When PaaS compatibility is an option but remediation is required, consult the teams responsible for architecture activities and remediation activities. In many scenarios, the advantages of PaaS migrations for database solutions can outweigh the increase in remediation time.
- Document the size and rate of change for each database to be migrated.
- When possible, document any applications or other assets that make calls to each database.

> **NOTE**
>
> Synchronization of any asset consumes bandwidth during the replication processes. A very common pitfall is to overlook the bandwidth consumption required to keep assets synchronized between the point of replication and release. Databases are common consumers of bandwidth during release cycles, and databases with large storage footprints or a high rate of change are especially concerning. Consider an approach of replicating the data structure, with controlled updates before user acceptance testing (UAT) and release. In such scenarios, alternatives to Azure Site Recovery may be more appropriate. For more information, see guidance from the Azure Database Migration Guide.

Common network evaluation activities

- Calculate the total storage for all VMs to be replicated during the iterations leading up to a release.
- Calculate the drift or change rate of storage for all VMs to be replicated during the iterations leading up to a release.
- Calculate the bandwidth requirements needed for each iteration by summing total storage and drift.
- Calculate unused bandwidth available on the current network to validate per iteration alignment.
- Document bandwidth needed to reach anticipated migration velocity. If any remediation is required to provide necessary bandwidth, notify the team responsible for remediation activities.

> **NOTE**
>
> Total storage directly affects bandwidth requirements during initial replication. However, storage drift continues from the point of replication until release. This means that drift has a cumulative effect on available bandwidth.

Next steps

After the evaluation of a system is complete, the outputs feed the development of a new cloud architecture.

Architect workloads prior to migration

Architect workloads prior to migration

5/18/2021 • 4 minutes to read • Edit Online

This article expands on the assessment process by reviewing activities associated with defining the architecture of a workload within a given iteration. As discussed in the article on incremental rationalization, some architectural assumptions are made during any business transformation that requires a migration. This article clarifies those assumptions, shares a few roadblocks that can be avoided, and identifies opportunities to accelerate business value by challenging those assumptions. This incremental model for architecture allows teams to move faster and to obtain business outcomes sooner.

Architecture assumptions prior to migration

The following assumptions are typical for any migration effort:

- **IaaS.** It is commonly assumed that migrating workloads primarily involves the movement of virtual machines from a physical datacenter to a cloud datacenter via an IaaS migration, requiring a minimum of redevelopment or reconfiguration. This approach is known as a *lift and shift* migration. (Exceptions follow.)
- **Architecture consistency.** Changes to core architecture during a migration considerably increase complexity. Debugging a changed system on a new platform introduces many variables that can be difficult to isolate. For this reason, workloads should undergo only minor changes during migration and any changes should be thoroughly tested.
- **Retirement test.** Migrations and the hosting of assets consume operational and potential capital expenses. It is assumed that any workloads being migrated have been reviewed to validate ongoing usage. The choice to retire unused assets produces immediate cost savings.
- **Resize assets.** It is assumed that few on-premises assets are fully using the allocated resources. Prior to migration, it is assumed that assets will be resized to best fit actual usage requirements.
- **Business continuity and disaster recovery (BCDR) requirements.** It is assumed that an agreed-on SLA for the workload has been negotiated with the business prior to release planning. These requirements are likely to produce minor architecture changes.
- **Migration downtime.** Likewise, downtime to promote the workload to production can have an adverse effect on the business. Sometimes, the solutions that must transition with minimum downtime need architecture changes. It is assumed that a general understanding of downtime requirements has been established prior to release planning.
- **User traffic patterns.** Existing solutions may depend on existing network routing patterns. These patterns could slow performance considerably. Further, introduction of new hybrid wide area network (WAN) solutions can take weeks or even months. Prior to migration, it is assumed that your landing zones have already considered the relevant traffic patterns and changes to any core infrastructure services.

Mitigating potential roadblocks

The itemized assumptions can create roadblocks that could slow progress or cause later pain points. The following are a few roadblocks to watch for, prior to the release:

- **Paying for technical debt.** Some aging workloads carry with them a high amount of technical debt. Technical debt can lead to long-term challenges by increasing hosting costs with any cloud provider. When technical debt unnaturally increases hosting costs, alternative architectures should be evaluated.
- **Improving reliability.** Standard operations baselines provide a degree of reliability and recovery in the cloud. But, some workload teams may require higher SLAs which could lead to architectural changes.
- **High-cost workloads.** During migration, all assets should be optimized to align sizing with actual usage.

But, some workloads may require architectural modifications to address specific cost concerns.

- **Performance requirements.** When workload performance has a direct business impact, extra architectural consideration may be required.
- **Secure applications.** Security requirements tend to be implemented centrally and applied to all workloads in the portfolio. But, some workloads may have specific security requirements that could lead to architectural changes.

Each of the above criteria serve as indicators of potential migration roadblocks. The above criteria is usually addressed after a workload is migrated. But if any of those criteria are required before a workload is migrated, it should be removed from the migration wave and evaluated individually.

The Microsoft Azure Well-Architected Framework and Microsoft Azure Well-Architected Review can help guide those conversations with the technical owner of a specific workload to consider alternative options for deploying the workload. Those workloads would then be classified as a rearchitecture effort in your cloud adoption plan. Given the extra time required to rearchitect a workload, these alternative workload adoption paths should not be considered part of the migration process.

Accelerate business value

Some scenarios could require an different architecture than the assumed IaaS rehosting strategy. The following are a few examples:

- **PaaS modernization.** Some technology assets can be migrated to more modern Platform as a Service solutions, reducing risk during migration. Automated migration tools like Azure Migrate suggest and even automate modernization opportunities. A few examples of in-flight modernization would include low risk changes like the use of Azure Database Migration Service (DMS) to modernize databases. For a list of approaches that could benefit from a PaaS conversion, see the article on evaluating assets.
- **Scripted deployments/DevOps.** If a workload has an existing DevOps deployment or other forms of scripted deployment, the cost of changing those scripts could be lower than the cost of migrating the asset.
- **Remediation efforts.** The remediation efforts required to prepare a workload for migration can be extensive. In some cases, it makes more sense to modernize the solution than it does to remediate underlying compatibility issues.

In each of these itemized scenarios, an alternative architecture could be the best possible solution.

Next steps

After the new architecture is defined, accurate cost estimations can be calculated.

Estimate cloud costs

Estimate cloud costs

3/6/2021 • 2 minutes to read • Edit Online

During migration, there are several factors that can affect decisions and execution activities. To help understand which of those options are best for different situations, this article discusses various options for estimating cloud costs.

Digital estate size

The size of your digital estate directly affects migration decisions. Migrations that involve fewer than 250 VMs can be estimated much more easily than a migration involving 10,000+ VMs. It's highly recommended that you select a smaller workload as your first migration. This gives your team a chance to learn how to estimate the costs of a simple migration effort before attempting to estimate larger and more complicated workload migrations.

However, note that smaller, single-workload, migrations can still involve a widely varying amount of supporting assets. If your migration involves under 1,000 VMs, a tool like Azure Migrate is likely sufficient to gather data on the inventory and forecast costs. Additional cost-estimate tooling options are described in the article on digital estate cost calculations.

For 1,000+ unit digital estates, it's still possible to break down an estimate into four or five actionable iterations, making the estimation process manageable. For larger estates or when a higher degree of forecast accuracy is required, a more comprehensive approach, like that outlined in the Digital estate section of the Cloud Adoption Framework, will likely be required.

Accounting models

Accounting models

If you're familiar with traditional IT procurement processes, estimation in the cloud may seem foreign. When adopting cloud technologies, acquisition shifts from a rigid, structured capital expense model to a fluid operating expense model. In the traditional capital expense model, the IT team would attempt to consolidate buying power for multiple workloads across various programs to centralize a pool of shared IT assets that could support each of those solutions. In the operating expenses cloud model, costs can be directly attributed to the support needs of individual workloads, teams, or business units. This approach allows for a more direct attribution of costs to the supported internal customer. When estimating costs, it's important to first understand how much of this new accounting capability will be used by the IT team.

For those wanting to replicate the legacy capital expense approach to accounting, use the outputs of either approach suggested in the Digital estate size section above to get an annual cost basis. Next, multiply that annual cost by the company's typical hardware refresh cycle. Hardware refresh cycle is the rate at which a company replaces aging hardware, typically measured in years. Annual run rate multiplied by hardware refresh cycle creates a cost structure similar to a capital expense investment pattern.

Next steps

After estimating costs, migration can begin. However, it would be wise to review partnership and support options before beginning any migration.

Understand partnership and support options

Understand partnership and support options

7/15/2021 • 5 minutes to read • Edit Online

During migration, the cloud adoption team performs the actual migration of workloads to the cloud. Unlike the collaborative and problem-solving tasks when defining the digital estate or building the core cloud infrastructure, migration tends to be a series of repetitive execution tasks. Beyond the repetitive aspects, there are likely testing and tuning efforts that require deep knowledge of the chosen cloud provider. The repetitive nature of this process can sometimes be best addressed by a partner, reducing strain on full-time staff. Additionally, partners may be able to better align deep technical expertise when the repetitive processes encounter execution anomalies.

Partners tend to be closely aligned with a single cloud vendor or a small number of cloud vendors. To better illustrate partnership options, the remainder of this article assumes that Microsoft Azure is the chosen cloud provider.

During plan, build, or migrate, a company generally has four execution partnership options:

- **Guided self-service.** The existing technical team executes the migration, with help from Microsoft.
- **FastTrack for Azure.** Use the Microsoft FastTrack for Azure program to accelerate migration.
- **Solutions partner.** Get connected with Azure partners or cloud solution providers (CSPs) to accelerate migration.
- **Supported self-service.** Execution is completed by the existing technical staff with support from Microsoft.

Guided self-service

If an organization is planning an Azure migration on its own, Microsoft is always there to assist throughout the journey. To help fast-track migration to Azure, Microsoft and its partners have developed an extensive set of architectures, guides, tools, and services to reduce risk and to speed migration of virtual machines, applications, and databases. These tools and services support a broad selection of operating systems, programming languages, frameworks, and databases.

- **Assessment and migration tools.** Azure provides a wide range of tools to be used in different phases for your cloud transformation, including assessing your existing infrastructure. For more information, refer to the "assess" section in the "migration" chapter that follows.
- Microsoft Cloud Adoption Framework. This framework presents a structured approach to cloud adoption and migration. It is based on best practices across many Microsoft-supported customer engagements and is organized as a series of steps, from architecture and design to implementation. For each step, supporting guidance helps you with the design of your application architecture.
- Cloud design patterns. Azure provides some useful cloud design patterns for building reliable, scalable, secure workloads in the cloud. Each pattern describes the problem that the pattern addresses, considerations for applying the pattern, and an example based on Azure. Most of the patterns include code samples or snippets that show how to implement the pattern on Azure. However, they're relevant to any distributed system, whether hosted on Azure or on other cloud platforms.
- Cloud fundamentals. Fundamentals help teach the basic approaches to implementation of core concepts. This guide helps technicians think about solutions that go beyond a single Azure service.
- Example scenarios. The guide provides references from real customer implementations, outlining the tools, approaches, and processes that past customers have followed to accomplish specific business goals.
- Reference architectures. Reference architectures are arranged by scenario, with related architectures grouped together. Each architecture includes best practices, along with considerations for scalability, availability,

manageability, and security. Most also include a deployable solution.

FastTrack for Azure

FastTrack for Azure provides direct assistance from Azure engineers, working hand in hand with partners, to help customers build Azure solutions quickly and confidently. FastTrack brings best practices and tools from real customer experiences to guide customers from setup, configuration, and development to production of Azure solutions, including:

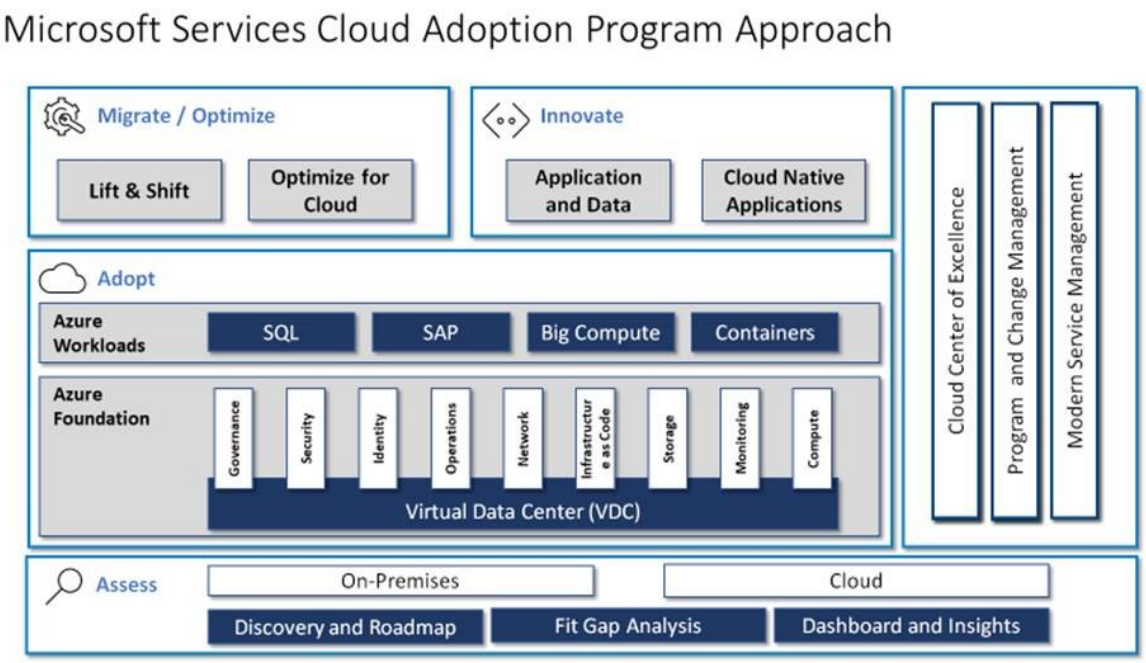
- Datacenter migration
- Windows Server on Azure
- Linux on Azure
- SAP on Azure
- Business continuity and disaster recovery (BCDR)
- High-performance computing
- Cloud-native applications
- DevOps
- Application modernization
- Cloud-scale analytics
- Intelligent applications
- Intelligent agents
- Data modernization to Azure
- Security and management
- Globally distributed data
- Azure Virtual Desktop
- Azure Marketplace
- Fundamentals and governance

During a typical FastTrack for Azure engagement, Microsoft helps to define the business vision to plan and develop Azure solutions successfully. The team assesses architectural needs and provides guidance, design principles, tools, and resources to help build, deploy, and manage Azure solutions. The team matches skilled partners for deployment services on request and periodically checks in to ensure that deployment is on track and to help remove blockers.

The main phases of a typical FastTrack for Azure engagement are:

- **Discovery.** Identify key stakeholders, understand the goal or vision for problems to be solved, and then assess architectural needs.
- **Solution enablement.** Learn design principles for building applications, review architecture of applications and solutions, and receive guidance and tools to drive proof-of-concept work through to production.
- **Continuous partnership.** Azure engineers and program managers check in every so often to ensure that deployment is on track and to help remove blockers.

Microsoft Services offerings aligned to Cloud Adoption Framework approaches



Assess: Microsoft Services uses a unified data- and tool-driven approach consisting of architectural workshops, Azure real-time information, security and identity threat models and various tools to provide insights into challenges, risks, recommendations and issues to an existing Azure environment with a key outcome such as high-level modernization roadmap.

Adopt: Using the Azure cloud foundation from Microsoft Services, establish your core Azure designs, patterns and governance architecture by mapping your requirements to the most appropriate reference architecture and plan, design and deploy the infrastructure, management, security, and identity required for workloads.

Migrate/optimize: The cloud modernization solution from Microsoft Services offers a comprehensive approach to move applications and infrastructure to Azure, as well as to optimize and modernize after cloud deployment, backed by streamlined migration.

Innovate: The cloud center of excellence (CCoE) solution from Microsoft Services offers a DevOps coaching engagement and uses DevOps principles combined with prescriptive cloud-native service management and security controls to help drive business innovation, increase agility, and reduce time to value within a secure, predictable, and flexible services delivery and operations management capability.

Azure Support

If you have questions or need help, create a support request. If your support request requires deep technical guidance, visit Azure Support plans to align the best plan for your needs.

Azure solutions partner

Microsoft certified solution providers specialize in providing modern customer solutions based on Microsoft technologies across the world. Optimize your business in the cloud with help from an experienced partner.

Get help from partners with ready-made or custom Azure solutions and partners who can help deploy and manage those solutions:

- Find a cloud solutions partner. A certified CSP can help take full advantage of the cloud by assessing business goals for cloud adoption, identifying the right cloud solution that meets business needs and helps the business become more agile and efficient.
- Find an Azure Expert Managed Service Provider (MSPs). MSPs help businesses transition to Azure by guiding all aspects of the cloud journey. From consulting to migrations and operations management, cloud MSPs

show customers all the benefits that come with cloud adoption. They also act as a one-stop shop for common support, provisioning, and the billing experience, all with a flexible pay-as-you-go business model.

Next steps

After a partner and support strategy is selected, the release and iteration backlogs can be updated to reflect planned efforts and assignments.

Manage change using release and iteration backlogs

Manage change in an incremental migration effort

3/6/2021 • 2 minutes to read • Edit Online

This article assumes that migration processes are incremental in nature, running parallel to the govern process. However, the same guidance could be used to populate initial tasks in a work breakdown structure for traditional waterfall change management approaches.

Release backlog

A *release backlog* consists of a series of assets (VMs, databases, files, and applications, among others) that must be migrated before a workload can be released for production usage in the cloud. During each iteration, the cloud adoption team documents and estimates the efforts required to move each asset to the cloud. See the "iteration backlog" section that follows.

Iteration backlog

An *iteration backlog* is a list of the detailed work required to migrate a specific number of assets from the existing digital estate to the cloud. The entries on this list are often stored in an agile management tool, like Azure DevOps, as work items.

Prior to starting the first iteration, the cloud adoption team specifies an iteration duration, usually two to four weeks. This time box is important to create a start and finish time period for each set of committed activities. Maintaining consistent execution windows makes it easy to gauge velocity (pace of migration) and alignment to changing business needs.

Prior to each iteration, the team reviews the release backlog, estimating the effort and priorities of assets to be migrated. It then commits to deliver a specific number of agreed-on migrations. After this is agreed to by the cloud adoption team, the list of activities becomes the *current iteration backlog*.

During each iteration, team members work as a self-organizing team to fulfill commitments in the current iteration backlog.

Next steps

After an iteration backlog is defined and accepted by the cloud adoption team, change management approvals can be finalized.

Approve architecture changes prior to migration

Approve architecture changes before migration

6/18/2021 • 6 minutes to read • Edit Online

During the assess process of migration, each workload is evaluated, architected, and estimated to develop a future state plan for the workload. Some workloads can be migrated to the cloud with no change to the architecture. Maintaining on-premises configuration and architecture can reduce risk and streamline the migration process. Unfortunately, not every application can run in the cloud without changes to the architecture. When architecture changes are required, this article can help classify the change and can provide some guidance on the proper approval activities.

Business impact and approval

During migration, some things are likely to change in ways that impact the business. Although change sometimes can't be avoided, surprises as a result of undisclosed or undocumented changes should be. To maintain stakeholder support throughout the migration effort, it's important to avoid surprises as well as negative disruption to business or operations. Surprising application owners or business stakeholders can slow or halt a cloud adoption effort.

Prior to migration, it is important to prepare the workload's business owner for any changes that could affect business processes, such as changes to:

- Service-level agreements.
- Access patterns or security requirements that impact the end user.
- Data retention practices.
- Core application performance.
- Operational practice changes such as support model, change model, cost model changes.
- Anticipated user experience changes.

Even when a workload can be migrated with minimal to no change, there could still be a business impact. Replication processes can slow the performance of production systems. Changes to the environment in preparation for migration have the potential to cause routing or network performance limitations. There are many additional impacts that could result from replication, staging, or promotion activities.

Regular approval activities can help minimize or avoid surprises as a result of change or performance-driven business impacts. The cloud adoption team should establish a suitable change approval process at the end of the assessment process, before beginning the migration process.

Existing culture

Your IT teams likely have existing mechanisms for managing change involving your on-premises assets. Typically these mechanisms are governed by traditional Information IT Service Management (ITSM) change management processes. In many enterprise migrations, these processes typically involve a change advisory board (CAB) that's responsible for reviewing, documenting, and approving all IT-related requests for changes (RFC).

The CAB generally includes experts from multiple IT and business teams, offering a variety of perspectives and detailed review for all IT-related changes. A CAB approval process is a proven way to reduce risk and minimize the business impact of changes involving stable workloads managed by IT operations.

Technical approval

Organizational readiness for the approval of technical change is among the most common reasons for cloud migration failure. More projects are stalled by a series of technical approvals than any deficit in a cloud platform. Preparing the organization for technical change approval is an important requirement for migration success. The following are a few best practices to ensure that the organization is ready for technical approval.

ITSM change advisory board challenges

Every change management approach has its own set of controls and approval processes. Migration is a series of continuous changes that start with a high degree of ambiguity and develop additional clarity through the course of execution. As such, migration is best governed by agile-based change management approaches, with the cloud strategy team serving as a product owner.

However, the scale and frequency of change during a cloud migration doesn't fit well with the nature of change advisory boards. The requirements of a CAB approval can risk the success of a migration, slowing or stopping the effort. Further, in the early stages of migration, ambiguity is high and subject matter expertise tends to be low. For the first several workload migrations or releases, the cloud adoption team is often in a learning mode. As such, it could be difficult for the team to provide the types of data needed to pass a typical CAB approval.

The following best practices can help the CAB maintain a degree of comfort during migration without become a painful blocker.

Standardize and pre-approve change through standard change models

It is tempting for a cloud adoption team to consider detailed architectural decisions for each workload being migrated to the cloud. It is equally tempting to use cloud migration as a catalyst to refactor past architectural decisions. For organizations that are migrating a few hundred VMs or a few dozen workloads, either approach can be properly managed. When migrating a datacenter consisting of 1,000 or more assets, each of these approaches is considered a high-risk antipattern that significantly reduces the likelihood of success. Modernizing, refactoring, and rearchitecting every application requires a diverse skill set and a wide variety of changes, and these tasks create dependencies on human efforts at scale. Each of these dependencies injects risk into the migration effort.

The article on digital estate rationalization discusses the agility and time-saving impact of basic assumptions when rationalizing a digital estate. By choosing a default (standard) rationalization approach to govern the migration effort, the cloud advisory board or product owner can review and approve the application of one change to a long list of workloads. This reduces technical approval of each workload to those that require a significant architecture change to be cloud compatible. Additionally to this, once the assessment process for the migration has been performed, there should be a clear idea of what changes will be required going forward and this initial list could be used to identify common change types or categories of change (standard changes). The cloud adoption team should meet with the relevant change managers and approvers early in the project to agree on the categories and types of changes in the list, and then establish appropriate pre-approved standard change models that will enable changes to be made without the need of the CAB approval, provided certain criteria or prerequisites for the changes are met. For example, changes can be preapproved for none production, or changes will require a tested backout plan, or changes might need a peer review.

> **IMPORTANT**
> An organizations ITSM change processes might need to be realigned to work effectively and iteratively with the fast pace of change expected in the cloud. See Technical complexity for more guidance.

It needs to be understood that this realignment does not happen overnight. The use of standard pre-approved changes will enable a lot more flexibility around the change process, but they need to be carefully managed to ensure the do not result in negative technical or business impact. If a standard change fails and has negative impact technically or operationally, then that pre-approved status should be removed and the change will need to go through a proper risk review and CAB the next time it is required. This practice will help increase confidence with change managers and approvers in the use of standard pre-approved changes.

Seek approval early

When possible, technical change should be detected and documented during the assessment process as mentioned above. Regardless of approval processes, the cloud adoption team should engage approvers early. The sooner that change approval can begin, the less likely an approval process is to block migration activities.

Clarify expectations and roles of approvers

Before the first workload is assessed, the cloud strategy team should document and communicate the expectations of anyone involved in the approval of change. This simple activity can avoid costly delays when the cloud adoption team is fully engaged.

Establishment of a Cloud Change Authority

If possible, a Cloud Change Authority could be established to facilitate triage and potential approvals of cloud changes. Having an authority focusing on the approval of cloud changes can ensure the correct subject matter expertise and context is applied to assess cloud change risk. It can also ensure that additional pressure and risk of a bottleneck at the CAB authority is minimized. Establishing this will require an agreement with change managers, approvers, and likely clear criteria about what can be approved through the Cloud Change Authority, and what will need the CAB authorities approval.

Next steps

With the help of these best practices, it should be easier to integrate proper, low-risk approval into migration efforts. After workload changes are approved, the cloud adoption team is ready to migrate workloads.

Migrate workloads

Deploy workloads

3/6/2021 • 2 minutes to read • Edit Online

After workloads have been assessed, they can be deployed to the cloud or staged for release. This series of articles explains the various activities that may be involved in this phase of migration effort.

Objective

The objective of a migration is to migrate a single workload to the cloud.

Definition of done

The migration phase is complete when a workload is staged and ready for testing in the cloud, including all dependent assets required for the workload to function. During the optimize process, the workload is prepared for production usage.

This *definition of done* can vary, depending on your testing and release processes. The next article in this series covers deciding on a promotion model and can help you understand when it would be best to promote a migrated workload to production.

Accountability during migration

The cloud adoption team is accountable for the entire migration process. However, members of the cloud strategy team have a few responsibilities, as discussed in the following section.

Responsibilities during migration

In addition to the high-level accountability, there are actions that an individual or group needs to be directly responsible for. The following are a few activities that require assignments to responsible parties:

- **Remediation.** Resolve any compatibility issues that prevent the workload from being migrated to the cloud.
  - As discussed in the prerequisite article regarding technical complexity and change management, a decision should be made in advance to determine how this activity is to be executed. In particular, will remediation be completed by the cloud adoption team during the same sprint as the actual migration effort? Alternatively, will a wave or factory model be used to complete remediation in a separate iteration? If the answer to this basic process question can't be answered by every member of the team, it may be wise to revisit the section on prerequisites.
- **Replication.** Create a copy of each asset in the cloud to synchronize VMs, data, and applications with resources in the cloud.
  - Depending on the promotion model, different tools may be required to complete this activity.
- **Staging.** After all assets for a workload have been replicated and verified, the workload can be staged for business testing and execution of a business change plan.

Next steps

With a general understanding of the migration process, you're ready to decide on a promotion model.

Decide on a promotion model

Promotion models: Single-step, staged, or flight

3/6/2021 • 5 minutes to read • Edit Online

Workload migration is often discussed as a single activity. In practice, it's a collection of smaller activities that facilitate the movement of a digital asset to the cloud. One of the last activities in a migration is the promotion of an asset to production. Promotion is the point at which the production system changes for end users. It can often be as simple as changing the network routing, redirecting end users to the new production asset. Promotion is also the point at which IT operations or cloud operations change the focus of operational management processes from the previous production system to the new production systems.

There are several promotion models. This article outlines three of the most common ones used in cloud migrations. The choice of a promotion model changes the activities seen within the migrate and optimize processes. As such, promotion model should be decided early in a release.

Impact of promotion model on migrate and optimize activities

In each of the following promotion models, the chosen migration tool replicates and stages the assets that make up a workload. After staging, each model treats the asset a bit differently.

- **Single-step promotion.** In a *single-step* promotion model, the staging process doubles as the promotion process. After all assets are staged, end-user traffic is rerouted and staging becomes production. In such a case, promotion is part of the migration process. This is the fastest migration model. However, this approach makes it more difficult to integrate robust testing or optimization activities. Further, this type of model assumes that the migration team has access to the staging and production environment, which compromises separation of duties in some environments.

  > **NOTE**
  > The table of contents for this site lists the promotion activity as part of the optimize process. In a single-step model, promotion occurs during the migration phase. When using this model, roles and responsibilities should be updated to reflect this.

- **Staged.** In a *staged* promotion model, the workload is considered migrated after it is staged, but it is not yet promoted. Prior to promotion, the migrated workload undergoes a series of performance tests, business tests, and optimization changes. It is then promoted at a future date in conjunction with a business test plan. This approach improves the balance between cost and performance, while making it easier to obtain business validation.
- **Flight.** The *flight* promotion model combines single-step and staged models. In a flight model, the assets in the workload are treated like production after landing in staging. After a condensed period of automated testing, production traffic is routed to the workload. However, it is a subset of the traffic. That traffic serves as the first flight of production and testing. Assuming the workload performs from a feature and performance perspective, additional traffic is migrated. After all production traffic has been moved onto the new assets, the workload is considered fully promoted.

The chosen promotion model affects the sequence of activities to be performed. It also affects the roles and responsibilities of the cloud adoption team. It may even impact the composition of a sprint or multiple sprints.

Single-step promotion

This model uses migration automation tools to replicate, stage, and promote assets. The assets are replicated

into a contained staging environment controlled by the migration tool. After all assets have been replicated, the tool can execute an automated process to promote the assets into the chosen subscription in a single step. While in staging, the tool continues to replicate the asset, minimizing loss of data between the two environments. After an asset is promoted, the linkage between the source system and the replicated system is severed. In this approach, if additional changes occur in the initial source systems, the changes are lost.

Pros. Positive benefits of this approach include:

- This model introduces less change to the target systems.
- Continuous replication minimizes data loss.
- If a staging process fails, it can quickly be deleted and repeated.
- Replication and repeated staging tests enable an incremental scripting and testing process.

Cons. Negative aspects of this approach include:

- Assets staged within the tools-isolated sandbox don't allow for complex testing models.
- During replication, the migration tool consumes bandwidth in the local datacenter. Staging a large volume of assets over an extended duration has an exponential impact on available bandwidth, hurting the migration process and potentially affecting performance of production workloads in the on-premises environment.

Staged promotion

In this model, the staging sandbox managed by the migration tool is used for limited testing purposes. The replicated assets are then deployed into the cloud environment, which serves as an extended staging environment. The migrated assets run in the cloud, while additional assets are replicated, staged, and migrated. When full workloads become available, richer testing is initiated. When all assets associated with a subscription have been migrated, the subscription and all hosted workloads are promoted to production. In this scenario, there is no change to the workloads during the promotion process. Instead, the changes tend to be at the network and identity layers, routing users to the new environment and revoking access of the cloud adoption team.

Pros. Positive benefits of this approach include:

- This model provides more accurate business testing opportunities.
- The workload can be studied more closely to better optimize performance and cost of the assets.
- A larger numbers of assets can be replicated within similar time and bandwidth constraints.

Cons. Negative aspects of this approach include:

- The chosen migration tool can't facilitate ongoing replication after migration.
- A secondary means of data replication is required to synchronize data platforms during the staged time frame.

Flight promotion

This model is similar to the staged promotion model. However, there is one fundamental difference. When the subscription is ready for promotion, end-user routing happens in stages or flights. At each flight, additional users are rerouted to the production systems.

Pros. Positive benefits of this approach include:

- This model mitigates the risks associated with a big migration or promotion activity. Errors in the migrated solution can be identified with less impact to business processes.
- It allows for monitoring of workload performance demands in the cloud environment for an extended duration, increasing accuracy of asset-sizing decisions.
- Larger numbers of assets can be replicated within similar time and bandwidth constraints.

Cons. Negative aspects of this approach include:

- The chosen migration tool can't facilitate ongoing replication after migration.
- A secondary means of data replication is required to synchronize data platforms during the staged time frame.

Next steps

After a promotion model is defined and accepted by the cloud adoption team, remediation of assets can begin.

Remediating assets prior to migration

Remediate assets prior to migration

3/6/2021 • 4 minutes to read • Edit Online

During the assessment process of migration, the team seeks to identify any configurations that would make an asset incompatible with the chosen cloud provider. *Remediate* is a checkpoint in the migration process to ensure that those incompatibilities have been resolved. This article discusses a few common remediation tasks for reference. It also establishes a skeleton process for deciding whether remediation is a wise investment.

Common remediation tasks

In any corporate environment, technical debt exists. Some of this is healthy and expected. Architecture decisions that were well suited for an on-premises environment may not be entirely suitable in a cloud platform. In either case, common remediation tasks may be required to prepare assets for migration. The following are a few examples:

- **Minor host upgrades.** Occasionally, an outdated host needs to be upgraded prior to replication.
- **Minor guest OS upgrades.** It is more likely that an OS will need patching or upgrading prior to replication.
- **SLA modifications.** Backup and recovery change significantly in a cloud platform. It is likely that assets will need minor modifications to their backup processes to ensure continued function in the cloud.
- **PaaS migration.** In some cases, a PaaS deployment of a data structure or application may be required to accelerate deployment. Minor modifications may be required to prepare the solution for PaaS deployment.
- **PaaS code changes.** It is not uncommon for custom applications to require minor code modifications to be PaaS ready. Examples could include methods that write to local disk or use of in-memory session state, among others.
- **Application configuration changes.** Migrated applications may require changes to variables, such as network paths to dependent assets, service account changes, or updates to dependent IP addresses.
- **Minor changes to network paths.** Routing patterns may need to be modified to properly route user traffic to the new assets.

> **NOTE**
> This isn't production routing to the new assets, but rather configuration to allow for proper routing to the assets in general.

Large-scale remediation tasks

When a datacenter is properly maintained, patched, and updated, there is likely to be little need for remediation. Remediation-rich environments tend to be common among large enterprises, organizations that have been through large IT downsizing, some legacy managed service environments, and acquisition-rich environments. In each of these types of environments, remediation may consume a large portion of the migration effort. When the following remediation tasks frequently appear and are negatively affecting migration speed or consistency, it may be wise to break out remediation into a parallel effort and team (similar to how cloud adoption and cloud governance run in parallel).

- **Frequent host upgrades.** When large numbers of hosts must be upgraded to complete the migration of a workload, the migration team is likely to suffer from delays. It may be wise to break out affected applications and address the remediation steps prior to including affected applications in any planned releases.
- **Frequent guest OS upgrade.** Large enterprises commonly have servers running on outdated versions of Linux or Windows. Aside from the apparent security risks of operating an outdated OS, there are also

incompatibility issues that prevent affected workloads from being migrated. When a large number of VMs require OS remediation, it may be wise to break out these efforts into a parallel iteration.

- **Major code changes.** Older custom applications may require significantly more modifications to prepare them for PaaS deployment. When this is the case, it may be wise to remove them from the migration backlog entirely, managing them in a wholly separate program.

Decision framework

Because remediation for smaller workloads can be straightforward, you should choose a smaller workload for your initial migration. However, as your migration efforts mature and you begin to tackle larger workloads, remediation can be a time consuming and costly process. For example, remediation efforts for a Windows Server 2003 migration involving a 5,000+ VM pool of assets can delay a migration by months. When such large-scale remediation is required, the following questions can help guide decisions:

- Have all workloads affected by the remediation been identified and notated in the migration backlog?
- For workloads that are not affected, will a migration produce a similar return on investment (ROI)?
- Can the affected assets be remediated in alignment with the original migration timeline? What impact would timeline changes have on ROI?
- Is it economically feasible to remediate the assets in parallel with migration efforts?
- Is there sufficient bandwidth on staff to remediate and migrate? Should a partner be engaged to execute one or both tasks?

If these questions don't yield favorable answers, a few alternative approaches that move beyond a basic IaaS rehosting strategy may be worth considering:

- **Containerization.** Some assets can be hosted in a containerized environment without remediation. This could produce less-than-favorable performance and doesn't resolve security or compliance issues.
- **Automation.** Depending on the workload and remediation requirements, it may be more profitable to script the deployment to new assets using a DevOps approach.
- **Rebuild.** When remediation costs are very high and business value is equally high, a workload may be a good fit as a candidate for rebuilding or rearchitecting.

Next steps

After remediation is complete, replication activities are ready.

Replicate assets

What role does replication play in the migration process?

3/6/2021 • 4 minutes to read • Edit Online

On-premises datacenters are filled with physical assets like servers, appliances, and network devices. However, each server is only a physical shell. The real value comes from the binary running on the server. The applications and data are the purpose for the datacenter. Those are the primary binaries to migrate. Powering these applications and data stores are other digital assets and binary sources, like operating systems, network routes, files, and security protocols.

Replication is the workhorse of migration efforts. It is the process of copying a point-in-time version of various binaries. The binary snapshots are then copied to a new platform and deployed onto new hardware, in a process referred to as *seeding*. When executed properly, the seeded copy of the binary should behave identically to the original binary on the old hardware. However, that snapshot of the binary is immediately out of date and misaligned with the original source. To keep the new binary and the old binary aligned, a process referred to as *synchronization* continuously updates the copy stored in the new platform. Synchronization continues until the asset is promoted in alignment with the chosen promotion model. At that point, the synchronization is severed.

Required prerequisites to replication

Prior to replication, the new platform and hardware must be prepared to receive the binary copies. The article on prerequisites outlines minimum environment requirements to help create a safe, robust, high-performance platform to receive the binary replicas.

The *source binaries* must also be prepared for replication and synchronization. The articles on assessment, architecture, and remediation each address the actions necessary to ensure that the source binary is ready for replication and synchronization.

A *toolchain* that aligns with the new platform and source binaries must be implemented to execute and manage the replication and synchronization processes. The article on replication options outlines various tools that could contribute to a migration to Azure.

Replication risks - physics of replication

When planning for the replication of any binary source to a new destination, there are a few fundamental laws to seriously consider during planning and execution.

- **Speed of light.** When moving high volumes of data, fiber is still the fastest option. Unfortunately, those cables can only move data at two-thirds the speed of light. This means that there is no method for instantaneous or unlimited replication of data.
- **Speed of WAN pipeline.** More consequential than the speed of data movement is the uplink bandwidth, which defines the volume of data per second that can be carried over a company's existing WAN to the target datacenter.
- **Speed of WAN expansion.** If budgets allow, additional bandwidth can be added to a company's WAN solution. However, it can take weeks or months to procure, provision, and integrate additional fiber connections.
- **Speed of disks.** If data could move faster and there was no limit to the bandwidth between the source binary and the target destination, physics would still be a limiter. Data can be replicated only as quickly as it can be read from source disks. Reading every one or zero from every spinning disk in a datacenter takes

time.

- **Speed of human calculations.** Disks and light move faster than human decision processes. When a group of humans is required to collaborate and make decisions together, the results will come even more slowly. Replication can never overcome delays related to human intelligence.

Each of these laws of physics drive the following risks that commonly affect migration plans:

- **Replication time.** Replication requires time and bandwidth. Plans should include realistic timelines that reflect the amount of time it takes to replicate binaries. *Total available migration bandwidth* is the amount of up-bound bandwidth, measured in megabits per second (Mbps) or gigabits per second (Gbps), that is not consumed by other higher priority business needs. *Total migration storage* is the total disk space, measured in gigabytes or terabytes, required to store a snapshot of all assets to be migrated. An initial estimate of time can be calculated by dividing the *total migration storage* by *total available migration bandwidth*. Note the conversion from bits to bytes. See the next item, "Cumulative effect of disk drift," for a more accurate calculation of time.
- **Cumulative effect of disk drift.** From the point of replication to the promotion of an asset to production, the source and destination binaries must remain synchronized. *Drift* in binaries consumes additional bandwidth, as all changes to the binary must be replicated on a recurring basis. During synchronization, all binary drift must be included in the calculation for total migration storage. The longer it takes to promote an asset to production, the more cumulative drift will occur. The more assets being synchronized, the more bandwidth consumed. With each asset being held in a synchronization state, a bit more of the total available migration bandwidth is lost.
- **Time to business change.** As mentioned previously, synchronization time has a cumulative negative effect on migration speed. Prioritization of the migration backlog and advanced preparation for the business change plan are crucial to the speed of migration. The most significant test of business and technical alignment during a migration effort is the pace of promotion. The faster an asset can be promoted to production, the less impact disk drift will have on bandwidth and the more bandwidth/time that can be allocated to replication of the next workload.

Next steps

After replication is complete, staging activities can begin.

Staging activities during a migration

Replication options

3/6/2021 • 2 minutes to read • Edit Online

Before any migration, you should ensure that primary systems are safe and will continue to run without issues. Any downtime disrupts users or customers, and it costs time and money. Migration is not as simple as turning off the virtual machines on-premises and copying them across to Azure. Migration tools must take into account asynchronous or synchronous replication to ensure that live systems can be copied to Azure with no downtime. Most of all, systems must be kept in lockstep with on-premises counterparts. You might want to test migrated resources in isolated partitions in Azure, to ensure that workloads work as expected.

The content within the Cloud Adoption Framework assumes that Azure Migrate (or Azure Site Recovery) is the most appropriate tool for replicating assets to the cloud. However, there are other options available. This article discusses those options to help enable decision-making.

Azure Site Recovery (also known as Azure Migrate)

Azure Site Recovery orchestrates and manages disaster recovery for Azure VMs, on-premises VMs, and physical servers. You can also use Site Recovery to manage migration of machines on-premises and other cloud providers to Azure. Replicate on-premises machines to Azure or Azure VMs to a secondary region. Then, you fail the VM over from the primary site to the secondary and complete the migration process. With Azure Site Recovery, you can achieve various migration scenarios:

- **Migrate from on-premises to Azure.** Migrate on-premises VMware VMs, Hyper-V VMs, and physical servers to Azure. To do this, complete almost the same steps as you would for full disaster recovery. Simply don't fail machines back from Azure to the on-premises site.
- **Migrate between Azure regions.** Migrate Azure VMs from one Azure region to another. After the migration is complete, configure disaster recovery for the Azure VMs now in the secondary region to which you migrated.
- **Migrate from other cloud to Azure.** You can migrate your compute instances provisioned on other cloud providers to Azure VMs. Site Recovery treats those instances as physical servers for migration purposes.

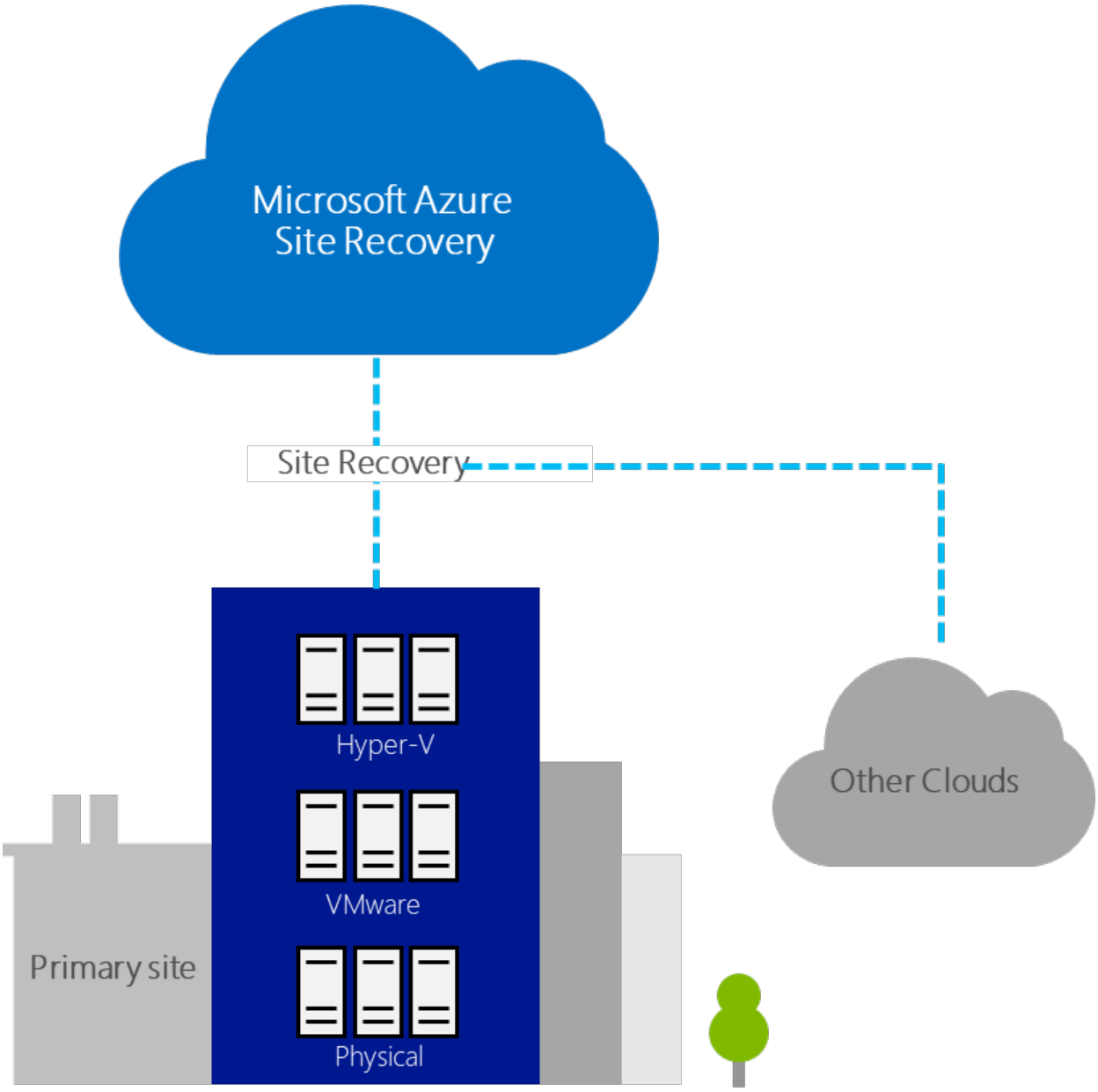


Figure 1: Azure Site Recovery moving assets to Azure or other clouds.

After you have assessed on-premises and cloud infrastructure for migration, Azure Site Recovery contributes to your migration strategy by replicating on-premises machines. With the following easy steps, you can set up migration of on-premises VMs, physical servers, and cloud VM instances to Azure:

- Verify prerequisites.
- Prepare Azure resources.
- Prepare on-premises VM or cloud instances for migration.
- Deploy a configuration server.
- Enable replication for VMs.
- Test failover to make sure everything's working.
- Run a one-time failover to Azure.

Azure Database Migration Service

This service helps reduce the complexity of your cloud migration by using a single comprehensive service instead of multiple tools. Azure Database Migration Service is designed as a seamless, end-to-end solution for moving on-premises SQL Server databases to the cloud. It is a fully managed service designed to enable seamless migrations from multiple database sources to Azure data platforms with minimal downtime. It integrates some of the functionality of existing tools and services, providing customers with a comprehensive,

highly available solution.

The service uses Data Migration Assistant to generate assessment reports that provide recommendations to guide you through the changes required prior to performing a migration. It's up to you to perform any required remediation. When you're ready to begin the migration process, Azure Database Migration Service performs all of the associated steps. You can fire and forget your migration projects with peace of mind, knowing that the process takes advantage of best practices as determined by Microsoft.

Next steps

After replication is complete, staging activities can begin.

Staging activities during a migration

Understand staging activities during a migration

3/6/2021 • 2 minutes to read • Edit Online

As described in the article on promotion models, *staging* is the point at which assets have been migrated to the cloud. However, they're not ready to be promoted to production yet. This is often the last step in the migrate process of a migration. After staging, the workload is managed by an IT operations or cloud operations team to prepare it for production usage.

Deliverables

Staged assets may not be ready for use in production. There are several production readiness checks that should be finalized before this stage is considered complete. The following is a list of deliverables often associated with completion of asset staging.

- **Automated testing.** Any automated tests available to validate workload performance should be run before concluding the staging process. After the asset leaves staging, synchronization with the original source system is terminated. As such, it is harder to redeploy the replicated assets, after the assets are staged for optimization.
- **Migration documentation.** Most migration tools can produce an automated report of the assets being migrated. Before concluding the staging activity, all migrated assets should be documented for clarity.
- **Configuration documentation.** Any changes made to an asset (during remediation, replication, or staging) should be documented for operational readiness.
- **Backlog documentation.** The migration backlog should be updated to reflect the workload and assets staged.

Next steps

After staged assets are tested and documented, you can proceed to optimization activities.

Optimize migrated workloads

Release workloads

3/6/2021 • 2 minutes to read • Edit Online

After a collection of workloads and their supporting assets have been deployed to the cloud, it must be prepared before it can be released. In this phase of the migration effort, the collection of workloads are load tested and validated with the business. They're then optimized and documented. Once the business and IT teams have reviewed and signed off on workload deployments, those workloads can be released or handed off to governance, security, and operations teams for ongoing operations.

The objective of "release workloads" is to prepare migrated workloads for promotion to production usage.

Definition of done

The optimization process is complete when a workload has been properly configured, sized, and deployed to production.

Accountability during optimization

The cloud adoption team is accountable for the entire optimization process. However, members of the cloud strategy team, the cloud operations team, and the cloud governance team should also be responsible for activities within this process.

Responsibilities during optimization

In addition to the high-level accountability, there are actions that an individual or group needs to be directly responsible for. The following are a few activities that require assignments to responsible parties:

- **Business testing.** Resolve any compatibility issues that prevent the workload from completing its migration to the cloud.
  - Power users from within the business should participate heavily in testing of the migrated workload. Depending on the degree of optimization attempted, multiple testing cycles may be required.
- **Business change plan.** Development of a plan for user adoption, changes to business processes, and modification to business KPIs or learning metrics as a result of the migration effort.
- **Benchmark and optimize.** Study of the business testing and automated testing to benchmark performance. Based on usage, the cloud adoption team refines sizing of the deployed assets to balance cost and performance against expected production requirements.
- **Ready for production.** Prepare the workload and environment for the support of the workload's ongoing production usage.
- **Promote.** Redirect production traffic to the migrated and optimized workload. This activity represents the completion of a release cycle.

In addition to core activities, there are a few parallel activities that require specific assignments and execution plans:

- **Decommission.** Generally, cost savings can be realized from a migration, when the previous production assets are decommissioned and properly disposed of.
- **Retrospective.** Every release creates an opportunity for deeper learning and adoption of a growth mindset. When each release cycle is completed, the cloud adoption team should evaluate the processes used during migration to identify improvements.

Next steps

With a general understanding of the optimization process, you're ready to begin the process by establishing a business change plan for the candidate workload.

Business change plan

Business change plan

3/6/2021 • 3 minutes to read • Edit Online

Traditionally, IT has overseen the release of new workloads. During a major transformation, like a datacenter migration or a cloud migration, a similar pattern of IT lead adoption could be applied. However, the traditional approach might miss opportunities to realize additional business value. For this reason, before a migrated workload is promoted to production, implementing a broader approach to user adoption is suggested. This article outlines the ways in which a business change plan adds to a standard user adoption plan.

Traditional user adoption approach

User adoption plans focus on how users will adopt a new technology or change to a given technology. This approach is time tested for introducing users to new tools. In a typical user adoption plan, IT focuses on the installation, configuration, maintenance, and training associated with the technical changes being introduced to the business environment.

Although approaches may vary, general themes are present in most user adoption plans. These themes are typically based on a risk control and facilitation approach that aligns to incremental improvement. The Eason Matrix, illustrated in the following figure, represents the drivers behind those themes across a spectrum of adoption types.

	Revolution → Evolution		
Adoption type / Criteria	Big Bang	Parallel Running	Phased adoption
Needed "critical mass"	Big		Small
Need for risk control	Low		High
Need for facilitation of the change.	Low		High
Pace of change over	High		Low
Local design needs	Low		High
User adaptation	Difficult →		Easy

Diagram: Eason Matrix of user adoption types.

These themes are often based on the assumption that introduction of new solutions to users should focus largely on risk control and facilitation of change. Additionally, IT has focused mostly on risk from the technology change and facilitation of that change.

Create business change plans

A business change plan looks beyond the technical change and assumes that every release in a migration effort drives some level of business process change. It looks upstream and downstream from the technical changes. The following questions help participants think about user adoption from a business change perspective, to maximize business impact:

Upstream questions. Upstream questions look at impacts or changes that come before user adoption happens:

- Has an expected business outcome been quantified?

- Does the business impact map to defined learning metrics?
- Which business processes and teams take advantage of this technical solution?
- Who in the business can best align power users for testing and feedback?
- Have the affected business leaders been involved in the prioritization and migration planning?
- Are there any critical events or dates for the business that could be affected by this change?
- Does the business change plan maximize impact but minimize business disruption?
- Is downtime expected? Has a downtime window been communicated to end users?

Downstream questions. After the adoption is complete, the business change can begin. Unfortunately, this is where many user adoption plans end. Downstream questions help the cloud strategy team maintain a focus on transformation after technical change is completed:

- Are business users responding well to the changes?
- Has performance anticipation been maintained, now that the technical change has been adopted?
- Are business processes or customer experiences changing in the anticipated ways?
- Are additional changes required to realize learning metrics?
- Did the changes align to the targeted business outcomes? If not, why not?
- Are additional changes required to contribute to business outcomes?
- Have any negative effects been observed as a result of this change?

The business change plan varies from company to company. The goal of these questions is to help better integrate the business into the change associated with each release. By looking at each release not as a technology change to be adopted but instead as a business change plan, business outcomes can become more obtainable.

Next steps

After business change is documented and planned, business testing can begin.

Guidance for business testing (UAT) during migration

References

- Eason, K. (1988) *Information Technology and Organizational Change*, New York: Taylor and Francis.

Guidance for business testing (UAT) during migration

3/6/2021 • 3 minutes to read • Edit Online

Traditionally seen as an IT function, user acceptance testing during a business transformation can be orchestrated solely by IT. However, this function is often most effectively executed as a business function. IT then supports this business activity by facilitating the testing, developing testing plans, and automating tests when possible. Although IT can often serve as a surrogate for testing, there is no replacement for firsthand observation of real users attempting to take advantage of a new solution in the context of a real or replicated business process.

> **NOTE**
>
> When available, automated testing is a much more effective and efficient means of testing any system. However, cloud migrations often focus most heavily on legacy systems or at least stable production systems. Often, those systems aren't managed by thorough and well-maintained automated tests. This article assumes that no such tests are available at the time of migration.

Second to automated testing is testing of the process and technology changes by power users. Power users are the people that commonly execute a real-world process that requires interactions with a technology tool or set of tools. They could be represented by an external customer using an e-commerce site to acquire goods or services. Power users could also be represented by a group of employees executing a business process, such as a call center servicing customers and recording their experiences.

The goal of business testing is to solicit validation from power users to certify that the new solution performs in line with expectations and does not impede business processes. If that goal isn't met, the business testing serves as a feedback loop that can help define why and how the workload isn't meeting expectations.

Business activities during business testing

During business testing, the first iteration is manually driven directly with customers. This is the purest but most time-consuming form of feedback loop.

- **Identify power users.** The business generally has a better understanding of the power users who are most affected by a technical change.
- **Align and prepare power users.** Ensure that power users understand the business objectives, desired outcomes, and expected changes to business processes. Prepare them and their management structure for the testing process.
- **Engage in feedback loop interpretation.** Help the IT staff understand the impact of various points of feedback from power users.
- **Clarify process change.** When transformation could trigger a change to business processes, communicate the change and any downstream impacts.
- **Prioritize feedback.** Help the IT team prioritize feedback based on the business impact.

At times, IT may employ analysts or product owners who can serve as proxies for the itemized business testing activities. However, business participation is highly encouraged and is likely to produce favorable business outcomes.

IT activities during business testing

IT serves as one of the recipients of the business testing output. The feedback loops exposed during business testing eventually become work items that define technical change or process change. As a recipient, IT is expected to aid in facilitation, collection of feedback, and management of resultant technical actions. The typical activities IT performs during business testing include:

- Provide structure and logistics for business testing.
- Aid in facilitation during testing.
- Provide a means and process for recording feedback.
- Help the business prioritize and validate feedback.
- Develop plans for acting on technical changes.
- Identify existing automated tests that could streamline the testing by power users.
- For changes that could require repeated deployment or testing, study testing processes, define benchmarks, and create automation to further streamline power user testing.

Next steps

In conjunction with business testing, optimization of migrated assets can refine cost and workload performance.

Benchmark and resize cloud assets

Benchmark and resize cloud assets

3/6/2021 • 3 minutes to read • Edit Online

Monitoring usage and spending is critically important for cloud infrastructures. Organizations pay for the resources they consume over time. When usage exceeds agreement thresholds, unexpected cost overages can quickly accumulate. Cost management reports monitor spending to analyze and track cloud usage, costs, and trends. Using overtime reports, detect anomalies that differ from normal trends. Inefficiencies in cloud deployment are visible in optimization reports. Note inefficiencies in cost-analysis reports.

In the traditional on-premises models of IT, requisition of IT systems is costly and time consuming. The processes often require lengthy capital expenditure review cycles and may even require an annual planning process. As such, it is common practice to buy more than is needed. It is equally common for IT administrators to then overprovision assets in preparation for anticipated future demands.

In the cloud, the accounting and provisioning models eliminate the time delays that lead to overbuying. When an asset needs additional resources, it can be scaled up or out almost instantly. This means that assets can safely be reduced in size to minimize resources and costs consumed. During benchmarking and optimization, the cloud adoption team seeks to find the balance between performance and costs, provisioning assets to be no larger and no smaller than necessary to meet production demands.

Should assets be optimized during or after the migration?

Should an asset be optimized during or after the migration? The simple answer is **both**. However, that's not entirely accurate. To explain, take a look at two basic scenarios for optimizing resource sizing:

- **Planned resizing.** Often, an asset is clearly oversized and underutilized and should be resized during deployment. Determining if an asset has been successfully resized in this case requires user acceptance testing after migration. If a power user does not experience performance or functionality losses during testing, you can conclude the asset has been successfully sized.
- **Optimization.** In cases where the need for optimization is unclear, IT teams should use a data-driven approach to resource size management. Using benchmarks of the asset's performance, an IT team can make educated decisions regarding the most appropriate size, services, scale, and architecture of a solution. They can then resize and test performance theories post-migration.

During the migration, use educated guesses and experiment with sizing. However, true optimization of resources requires data based on actual performance in a cloud environment. For true optimization to occur, the IT team must first implement approaches to monitoring performance and resource utilization.

Benchmark and optimize with Azure Cost Management + Billing

Azure Cost Management + Billing manages cloud spend with transparency and accuracy. This service monitors, benchmarks, allocates, and optimizes cloud costs.

Historical data can help manage costs by analyzing usage and costs over time to identify trends, which are then used to forecast future spending. Cost management also includes useful projected cost reports. Cost allocation manages costs by analyzing costs based on tagging policies. Use cost allocation for showback/chargeback to show resource utilization and associated costs to influence consumption behaviors or charge tenant customers. Access control helps manage costs by ensuring that users and teams access only the cost management data that they need. Alerting helps manage costs through automatic notification when unusual spending or overspending occurs. Alerts can also notify other stakeholders automatically for spending anomalies and overspending risks. Various reports support alerts based on budget and cost thresholds.

Improve efficiency

Determine optimal VM usage, identify idle VMs, or remove idle VMs and unattached disks using Azure Cost Management + Billing. Using information in sizing optimization and inefficiency reports, create a plan to downsize or remove idle VMs.

Next steps

After a workload has been tested and optimized, it is time to ready the workload for promotion.

Getting a migrated workload ready for production promotion

Prepare a migrated application for production promotion

3/6/2021 • 2 minutes to read • Edit Online

After a workload is promoted, production user traffic is routed to the migrated assets. Readiness activities provide an opportunity to prepare the workload for that traffic. The following are a few business and technology considerations to help guide readiness activities.

Validate the business change plan

Transformation happens when business users or customers take advantage of a technical solution to execute processes that drive the business. Readiness is a good opportunity to validate the business change plan and to ensure proper training for the business and technical teams involved. In particular, ensure that the following technology-related aspects of the change plan are properly communicated:

- End-user training is completed (or at least planned).
- Any outage windows have been communicated and approved.
- Production data has been synchronized and validated by end users.
- Validate promotion and adoption timing; ensure timelines and changes have been communicated to end users.

Final technical readiness tests

Ready is the last step prior to production release. That means it is also the last chance to test the workload. The following are a few tests that are suggested during this phase:

- **Network isolation testing.** Test and monitor network traffic to ensure proper isolation and no unexpected network vulnerabilities. Also validate that any network routing to be severed during cutover is not experiencing unexpected traffic.
- **Dependency testing.** Ensure that all workload application dependencies have been migrated and are accessible from the migrated assets.
- **Business continuity and disaster recovery (BCDR) testing.** Validate that any backup and recovery SLAs are established. If possible, perform a full recovery of the assets from the BCDR solution.
- **End-user route testing.** Validate traffic patterns and routing for end-user traffic. Ensure that network performance aligns with expectations.
- **Final performance check.** Ensure that performance testing has been completed and approved by end users. Execute any automated performance testing.

Final business validation

After the business change plan and technical readiness have been validated, the following final steps can complete the business validation:

- **Cost validation (plan versus actual).** Testing is likely to produce changes in sizing and architecture. Ensure that actual deployment pricing still aligns with the original plan.
- **Communicate and execute cutover plan.** Prior to cutover, communicate the cutover and execute accordingly.

Next steps

After all readiness activities have been completed, its time to promote the workload.

What is required to promote a migrated resource to production?

What is required to promote a migrated resource to production?

5/18/2021 • 2 minutes to read • Edit Online

Promotion to production marks the completion of a workload's migration to the cloud. After the asset and all of its dependencies are promoted, production traffic is rerouted. The rerouting of traffic makes the on-premises assets obsolete, allowing them to be decommissioned.

The process of promotion varies according to the workload's architecture. However, there are several consistent prerequisites and a few common tasks. This article describes each and serves as a kind of pre-promotion checklist.

Prerequisite processes

Each of the following processes should be executed, documented, and validated prior to production deployment:

- Assess: The workload has been assessed for cloud compatibility.
- Architect: The structure of the workload has been properly designed to align with the chosen cloud provider.
- Replicate: The assets have been replicated to the cloud environment.
- Stage: The replicated assets have been restored in a staged instance of the cloud environment.
- Business testing: The workload has been fully tested and validated by business users.
- Business change plan: The business has shared a plan for the changes to be made in accordance with the production promotion; this should include a user adoption plan, changes to business processes, users that require training, and timelines for various activities.
- Ready: Generally, a series of technical changes must be made before promotion.

Best practices to execute prior to promotion

The following technical changes will likely need to be completed and documented as part of the promotion process:

- **Domain alignment.** Some corporate policies require separate domains for staging and production. Ensure that all assets are joined to the proper domain.
- **User routing.** Validate that users are accessing the workload through proper network routes; verify consistent performance expectations.
- **Identity alignment.** Validate that the users being rerouted to the application have proper permissions within the domain to host the application.
- **Performance.** Perform a final validation of workload performance to minimize surprises.
- **Validation of business continuity and disaster recovery.** Validate that proper backup and recovery processes are functioning as expected.
- **Data classification.** Validate data classification to ensure that proper protections and policies have been implemented.
- **Chief information security officer (CISO) verification.** Validate that the information security officer has reviewed the workload, business risks, risk tolerance, and mitigation strategies.

Final step: Promote

Workloads will require varying levels of detailed review and promotion processes. However, network

realignment serves as the common final step for all promotion releases. When everything else is ready, update DNS records or IP addresses to route traffic to the migrated workload.

Next steps

Promotion of a workload signals the completion of a release. However, in parallel with migration, retired assets need to be decommissioned taking them out of service.

Decommission retired assets

Decommission retired assets

3/6/2021 • 2 minutes to read • Edit Online

After a workload is promoted to production, the assets that previously hosted the production workload are no longer required to support business operations. At that point, the older assets are considered retired. Retired assets can then be decommissioned, reducing operational costs. Decommissioning a resource can be as simple as turning off the power to the asset and disposing of the asset responsibly. Unfortunately, decommissioning resources can sometimes have undesired consequences. The following guidance can aid in properly decommissioning retired resources, with minimal business interruptions.

Cost savings realization

When cost savings are the primary motivation for a migration, decommissioning is an important step. Until an asset is decommissioned, it continues to consume power, environmental support, and other resources that drive costs. After the asset is decommissioned, the costs savings can start to be realized.

Continued monitoring

After a migrated workload is promoted, the assets to be retired should continue to be monitored to validate that no additional production traffic is being routed to the wrong assets.

Testing windows and dependency validation

Even with the best planning, production workloads may still contain dependencies on assets that are presumed retired. In such cases, turning off a retired asset could cause unexpected system failures. As such, the termination of any assets should be treated with the same level of rigor as a system maintenance activity. Proper testing and outage windows should be established to facilitate the termination of the resource.

Holding period and data validation

It's not uncommon for migrations to miss data during replication processes. This is especially true for older data that isn't used on a regular basis. After a retired asset has been turned off, it is still wise to maintain the asset for a while to serve as a temporary backup of the data. Companies should allow at least 30 days for holding and testing before destroying retired assets.

Next steps

After retired assets are decommissioned, the migration is completed. This creates a good opportunity to improve the migration process, and a retrospective engages the cloud adoption team in a review of the release in an effort to learn and improve.

Retrospective

How do retrospectives help build a growth mindset?

3/6/2021 • 2 minutes to read • Edit Online

"Culture eats strategy for breakfast." The best migration plan can easily be undone, if it doesn't have executive support and encouragement from leadership. Learning, growing, and even failure are at the heart of a growth mindset. They're also at the heart of any transformation effort.

Humility and curiosity are never more important than during a business transformation. Embracing digital transformation requires both in ample supply. These traits are strengthened by regular introspection and an environment of encouragement. When employees are encouraged to take risks, they find better solutions. When employees are allowed to fail and learn, they succeed. Retrospectives are an opportunity for such investigation and growth.

Retrospectives reinforce the principles of a growth mindset: experimentation, testing, learning, sharing, growing, and empowering. They provide a safe place for team members to share the challenges faced in the current sprint. And they allow the team to discuss and collaborate on ways to overcome those challenges. Retrospectives empower the team to create sustainable growth.

Retrospective structure

A quick search on any search engine will offer many different approaches and tools for running a retrospective. Depending on the maturity of the culture and experience level of the team, these could prove useful. However, the general structure of a retrospective remains roughly the same. During these meetings, each member of the team is expected to contribute a thought regarding three basic questions:

- What went well?
- What could have been better?
- What did we learn?

Although these questions are simple in nature, they require employees to pause and reflect on their work over the last iteration. This small pause for introspection is the primary building block of a growth mindset. The humility and honesty produced when sharing the answers can become infectious beyond the time contract for the retrospective meeting.

Leadership's role in a retrospective

The topic of leadership involvement in a retrospective is highly debated. Many technical teams suggest that leaders of any level should not be involved in the process, since it could discourage transparency and open dialogue. Others suggest that retrospectives are a good place for leaders to stay connected and to find ways to provide additional support. This decision is best left to the team and its leadership structure.

If leaders are involved in the retrospective, one role is highly encouraged. The leader's primary duty in a retrospective is to make the team feel safe. Creating a growth mindset within a culture requires employees to be free to share their failures and successes without fear of rebuke. Leaders who applaud the courage and humility required to admit shortcomings are more likely to see a growth mindset established in their teams. When leaders take action on data points shared in a retrospective, they're likely to see this tool become an ineffective formality.

Lessons learned

Highly effective teams don't just run retrospective meetings. They live retrospective processes. The lessons

learned and shared in these meetings can influence process, shape future work, and help the team execute more effectively. Lessons learned in a retrospective should help the team grow organically. The primary byproducts of a retrospective are an increase in experimentation and a refinement of the lessons learned by the team.

That new growth is most tangibly represented in changes to the release or iteration backlog.

The retrospective marks the end of a release or iteration, as teams gain experience and learn lessons, and they adjust the adjust the release and iteration backlog to reflect new processes and experiments to be tested. This starts the next iteration through the migration processes.

Skills readiness for cloud migration

3/6/2021 • 2 minutes to read • Edit Online

During a cloud migration, it is likely that employees, as well as some incumbent systems integration partners or managed services partners, will need to develop new skills to be effective during migration efforts.

There are four distinct processes that are completed iteratively in the Migrate methodology. The following sections align the necessary skills for each of those processes with references to two prerequisites for skilling resources.

Prerequisites skilling resources

Implementation of the Migrate methodology builds on the skills acquired during planning and preparing a migration journey.

Assess skilling resources

The following tools can aid the team in execution of assess activities:

- Balance the portfolio: Ensure balance and proper investment allocations across an application portfolio.
- Build a business justification: Create and understand the business justification driving the cloud migration effort.
- Rationalize the digital estate: Rationalize assets in the digital estate.
- Application portfolio assessment: Criteria for making decisions regarding migration or innovation options within the application portfolio.
- Assessing and Planning Microsoft Azure Migration: A Pluralsight course to aid in assessing on-premises workloads.

During assess processes, architects will design solutions for each workload. The following skilling resources help prepare architects for these tasks:

- Foundations for Cloud Architecture: A Pluralsight course to help architect the right foundational solutions.
- Microsoft Azure Architecture: Getting Started: A Pluralsight course to give architects a foundational knowledge in Azure architecture.
- Designing migrations for Microsoft Azure: A Pluralsight course to help architects design a migration solution.

Migrate skilling resources

The following tutorial can prepare the team for migration activities:

- Migrate to Azure: Using Azure Migrate to migrate VMs to Azure.
- Rehosting Workloads to Azure: A Pluralsight course that teaches viewers how to rehost workloads to Azure.
- Migrating physical and virtual servers to Azure: A Pluralsight course for migrating servers to Azure.
- Import and export data to Azure: A Pluralsight course on the movement of data to and from Azure.

Optimize and promote process changes

The following tools can help the team optimize resources and promote to production:

- Cost and sizing: Adjust sizing to align costs and budgets.
- Promote a workload: Change network configuration to reroute production users to migrated workloads.

Secure and manage process changes

The following tools can help the team find ways to secure and manage migrated assets:

- Secure and manage workloads in Azure: Best practices for securing and managing workloads in Azure.

Next steps

Return to the migration best practices checklist to ensure your migration method is fully aligned.

Migration best practices checklist

Cloud adoption-related innovation

5/20/2021 • 5 minutes to read • Edit Online

All IT portfolios contain a few workloads and ideas that could significantly improve a company's position in the market. Most cloud adoption efforts focus on the migration and modernization of existing workloads. Cloud innovation, however, can provide the greatest business value. Innovation related to cloud adoption can unlock new technical skills and expanded business capabilities.

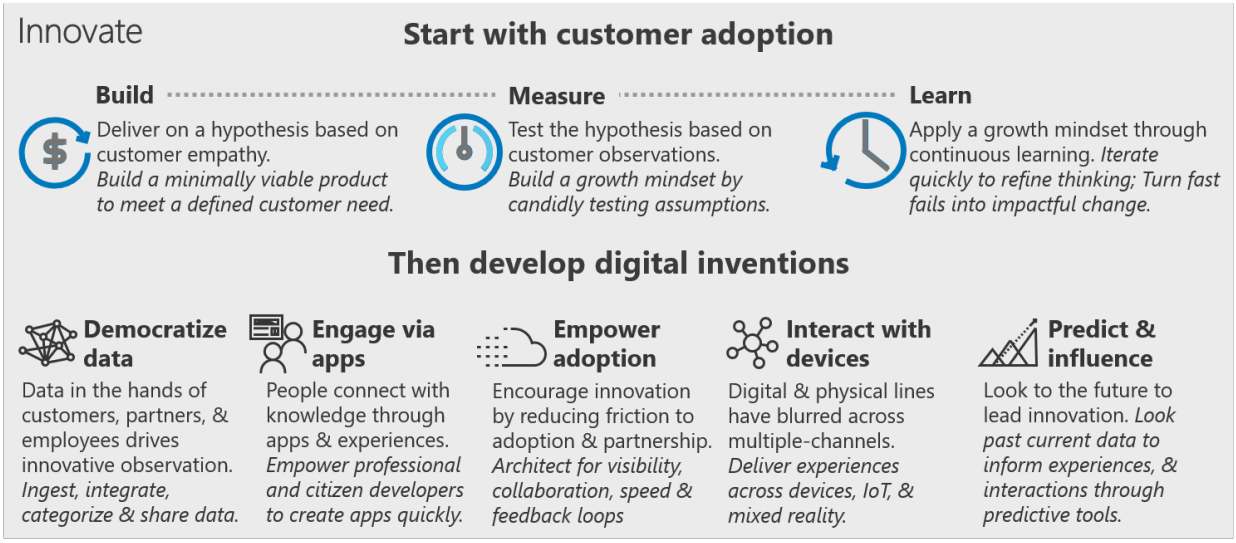
In the Innovate methodology of the Cloud Adoption Framework, we'll focus on understanding customer needs and rapidly building innovations that shape how your customers interact with your products. We'll also illustrate an approach to delivering on the value of a minimum viable product (MVP).

To prepare you for this phase of the cloud adoption lifecycle, the framework suggests the following cloud innovation exercises:

	Create hypothesis with business value consensus: Before you decide on technical solutions, identify how new innovations can drive business value and come up with a hypothesis about customer needs.
	Build your first MVP: Once a hypothesis has enough value potential to be built into your application, the build process starts. Development sprints should be as quick as possible, to allow quick verification or rejection of the hypothesis, or to fine tune the way in which the required functionality is to be integrated in the application.
	Measure & Learn from your MVP: You want to verify the accuracy of your hypothesis as soon as possible. A Minimum Viable Product (MVP) is a preliminary version of the new functionality, that offers just enough functionality to gather feedback that confirms if you are moving in the right direction.
	Expand digital innovation: To refine your hypothesis using the innovation disciplines or the digital inventions which include: democratize data, engage via apps, empower adoption, Interact with devices, and predict and influence.

Innovation summary

The following approach builds on existing lean methodologies. It's designed to help you create a cloud-focused conversation about customer adoption and a scientific model for creating business value. The approach also maps existing Azure services to manageable decision processes. This alignment can help you find the right technical options to address specific customer needs or hypotheses. As you build your MVP, you should consider using the five disciplines of digital inventions, which include democratize data, engage via applications, empower adoption, interact with devices, and predict and influence.



This article series emphasizes the following aspects of this methodology:

- First, always start with customer adoption to generate feedback that builds customer partnerships through the build-measure-learn feedback loop.
- Second, examine approaches to developing digital inventions that prioritize adoption.

Not all disciplines need to be adopted at once. These disciplines define a series of approaches to developing digital inventions while keeping adoption and customer empathy in mind.

The following section describes the formula for innovation and the commitments required for success with this approach.

Formula for innovation

Successful innovation is not a big-bang transformational event or an elusive magical unicorn. Success in innovation is more of a balancing act, illustrated by a simple equation: **`innovation = invention + adoption`**.

Innovation happens at the intersection of invention and adoption. True innovation stems from slowly adjusting human experiences through new approaches, new processes, and new technologies. In this formula, invention means creating a new solution that meets a customer need. Conversely, adoption means applying the new solution to shape human behaviors and interactions. Finding the right balance between invention and adoption requires iteration, data-driven decision making, constant learning, and a growth mindset. It also requires technologies that can keep pace with the countless opportunities to learn in today's digital society.

The cloud is often a great platform for invention or the technological aspects of innovation. Unfortunately, most great ideas fail during the hard work of adoption, rather than during the ideation or invention processes. To ensure success, development teams should always start with adoption as the test for innovation. That's why this methodology starts with adoption. To use this methodology, the following three commitments should be agreed upon by the team:

- Commitment to prioritize customers over technology
- Commitment to transparency
- Commitment to iteration

Cultural commitments

Adopting the Innovate methodology requires some cultural commitments to effectively use the metrics outlined in this article. Before you change your approach to driving innovation, make sure the adoption and leadership teams are ready to make these important commitments.

Commitment to prioritize customers over technology

Every development team has a set of tools or technologies that they're most familiar with. It's wise to play to those strengths and use what you know. However, for innovation to be successful, teams must maintain a focus on customer needs and the hypothesis being tested. At times, this focus may not align with the capabilities of a particular tool or architectural approach. To be successful in innovation, the development team must remain open-minded. During the invention process, focus technical decisions on the needs of the customer over the preferences of your team.

Commitment to transparency

To understand measurement in an innovation approach, you must first understand the commitment to transparency. Innovation can only thrive in an environment that adheres to a *growth mindset*. At the root of a growth mindset is a cultural imperative to learn from experiences. Successful innovation and continuous learning start with a commitment to transparency in measurement. This is a brave commitment for the cloud adoption team. However, that commitment is meaningless if it's not matched by a commitment to preserve transparency within the leadership and cloud strategy teams.

Transparency is important because measuring customer impact doesn't address the question of right or wrong. Nor are impact measurements indicative of the quality of work or the performance of the adoption team. Instead, they represent an opportunity to learn and better meet your customers' needs. Misuse of innovation metrics can stifle that culture. Eventually, such misuse will lead to manipulation of metrics, which in turn causes long-term failure of the invention, the supporting staff, and ultimately the management structure who misused the data. Leaders and contributors alike should avoid using measurements for anything other than an opportunity to learn and improve the MVP solution.

Commitment to iteration

Only one promise rings true across all innovation cycles: you won't get it right on the first try. Measurement helps you understand what adjustments you should make to achieve the desired results. Changes that lead to favorable outcomes stem from iterations of the build-measure-learn process. The cloud adoption team and the cloud strategy team must commit to an iterative mindset before adopting a growth mindset or a build-measure-learn approach.

Next steps

Before building the next great invention, review the different approaches to developing digital inventions while keeping adoption and customer empathy in mind.

Develop digital inventions

Azure innovative solutions guide overview

3/6/2021 • 2 minutes to read • Edit Online

> **NOTE**
> This guide of innovative solutions provides a starting point for innovation guidance in the Cloud Adoption Framework. It's also available in the Azure Quickstart Center.

Before you start developing innovative solutions by using Azure services, you need to prepare your environment, which includes preparing to manage customer feedback loops. In this guide, we introduce features that help you engage customers, build innovative solutions, and drive adoption. For more information, best practices, and considerations related to preparing your cloud environment, see the Cloud Adoption Framework innovate section.

In this innovation guide, you'll learn how to:

- **Manage customer feedback:** Set up tools and processes to manage the build-measure-learn feedback loop by using GitHub and Azure DevOps.
- **Democratize data:** Data alone might be enough to drive innovative solutions to your customers. Deploy common data options in Azure.
- **Engage via applications:** Some innovation requires an engaging experience. Use cloud-native application platforms to create engaging experiences.
- **Empower adoption:** Invention is great, but a plan to reduce friction is needed to empower and scale adoption. Deploy a foundation for CI/CD, DevOps, and other adoption enablers.
- **Interact through devices:** Create ambient experiences to bring your applications and data closer to the customers' point of need. IoT, mixed reality, and mobile experiences are easier with Azure.
- **Predict and influence:** Find patterns in data. Put those patterns to work to predict and influence customer behaviors by using Azure-based predictive analytics tools.

> **TIP**
> For an interactive experience, view this guide in the Azure portal. Go to the Azure Quickstart Center in the Azure portal, select **Azure innovation guide**, and then follow the step-by-step instructions.

Next steps:

- Prepare for innovation with a shared repository and ideation management tools

This guide provides interactive steps that let you try features as they're introduced. To come back to where you left off, use the breadcrumb for navigation.

Prepare for customer feedback

5/20/2021 • 4 minutes to read • Edit Online

User adoption, engagement, and retention are key to successful innovation. Why?

Building an innovative new solution isn't about giving users what they want or think they want. It's about the formulation of a hypothesis that can be tested and improved upon. That testing comes in two forms:

- **Quantitative (testing feedback):** This feedback measures the actions we hope to see.
- **Qualitative (customer feedback):** This feedback tells us what those metrics mean in the customer's voice.

Quantitative data is number-based using a quantifiable measurement process. Quantitative feedback gives numeric insights into data, which is useful for gathering a large number of answers from customers quickly. Examples of quantitative feedback would be multiple-choice questions and numerical user engagement data. Qualitative feedback is more in depth to get a wider variety of answers and insights into customer thoughts or opinions. Examples of qualitative feedback would be a customer survey with open-ended questions. Both methods of customer feedback provide valuable insights to improve your company's products and services.

Before you integrate feedback loops, you need to have a shared repository for your solution. A centralized repo will provide a way to record and act on all the feedback coming in about your project. GitHub is the home for open-source software. It's also one of the most commonly used platforms for hosting source code repositories for commercially developed applications. The article on building GitHub repositories can help you get started with your repo.

Each of the following tools in Azure integrates with (or is compatible with) projects hosted in GitHub:

- Quantitative feedback for web apps
- Quantitative feedback for APIs
- Qualitative feedback
- Close the loop with pipelines

Application Insights is a monitoring tool that provides near-real-time quantitative feedback on the usage of your application. This feedback can help you test and validate your current hypothesis to shape the next feature or user story in your backlog.

Action

To view quantitative data on your applications:

1. Go to **Application Insights**.
   - If your application doesn't appear in the list, select **add** and follow the prompts to start configuring Application Insights.
   - If the desired application is in the list, select it.
2. The **overview** pane includes some statistics on the application. Select **application dashboard** to build a custom dashboard for data that's more relevant to your hypothesis.

GO TO APPLICATION INSIGHTS

To view the data about your applications, go to the Azure portal.

Learn more

- Set up Azure Monitor

- Get started with Azure Monitor Application Insights
- Build a telemetry dashboard

Data democratization tools

5/20/2021 • 2 minutes to read • Edit Online

Data democratization is the ability to make digital information accessible to the average non-technical user of information systems, without having a gatekeeper or outside help to access the data. Democratizing data helps users gain unfettered access to important data without creating a bottleneck that impedes productivity.

One of the first steps in democratizing data is to enhance data discoverability. Cataloging and managing data sharing can help enterprises get the most value from their existing information assets. By democratizing a data catalog it makes data sources easy to discover and understand by the users who manage the data. Azure Data Catalog enables management inside an enterprise, whereas Azure Data Share enables management and sharing outside the enterprise.

Azure services that provide data processing, like Azure Time Series Insights and Stream Analytics, are other capabilities that customers and partners are successfully using for their innovation needs.

- Catalog
- Share
- Insights

Azure Data Catalog

Azure Data Catalog addresses the discovery challenges of data consumers and enables data producers who maintain information assets. It bridges the gap between IT and the business, allowing everyone to contribute their insights. You can store your data where you want it and connect with the tools you want to use. With Azure Data Catalog, you can control who can discover registered data assets. You can integrate into existing tools and processes by using open REST APIs.

- Register
- Search and annotate
- Connect and manage

Go to the Azure Data Catalog documentation

Action

You can use only one Azure Data Catalog per organization. If a catalog has already been created for your organization, you can't add more catalogs.

To create a catalog for your organization:

1. Go to **Azure Data Catalog**.
2. Select **Create**.

GO TO AZURE DATA CATALOG

Engage customers through applications

3/6/2021 • 9 minutes to read • Edit Online

Build cloud-native applications to connect customers in new ways. Cloud-native applications are built from the ground up, optimized for cloud scale and performance. Cloud-native applications are based on microservices architecture, use managed services, and take advantage of continuous delivery to achieve reliability and faster time to market.

Innovation with applications includes both modernizing your existing applications that are hosted on-premises and building cloud-native applications by using containers or serverless technologies. Azure provides PaaS services like Azure App Service to help you easily modernize your existing web and API apps written in .NET, .NET Core, Java, Node.js, Ruby, Python, or PHP for deployment in Azure.

With an open-standard container model, building microservices or containerizing your existing applications and deploying them on Azure is simple when you use managed services like Azure Kubernetes Service, Azure Container Instances, and Web App for Containers. Serverless technologies like Azure Functions and Azure Logic Apps use a consumption model (pay for what you use) and help you focus on building your application rather than deploying and managing infrastructure.

- Deliver value faster
- Create cloud-native applications
- Isolate points of failure

One of the advantages of cloud-based solutions is the ability to gather feedback faster and start delivering value to your user. Whether that user is an external customer or a user in your own company, the faster you can get feedback on your applications, the better.

Azure App Service

Azure App Service provides a hosting environment for your applications that removes the burden of infrastructure management and OS patching. It provides automation of scale to meet the demands of your users while bound by limits that you define to keep costs in check.

Azure App Service provides first-class support for languages like ASP.NET, ASP.NET Core, Java, Ruby, Node.js, PHP, and Python. If you need to host another runtime stack, Web App for Containers lets you quickly and easily host a Docker container within App Service, so you can host your custom code stack in an environment that gets you out of the server business.

Action

To configure or monitor Azure App Service deployments:

1. Go to **App Services**.
2. Configure a new service: select **Add** and follow the prompts.
3. Manage existing services: select the desired application from the list of hosted applications.

GO TO APP SERVICES

Azure Cognitive Services

With Azure Cognitive Services, you can infuse advanced intelligence directly into your application through a set of APIs that let you take advantage of Microsoft-supported AI and machine learning algorithms.

Action

To configure or monitor Azure Cognitive Services deployments:

1. Go to **Cognitive Services**.
2. Configure a new service: select **Add** and follow the prompts.
3. Manage existing services: select the desired service from the list of hosted services.

GO TO COGNITIVE SERVICES

Azure Bot Service

Azure Bot Service extends your standard application by adding a natural bot interface that uses AI and machine learning to create a new way to interact with your customers.

Action

To configure or monitor Azure Bot Service deployments:

1. Go to **Bot Services**.
2. Configure a new service: select **Add** and follow the prompts.
3. Manage existing services: select the desired bot from the list of hosted services.

GO TO BOT SERVICES

Azure DevOps

During your innovation journey, you'll eventually find yourself on the path to DevOps. Microsoft has long had an on-premises product known as Team Foundation Server (TFS). During our own innovation journey, Microsoft developed Azure DevOps, a cloud-based service that provides build and release tools supporting many languages and destinations for your releases. For more information, see Azure DevOps.

Visual Studio App Center

As mobile apps continue to grow in popularity, the need for a platform that can provide automated testing on real devices of various configurations grows. Visual Studio App Center not only provides a place where you can test your cloud-native applications across iOS, Android, Windows, and macOS, it also provides a monitoring platform that can use Azure Application Insights to analyze your telemetry quickly and easily. For more information, see Visual Studio App Center.

Visual Studio App Center also provides a notification service that lets you use a single call to send notifications to your application across platforms without having to contact each notification service individually. For more information, see Visual Studio App Center Push (ACP).

Learn more

- App Service overview
- Web App for Containers: run a custom container
- Introduction to Azure Functions
- Azure for .NET and .NET Core developers
- Azure SDK for Python documentation
- Azure for Java cloud developers
- Create a PHP web app in Azure
- Azure SDK for JavaScript documentation
- Azure SDK for Go documentation
- DevOps solutions

Empower cloud adoption

4/22/2021 • 7 minutes to read • Edit Online

You know that innovation and digital transformation is critical to business success. Digital transformation is the adoption of cloud-based and digital technologies to replace old systems and create better customer experiences, better efficiency, business insights, and greater innovation from the data. You don't accomplish innovation solely through the introduction of new technologies. You need to focus on supporting the people who catalyze change and create the new value that you seek. Developers are at the center of digital transformation, and to empower them to achieve more, you need to accelerate developer velocity. To unleash the creative energy of developer teams, you need to help them build productively, foster global and secure collaboration, and remove barriers so they can scale innovation.

Generate value

- In every industry, every organization is trying to do one thing: drive constant value generation.
- The focus on innovation is essentially a process to help your organization find new ways to generate value.
- Perhaps the biggest mistake organizations make is trying to create new value by introducing new technologies.
- Sometimes the attitude is "if we just use more technology, we'll see things improve." But innovation is first and foremost a people story.
- Innovation is about the combination of people and technology.

Organizations that successfully innovate toward digital transformation see vision, strategy, culture, unique potential, and capabilities as the foundational elements. They then turn to technology with a specific purpose in mind. Every company is becoming a software company. The hiring of software engineers is growing at a faster rate outside the tech industry than inside, according to LinkedIn data.

Innovation is accomplished when organizations support their people to create the value they seek. One group of those people, developers, is a catalyst for innovation. They play an increasingly vital role in value creation and growth across every industry. They're the builders of our era, writing the world's code and sitting at the heart of innovation. Innovative organizations build a culture that empowers developers to achieve more.

- Developer productivity
- Innovate collaboratively
- Innovation characteristics
- LiveOps innovation

Developer velocity

Empowering developers to invent means accelerating developer velocity, enabling them to create more, innovate more, and solve more problems. Developer velocity is the underpinning of each organization's tech intensity. Developer velocity isn't just about speed. It's also about unleashing developer ingenuity, turning your developers' ideas into software with speed and agility so that innovative solutions can be built. The differentiated Azure solution is uniquely positioned to unleash innovation and cloud adoption in your organization.

Build productively

There are several areas of opportunity where Azure can help you build productively:

- Ensure developers become and stay proficient in their domain by helping them advance their knowledge.

- Hone the right skills by giving them the right tools.

One of the best ways to improve your developers' skills is by giving them tools they know and love. Azure tools meet developers where they are today and introduce them to new digital transformation technologies in the context of the code they're writing. With the Azure commitment to open-source software and support for all languages and frameworks in Azure tools, your developers can build how they want and deploy where you want.

Azure DevOps provides best-in-class tools for every developer. Azure developer and digital transformation services infuse modern development practices and emerging trends into our tools. With the Azure platform, developers have access to the latest technologies and a cutting-edge toolchain that supports the way they work.

- AI-assisted development tools
- Integrated tools and cloud
- Remote development and pair programming

Go to Azure DevOps documentation

Action

To create a DevOps project:

1. Go to **Azure DevOps Projects**.
2. Select **Create DevOps project**.
3. Select **Runtime, Framework, and Service**.

GO TO AZURE DEVOPS PROJECTS

Interact through connected devices

3/23/2021 • 5 minutes to read • Edit Online

Innovate through intermittently connected devices and perceptive edge devices. Orchestrate millions of such devices, acquire and process limitless data, and take advantage of a growing number of multisensory, multidevice experiences. For devices at the edge of your network, Azure provides a framework for building immersive and effective business solutions. With ubiquitous computing, enabled by Azure combined with AI technology, you can build every type of intelligent application and system you can envision.

Ubiquitous computing is the processing of information that connects devices and processors to have constant availability, so that computing and processing is made to appear anytime and everywhere needed, using any connected device or perceptive edge device. Examples of ubiquitous computing include any system that sends information to another system to complete a task seamlessly, like a fitness watch that alerts that there is an incoming call from a cell phone and allows completion of the call through the watch, or systems that learn and adjust such as a thermostat or smart speakers.

Azure customers employ a continually expanding set of connected systems and devices that gather and analyze data (close to their users, the data, or both), with complete device management. Users get real-time insights and experiences, delivered by highly responsive and contextually aware applications. By moving parts of the workload to the edge, these connected devices can spend less time sending messages to the cloud and react more quickly to spatial events.

- Industrial assets
- Microsoft HoloLens 2
- Azure Sphere
- Azure Kinect DK
- Drones
- Azure SQL Edge
- IoT plug and play

- Global scale IoT service
- Azure Digital Twins
- Location intelligence
- Spatial experiences
- Azure Remote Rendering

Architect solutions that exercise bidirectional communication with IoT devices at billions scale. Use out-of-box, device-to-cloud telemetry data to understand the state of your devices and define message routes to other Azure services just through configuration. By taking advantage of cloud-to-device messages, you can reliably send commands and notifications to your connected devices and track message delivery with acknowledgment receipts. And you'll automatically resend device messages as needed to accommodate intermittent connectivity.

Here are a few features you'll find:

- **Security-enhanced communication** channel for sending and receiving data from IoT devices.
- **Built-in device management** and provisioning to connect and manage IoT and edge devices at scale.
- **Full integration with Event Grid** and serverless compute, simplifying IoT application development.
- **Compatibility with Azure IoT Edge** for building hybrid IoT applications.

Learn more

- Azure IoT Hub
- Azure IoT Hub Device Provisioning Service (DPS)
- use our modern IoT Azure DevOps project to help with your work item management

Action

To create an IoT hub:

1. Go to **IoT Hub**.
2. Select **Create IoT hub**.

GO TO IOT HUB

The Azure IoT Hub Device Provisioning Service is a helper service for Azure IoT Hub that enables zero-touch, just-in-time provisioning.

Action

To create an Azure IoT Hub Device Provisioning Service:

1. Go to **Device Provisioning Services**.
2. Select **Add**.

GO TO DEVICE PROVISIONING SERVICES

Innovate with AI in Azure

3/6/2021 • 4 minutes to read • Edit Online

As an innovator, your company has rich information about its business and its customers. Using AI innovation, your company can:

- Make predictions about customer needs.
- Automate business processes.
- Discover information that's latent in unstructured data.
- Engage with customers in new ways to deliver better experiences.

This article introduces a few approaches to innovating with AI. Innovations can expand your company's business insights into your existing data. The following table can help you find the best solution for your implementation needs.

SOLUTION CATEGORY	DESCRIPTION	REQUIRED SKILLS
Machine learning	**Azure Machine Learning** Build, deploy, and manage your own machine learning models.	Data scientist and developer
AI applications and agents	**Azure Cognitive Services** Use domain-specific AI models for vision, speech, language, and decision that can be customized with your data. **Azure Bot Service** Improve customer engagement by adding bots to your applications and websites.	Developer
Knowledge mining	**Azure Cognitive Search** Uncover insights that are latent in your content, including documents, contracts, images, and other data types.	Developer

Machine learning

Azure provides advanced machine learning capabilities. Build, train, and deploy your machine learning models across the cloud and edge by using Azure Machine Learning. Develop models faster by using automated machine learning. Use tools and frameworks of your choice without being locked in.

For more information, see Azure Machine Learning overview and getting started with your first machine learning experiment. For more information on the open-source model format and runtime for machine learning, see ONNX Runtime.

Action

A data scientist can use Azure Machine Learning to train and build a model by using advanced languages such as Python and R, as well as by using a drag-and-drop visual experience. To get started with Azure Machine Learning:

1. In the Azure portal, search for and select **Machine Learning**.

2. Select **Add**, and follow the steps in the portal to create a workspace.
3. The new workspace provides both low-code and code-driven approaches for data scientists to train, build, deploy, and manage models.

GO TO AZURE MACHINE LEARNING RESOURCES

Go directly to Azure Machine Learning resources in the Azure portal.

AI applications and agents

Azure provides a set of pre-built AI services called Cognitive Services to build AI applications. Additionally, Azure offers bot service, which allows developers to build conversational AI agents that improve customer and employee engagement.

AI applications

Cognitive Services enables you to incorporate the AI capabilities of vision, speech, language, and decision into your applications. Most predictive models don't require additional training. These services are useful when you don't have data scientists on staff to train the predictive model. Other services require minimal training.

For more information about the training that might be required and a list of available services across vision, speech, language, and decision-making, see the Cognitive Services documentation.

Action

To get started with a Cognitive Services API:

1. In the Azure portal, search for and select **Cognitive Services**.
2. Select **Add** to find a Cognitive Services API in Azure Marketplace.
3. Search for and select a service:
   - If you know the name of the service you want to use, enter the name in **Search the Marketplace**. Then select the service.
   - For a list of Cognitive Services APIs, select **See More** next to the **Cognitive Services** heading. Then select the service.
4. Select **Create**, and follow the steps in the portal to provision the service.

GO TO COGNITIVE SERVICES

Go directly to Cognitive Services in the Azure portal.

AI agents

Interact more naturally with your customers and improve customer engagement through conversational experiences powered by Bot Framework and Azure Bot Service. In addition, use Cognitive Services APIs like Language Understanding (LUIS), QnA Maker, and Speech service. These help your customers with common tasks, leaving your call center agents time to focus on more nuanced, higher value cases.

For more information on how to build bots, see Azure Bot Service.

Action

To get started with Azure Bot Service:

1. In the Azure portal, search for and select **Bot Services**.
2. Select **Add**, and then select **web app bot** or **bot channels registration**.

3. Select **Create**. Then follow the steps in the portal to provision the service.

GO TO AZURE BOT SERVICE

Go directly to Azure Bot Service in the Azure portal.

Knowledge mining

Knowledge mining uses AI to drive content understanding over vast amounts of unstructured, semi-structured, and structured information. Use Azure Cognitive Search to uncover latent insights from your content, including documents, images, and media. You can discover patterns and relationships in your content, understand sentiment, and extract key phrases.

Azure Cognitive Search uses the same natural language stack that Bing and Microsoft Office use. Spend more time innovating and less time maintaining a complex cloud search solution.

For more information, see What is Azure Cognitive Search?

Action

To get started:

1. In the Azure portal, search for and select **Azure Cognitive Search**.
2. Follow the steps in the portal to provision the service.

GO TO AZURE COGNITIVE SEARCH

Go directly to Azure Cognitive Search in the Azure portal.

Kubernetes in the Cloud Adoption Framework

8/18/2021 • 2 minutes to read • Edit Online

Review a prescriptive framework that includes the tools, programs, and content (best practices, configuration templates, and architecture guidance) to simplify adoption of Kubernetes and cloud-native practices at scale.

The list of required actions is categorized by persona to drive a successful deployment of applications on Kubernetes, from proof of concept to production, then scaling and optimization.

Get started

To prepare for this phase of the cloud adoption lifecycle, use the following exercises:

- Application development and deployment: Examine patterns and practices of application development, configure continuous integration and continuous delivery (CI/CD) pipelines, and implement site reliability engineering (SRE) best practices.
- Cluster design and operations: Identify for cluster configuration and network design. Ensure future scalability by automating infrastructure provisioning. Maintain high availability by planning for business continuity and disaster recovery.
- Cluster and application security: Familiarize yourself with Kubernetes security essentials. Review the secure setup for clusters and application security guidance.
- Enterprise-scale for AKS: Enterprise-scale for AKS is an architectural approach and reference implementation that enables effective workload and scenario operationalization of landing zones on Azure, at scale and aligned with the Azure roadmap and the Microsoft Cloud Adoption Framework for Azure.

Application development and deployment

4/22/2021 • 5 minutes to read • Edit Online

Examine patterns and practices of application development, configure Azure Pipelines, and implement site reliability engineering (SRE) best practices. Site reliability engineering (SRE) is a software engineering approach to application development and deployment, change management, monitoring, emergency response, and more.

Plan, train, and proof

As you get started, use the checklist and application development resources in the following sections to help you plan your application development and deployment. You should be able to answer these questions:

- Have you prepared your application development environment and setup workflow?
- How will you structure the project folder to support Kubernetes application development?
- Have you identified state, configuration, and storage requirements of your application?

Site reliability engineering checklist:

- **Prepare your development environment.** Configure your environment with the tools you need to create containers and set up your development workflow.

  For more information, see:

  - Working with Docker in Visual Studio Code
  - Working with Kubernetes in Visual Studio Code
  - Bridge to Kubernetes overview
- **Containerize your application.** Familiarize yourself with the end-to-end Kubernetes development experience, including application scaffolding, inner-loop workflows, application-management frameworks, CI/CD pipelines, log aggregation, monitoring, and application metrics.

  To learn more, see:

  - Get up and running with Kubernetes (e-book collection)
  - Containerize Your Applications with Kubernetes on Azure (webinar)
- **Review common Kubernetes scenarios.** Although Kubernetes is often thought of as a platform for delivering microservices, it's becoming a much broader platform. Watch this video to learn about common Kubernetes scenarios such as batch analytics and workflow: Overview of common Kubernetes scenarios (video).
- **Prepare your application for Kubernetes.** Prepare your application file system layout for Kubernetes and organize for weekly or daily releases. Learn how the Kubernetes deployment process enables reliable, zero-downtime upgrades.

  For more information, see:

  - Project design and layout for successful Kubernetes applications (webinar)
  - How Kubernetes deployments work (video)
  - Develop and deploy applications on Kubernetes (Microsoft Learn)
- **Manage application storage.** Understand the performance needs and access methods for pods so that you can provide the appropriate storage options. Also, plan for ways to back up and test the restore process for attached storage.

To learn more, see:

- The basics of stateful applications in Kubernetes (video)
- State and data in Docker applications
- Storage options in Azure Kubernetes Service

- **Manage application secrets.** Use a key vault to store and retrieve keys and credentials. Don't store credentials in your application code.

  For more information, see:

  - How Kubernetes and Configuration Management works (video)
  - Understand secrets management in Kubernetes (video)
  - Use Azure Key Vault with Kubernetes
  - Use Azure AD pod identity to authenticate and access Azure resources

Deploy to production and apply best practices

As you prepare the application for production, use the following checklist. You should be able to answer these questions:

- Can you monitor all aspects of your application?
- Have you defined resource requirements for your application? How about scaling requirements?
- Can you deploy new versions of the application without affecting production systems?

SRE best practices checklist:

- **Configure readiness and liveness health checks.** Kubernetes uses readiness and liveness checks to know when your application is ready to receive traffic and when it needs to be restarted. Without defining such checks, Kubernetes won't be able to determine whether your application is running. For more information, see Liveness and readiness checks.

- **Configure logging, application monitoring, and alerting.** Monitoring your containers is critical, especially when you're running a production cluster, at scale, with multiple applications. The recommended logging method for containerized applications is to write to the standard output (`stdout`) and standard error (`stderr`) streams.

  For more information, see:

  - Logging in Kubernetes
  - Get started with monitoring and alerting for Kubernetes (video)
  - Azure Monitor for containers
  - Enable and review Kubernetes control plane logs in Azure Kubernetes Service (AKS)
  - View Kubernetes logs, events, and pod metrics in real time

- **Define resource requirements for the application.** A primary way to manage the compute resources within a Kubernetes cluster is to use pod requests and limits. These requests and limits tell the Kubernetes scheduler what compute resources a pod should be assigned. For more information, see Define pod resource requests and limits.

- **Configure application scaling requirements.** Kubernetes supports horizontal pod autoscaling to adjust the number of pods in a deployment depending on CPU utilization or other select metrics. To use the autoscaler, all containers in your pods must have CPU requests and limits defined. To learn more, see Configure horizontal pod autoscaling.

- **Deploy applications using an automated pipeline and DevOps.** The full automation of all steps between code commit to production deployment allows teams to focus on building code and removes

the overhead and potential human error in manual steps. Deploying new code is quicker and less risky, helping teams become more agile, more productive, and more confident about their running code.

To learn more, see:

- Evolve your DevOps practices
- Setting up a Kubernetes build pipeline (video)
- Deployment Center for Azure Kubernetes Service
- GitHub Actions for deploying to Azure Kubernetes Service
- Tutorial: Deploy from GitHub to Azure Kubernetes Service using Jenkins

Optimize and scale

Now that the application is in production, how can you optimize your workflow and prepare your application and team to scale? Use the optimization and scaling checklist to prepare. You should be able to answer these questions:

- Are cross-cutting application concerns abstracted from your application?
- Are you able to maintain system and application reliability, while still iterating on new features and versions?

Application deployment checklist:

- **Deploy an API gateway.** An API gateway serves as an entry point to microservices, decouples clients from your microservices, adds another layer of security, and decreases the complexity of your microservices by removing the burden of handling cross-cutting concerns. For more information, see Use Azure API Management with microservices deployed in Azure Kubernetes Service.

- **Deploy a service mesh.** A service mesh provides capabilities like traffic management, resiliency, policy, security, strong identity, and observability to your workloads. Your application is decoupled from these operational capabilities and the service mesh moves them out of the application layer and down to the infrastructure layer.

  For more information, see:

  - How service meshes work in Kubernetes (video)
  - Learn about service meshes
  - Use Open Service Mesh with Azure Kubernetes Service
  - Use Istio with Azure Kubernetes Service
  - Use Linkerd with Azure Kubernetes Service
  - Use Consul with Azure Kubernetes Service
- **Implement site reliability engineering (SRE) practices.** Site reliability engineering (SRE) is a proven approach to maintain crucial system and application reliability while iterating at the speed demanded by the marketplace.

  To learn more, see:

  - Introduction to site reliability engineering (SRE)
  - DevOps at Microsoft: game streaming SRE

Cluster design and operations

5/18/2021 • 4 minutes to read • Edit Online

This article covers cluster configuration and network design. Learn how to future-proof scalability by automating infrastructure provisioning. Provisioning is the process of setting up the IT infrastructure that you want. Automated infrastructure provisioning supports a remote installation and sets up virtual environments. It also helps you maintain high availability by planning for business continuity and disaster recovery.

Plan, train, and proof

As you get started, the checklist and Kubernetes resources below will help you plan the cluster design. By the end of this section, you'll be able to answer these questions:

- Have you identified the networking design requirements for your cluster?
- Do you have services with varying requirements? How many node pools are you going to use?

Checklist:

- **Identify network design considerations.** Understand cluster network design considerations, compare network models, and choose the Kubernetes networking plug-in that fits your needs. For Azure container networking interface (CNI) networking, consider the number of IP addresses required as a multiple of the maximum pods per node (default of 30) and number of nodes. Add one node required during upgrade. When choosing load balancer services, consider using an ingress controller when there are too many services to reduce the number of exposed endpoints. For Azure CNI, the service CIDR has to be unique across the virtual network and all connected virtual networks to ensure appropriate routing.

  To learn more, see:

  - Kubenet and Azure CNI networking
  - Use kubenet networking with your own IP address ranges in Azure Kubernetes Service (AKS)
  - Configure Azure CNI networking in Azure Kubernetes Service (AKS)
  - Secure network design for an AKS cluster
- **Create multiple node pools.** To support applications that have different compute or storage demands, you can optionally configure your cluster with multiple node pools. For example, use more node pools to provide GPUs for compute-intensive applications or access to high-performance SSD storage. For more information, see Create and manage multiple node pools for a cluster in Azure Kubernetes Service.
- **Decide on availability requirements.** A minimum of two pods behind Azure Kubernetes Service ensures high availability of your application if there is pod failures or restarts. Use three or more pods to handle load during pod failures and restarts. For the cluster configuration, a minimum of two nodes in an availability set or virtual machine scale set is required to meet the service-level agreement of 99.95%. Use at least three pods to ensure pod scheduling during node failures and reboots.

  To provide a higher level of availability to your applications, clusters can be distributed across Availability Zones. These zones are physically separate datacenters within a given region. When the cluster components are distributed across multiple zones, your cluster can tolerate a failure in one of the zones. Your applications and management operations remain available even if an entire datacenter experiences an outage. For more information, see Create an Azure Kubernetes Service (AKS) cluster that uses Availability Zones.

Go to production and apply infrastructure best practices

As you prepare the application for production, implement a minimum set of best practices. Use this checklist at this stage. By the end of this section, you'll be able to answer these questions:

- Are you able to confidently redeploy the cluster infrastructure?
- Have you applied resource quotas?

Checklist:

- **Automate cluster provisioning.** With infrastructure as code, you can automate infrastructure provisioning to provide more resiliency during disasters and gain agility to quickly redeploy the infrastructure as needed. For more information, see Create a Kubernetes cluster with Azure Kubernetes Service using Terraform.
- **Plan for availability using pod disruption budgets.** To maintain the availability of applications, define pod disruption budgets (PDB) to ensure that a minimum number of pods are available in the cluster during hardware failures or cluster upgrades. To learn more, see Plan for availability using pod disruption budgets.
- **Enforce resource quotas on namespaces.** Plan and apply resource quotas at the namespace level. Quotas can be set on compute resources, storage resources, and object count. For more information, see Enforce resource quotas.

Optimize and scale

Once the application is in production, how can you optimize your workflow and prepare your application and team to scale? Use the optimization and scaling checklist to prepare. By the end of this section, you'll be able to answer these questions:

- Do you have a plan for business continuity and disaster recovery?
- Can your cluster scale to meet application demands?
- Are you able to monitor your cluster and application health and receive alerts?

Checklist:

- **Automatically scale a cluster to meet application demands.** To keep up with application demands, you may need to adjust the number of nodes that run your workloads automatically using the cluster autoscaler. For more information, see Configure Kubernetes cluster autoscaler.
- **Plan for business continuity and disaster recovery.** Plan for multiregion deployment, create a storage migration plan, and enable geo-replication for container images. To learn more, see Best practices for region deployments - Azure Container Registry geo-replication.
- **Configure monitoring and troubleshooting at scale.** Set up alerting and monitoring for applications in Kubernetes. Learn about the default configuration, how to integrate more advanced metrics, and how to add custom monitoring and alerting to operate your application.
  - Get started with monitoring and alerting for Kubernetes (video)
  - Configure alerts using Azure Monitor for containers
  - Review diagnostic logs for master components
  - Azure Kubernetes Service (AKS) diagnostics

Cluster and application security

8/31/2021 • 4 minutes to read • Edit Online

Familiarize yourself with Kubernetes security essentials and review the secure setup for clusters and application security guidance. Kubernetes security is important throughout the container lifecycle because of the distributed, dynamic nature of a Kubernetes cluster. Applications are only as secure as the weakest link in the chain of services that comprise the application's security.

Plan, train, and proof

As you get started, the security essentials checklist and Kubernetes security resources below will help you plan for cluster operations and application security. By the end of this section, you'll be able to answer these questions:

- Have you reviewed the security and threat model of Kubernetes clusters?
- Is your cluster enabled for Kubernetes role-based access control?

Security checklist:

- **Familiarize yourself with the security essentials white paper.** The primary goals of a secure Kubernetes environment are ensuring that the applications it runs are protected, that security issues can be identified and addressed quickly, and that future similar issues will be prevented. For more information, see The definitive guide to securing Kubernetes (white paper).
- **Review the security hardening setup for the cluster nodes.** A security hardened host OS reduces the surface area of attack and allows deploying containers securely. To learn more, see Security hardening in AKS virtual machine hosts.
- **Setup cluster Kubernetes role-based access control (Kubernetes RBAC).** This control mechanism lets you assign users, or groups of users, permission to do things like create or modify resources, or view logs from running application workloads.

  For more information, see

  - Understand Kubernetes role-based access control (Kubernetes RBAC) (video)
  - Integrate Azure AD with Azure Kubernetes Service
  - Limit access to cluster configuration file

Deploy to production and apply Kubernetes security best practices

As you prepare the application for production, implement a minimum set of best practices. Use this checklist at this stage. By the end of this section, you'll be able to answer these questions:

- Have you set up network security rules for ingress, egress, and intra-pod communication?
- Is your cluster set up to automatically apply node security updates?
- Are you running a security scanning solution for your cluster and container services?

Security checklist:

- **Control access to clusters using group membership.** Configure Kubernetes role-based access control (Kubernetes RBAC) to limit access to cluster resources based on user identity or group membership. For more information, see Control access to cluster resources using Kubernetes RBAC and

Azure AD identities.

- **Create a secrets management policy.** Securely deploy and manage sensitive information, such as passwords and certificates, using secrets management in Kubernetes. For more information, see Understand secrets management in Kubernetes (video).
- **Secure intra-pod network traffic with network policies.** Apply the principle of least privilege to control network traffic flow between pods in the cluster. For more information, see Secure intra-pod traffic with network policies.
- **Restrict access to the API server using authorized IPs.** Improve cluster security and minimize attack surface by limiting access to the API server to a limited set of IP address ranges. For more information, see Secure access to the API server.
- **Restrict cluster egress traffic.** Learn what ports and addresses to allow if you restrict egress traffic for the cluster. You can use Azure Firewall or a third-party firewall appliance to secure your egress traffic and define these required ports and addresses. To learn more, see Control egress traffic for cluster nodes in AKS.
- **Secure traffic with Web Application Firewall (WAF).** Use Azure Application Gateway as an ingress controller for Kubernetes clusters. For more information, see Configure Azure Application Gateway as an ingress controller.
- **Apply security and kernel updates to worker nodes.** Understand the AKS node update experience. To protect your clusters, security updates are automatically applied to Linux nodes in AKS. These updates include OS security fixes or kernel updates. Some of these updates require a node reboot to complete the process. To learn more, see Use kured to automatically reboot nodes to apply updates.
- **Configure a container and cluster scanning solution.** Scan containers pushed into Azure Container Registry and gain deeper visibility to your cluster nodes, cloud traffic, and security controls.

  For more information, see:

  - Azure Container Registry integration with Security Center
  - Azure Kubernetes Service integration with Security Center

Optimize and scale

Now that the application is in production, how can you optimize your workflow and prepare your application and team to scale? Use the optimization and scaling checklist to prepare. By the end of this section, you'll be able to answer this question:

- Can you enforce governance and cluster policies at scale?

Security checklist:

- **Enforce cluster governance policies.** Apply at-scale enforcements and safeguards on your clusters in a centralized, consistent manner. To learn more, see Control deployments with Azure Policy.
- **Rotate cluster certificates periodically.** Kubernetes uses certificates for authentication with many of its components. You might want to periodically rotate those certificates for security or policy reasons. To learn more, see Rotate certificates in Azure Kubernetes Service (AKS).

Innovation with AI in the Cloud Adoption Framework

5/18/2021 • 2 minutes to read • Edit Online

The Cloud Adoption Framework is a collection of documentation, implementation guidance, best practices, and tools that are proven guidance from Microsoft designed to accelerate your cloud adoption lifecycle.

Review a prescriptive framework that includes the tools, programs, and content (best practices, configuration templates, and architecture guidance) to simplify adoption of AI and cloud-native practices at scale.

The list of required actions is categorized by persona to drive a successful deployment of AI in applications: from proof of concept to production, and then scaling and optimization.

Get started

To prepare for this phase of the cloud adoption lifecycle, use the following exercises:

- Machine learning model development, deployment, and management: Examine patterns and practices of building your own machine learning models, including machine learning operations (MLOps), automated machine learning (AutoML), and Responsible ML learning tools such as InterpretML and FairLearn.
- Add domain-specific AI models to your applications: Learn about best practices for adding AI capabilities into your applications with Cognitive Services. Also learn about the key scenarios these services help you address.
- Build your own conversational AI solution: Learn how to build your own Virtual Assistant, a conversational AI application that can understand language and speech, perceive vast amounts of information, and respond intelligently.
- Build AI-driven knowledge mining solutions: Learn how to use knowledge mining to extract structured data from your unstructured content and discover actionable information across your organization's data.

Machine learning

5/18/2021 • 2 minutes to read • Edit Online

Machine learning is a subset of AI that allows machines to detect patterns and learn from data without expressly being programed for it. Azure Machine Learning solutions can advance your computing insights.

Azure empowers you with the most advanced machine learning capabilities. Quickly and easily build, train, and deploy your machine learning models by using Azure Machine Learning. Machine learning AI can be used for any kind of machine learning, from classical to deep, supervised, and unsupervised learning. Whether you prefer to write Python or R code, or use zero-code or low-code options such as the designer, you can build, train, and track highly accurate machine learning and deep learning models in a machine learning workspace.

You can even start training on your local machine and then scale out to the cloud. The service also interoperates with popular deep learning and reinforcement open-source tools such as PyTorch, TensorFlow, scikit-learn, and Ray and RLlib.

Get started by reading Machine Learning solutions, where you'll find a tutorial on how to set up your first machine learning experiment. To learn more about the open-source model format and runtime for machine learning, see ONNX Runtime.

Common scenarios for machine learning solutions include:

- Predictive maintenance
- Inventory management
- Fraud detection
- Demand forecasting
- Intelligent recommendations
- Sales forecasting

Machine learning checklist

- **Get started by first familiarizing yourself with Machine Learning**, and then choose which experience to begin with. You can follow along with steps to use a Jupyter notebook with Python, the visual drag-and-drop experience, or automated machine learning (AutoML).
  - Learn about Machine Learning
  - Create your first machine learning experiment with a Jupyter notebook with Python
  - Experiment with visual drag-and-drop
  - Use the AutoML UI
  - Configure your dev environment
- **Experiment with more advanced tutorials** to predict taxi fees, classify images, and build a pipeline for batch scoring.
  - Use AutoML to predict taxi fees
  - Classify images with scikit-learn
  - Build a Machine Learning pipeline for batch scoring
- **Follow along with video tutorials** to learn more about the benefits of machine learning, such as no-code model building, machine learning operations (MLOps), ONNX Runtime, model interpretability and transparency, and more.

    - What's new with Machine Learning
    - Use AutoML to build models
    - Build zero-code models with Machine Learning designer
    - MLOps for managing the end-to-end lifecycle
    - Incorporating ONNX Runtime into your models
    - Model interpretability and transparency
    - Building models with R
- Review reference architectures for AI machine learning solutions.

Next steps

Explore other AI solution categories:

- AI applications and agents
- Knowledge mining

AI applications and agents

5/18/2021 • 4 minutes to read • Edit Online

The applications of AI cover a wide variety of applications, such as electronic trading platforms, remote sensing, robot control, and medical diagnosis. Infusing AI into an application can be difficult and time-consuming. Until recently, you needed a deep understanding of machine learning and months of development to acquire data, train models, and deploy them at scale. Even then, success wasn't guaranteed. The path was filled with obstacles and pitfalls, causing teams to fail to realize value from their AI investments.

AI applications

Microsoft Azure Cognitive Services are domain-specific AI services available as APIs. These services remove the challenges of building AI into your applications and are designed to make them productive, enterprise ready, and trusted. They make it possible for you to build on the latest breakthroughs in AI without building and deploying your own models; instead, you can deploy AI models by using just a few lines of code. Even without a large data science team, you can quickly create AI applications that see, hear, speak, understand, and even begin to reason.

Common examples of AI applications include:

- Sentiment analysis
- Object detection
- Translation
- Personalization
- Robotic process automation

Follow these guidelines to plan your AI application development and deployment:

- Become familiar with the multitude of capabilities and services offered within Azure Cognitive Services, and determine which ones you'll be using.
- Determine whether you have custom data with which you want to train and customize AI models. Certain services can be customized.
- Explore the Azure Cognitive Services quickstart tutorials to learn how to use SDK and REST APIs. The Cognitive Services SDKs are available for many popular dev languages, including C#, Python, Java, JavaScript, and Go.
- Determine whether you need to deploy these Cognitive Services in containers.

AI applications checklist

To get started, first familiarize yourself with the various categories and services within Azure Cognitive Services, such as vision, speech, language, decision, and web search. Visit the product pages to learn more and to interact with demos. Each category on the page offers a set of quick starts, tutorials, and how-to guides for the REST APIs and SDKs. You can also read an e-book that describes common scenarios and how to build your first AI application with Cognitive Services.

- Read the Cognitive Services documentation.
- Explore interactive demos across the product/service pages.
- Read the e-book, Building Intelligent Apps with Cognitive APIs.
- Download the Intelligent Kiosk to experience and demo the services.
- Learn more about Container support in Azure Cognitive Services.

- Review reference architectures for AI solutions.

AI agents

The Microsoft Azure AI platform aims to empower developers to innovate and accelerate their projects. For conversational AI, Azure offers the Azure Bot Service and Bot Framework SDK and tools to enable developers to build rich conversational experiences. Additionally, developers can use Azure Cognitive Services like Language Understanding (LUIS), QnA Maker, and Speech service to add the abilities for chatbots to understand and speak with their end users.

Common scenarios for conversational AI or chatbot solutions include:

- Informational Q&A chatbot
- Customer service or support chatbot
- IT help desk or HR chatbot
- Sales or e-commerce chatbot
- Speech-enabled devices

> **NOTE**
> Microsoft offers Power Virtual Agents, built on top of the Bot Framework, for developers who want to build a chatbot with a primarily no-code or low-code experience. In this scenario, developers don't host the bot themselves and don't control the natural language or other AI models with Cognitive Services.

AI agents checklist

Azure Bot Service and Microsoft Bot Framework have the following characteristics:

- Bot Framework is an open-source offering that provides an SDK to help you design, build, and test your bot. This SDK is available in C#, JavaScript, Python, and Java. It also offers a free visual authoring canvas in Bot Framework Composer and a testing tool in Bot Framework Emulator.
- Azure Bot Service is a dedicated service within Azure that allows you to host or publish your bot in Azure and connect to popular channels.

For more information about Azure Bot Service and Bot Framework, see:

- Azure Bot Service and Bot Framework overview
- Principles of bot design

One of the easiest ways to get started is to use QnA Maker, part of Azure Cognitive Services, which can intelligently convert an FAQ document or website into a Q&A experience in minutes. To use QnA Maker, see:

- Tutorial: Use QnA Maker in your bot to answer questions
- Try out the QnA Maker service

To download and use Bot Framework SDK and tools for bot development, see:

- Latest versions of Bot Framework SDK and tools
- Create your first bot
- Create a bot with the Bot Framework SDK for .NET

To learn how to add Cognitive Services to make your bot even more intelligent, see:

- A Developer's Guide to Building AI Applications (e-book)
- Cognitive Services documentation

To learn how to build your own Virtual Assistant with Bot Framework solution accelerators, and select a common set of skills such as calendar, e-mail, point of interest, and to-do, see the Bot Framework Solutions Documentation.

Next steps

Explore other AI solution categories:

- Machine learning
- Knowledge mining

Knowledge mining

4/22/2021 • 2 minutes to read • Edit Online

Knowledge mining refers to an emerging category of AI designed to simplify the process of accessing the latent insights contained within structured and unstructured data. Knowledge mining defines the process of using an AI pipeline to discover hidden patterns and actionable information from sets of structured and unstructured data in a scalable way. Knowledge mining includes a complex logical understanding and can connect information streams to form real world business insights.

Azure Cognitive Search is a managed cloud search service and solution that gives developers APIs and tools for adding a rich search experience over private, heterogeneous content in web, mobile, and enterprise applications. It offers capabilities such as scoring, faceting, suggestions, synonyms, and geo-search to provide a rich user experience. Azure Cognitive Search is also the only cloud search service with built-in knowledge mining capabilities. Azure Cognitive Search acts as the orchestrator for your knowledge mining enrichment pipeline by following the steps to ingest, enrich, and explore and analyze.

Key scenarios for knowledge mining include:

- **Digital content management:** Help customers consume content more quickly by providing them relevant search results in your content catalog.
- **Customer support and feedback analysis:** Quickly find the right answer in documents and discover trends of what customers are asking for to improve customer experiences.
- **Data extraction and process management:** Accelerate processing documents by extracting key information and populating it in other business documentation.
- **Technical content review and research:** Quickly review documents and extract key information to make informed decisions.
- **Auditing and compliance management:** Quickly identify key areas and flag important ideas or information in documents.

Knowledge mining solutions checklist

- **Get started:** Access free knowledge mining solution accelerators, boot camps, and workshops.
  - Knowledge mining solution accelerator
  - Knowledge mining workshop
  - Knowledge mining boot camp
  - Knowledge mining e-book
  - Knowledge mining Azure DevOps project. For this project, sign in, go to **Cloud Adoption Framework** navigation, and then select **Knowledge Mining**.
- **Use power skills:** Azure search power skills provide useful functions deployable as custom skills for Azure Cognitive Search. You can use these skills as templates or starting points for your own custom skills. You can also deploy and use them as is, if they happen to meet your requirements. We also invite you to contribute your own work by submitting a pull request.
- **Explore additional resources:**
  - Azure Cognitive Search overview
  - Built-in cognitive skills for text and image processing during indexing
  - Documentation resources for AI enrichment in Azure Cognitive Search
  - Design tips and tricks for AI enrichment

- Full text search

Next steps

Explore other AI solution categories:

- AI applications and agents
- Machine learning

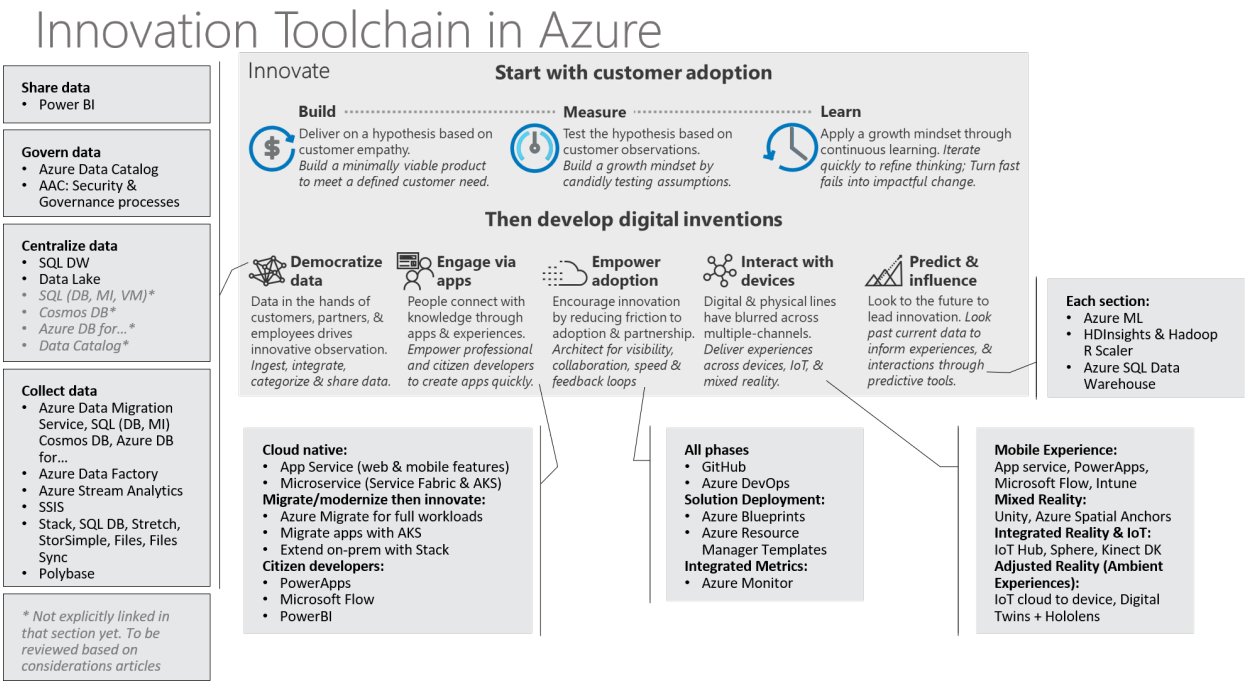
Develop digital inventions in Azure

5/18/2021 • 2 minutes to read • Edit Online

Azure can help accelerate the development of each area of digital invention. This digital best practices section of the Cloud Adoption Framework builds on the Innovate methodology. It shows how you can combine Azure services to create a toolchain for digital invention.

Alignment to the methodology

There are many combinations of cloud-based tools for digital invention and innovation within Azure. The following image shows an overview of how different digital invention tools align to each type of innovation.



Toolchain

The following article series demonstrates a few of the best practices and tools that closely align with the Innovate methodology. To test your hypothesis, start with the overview page of the type of digital invention you require. The overview page has the best practice guidance to act on so that you can build with customer empathy.

Here are the types of digital invention in this article series:

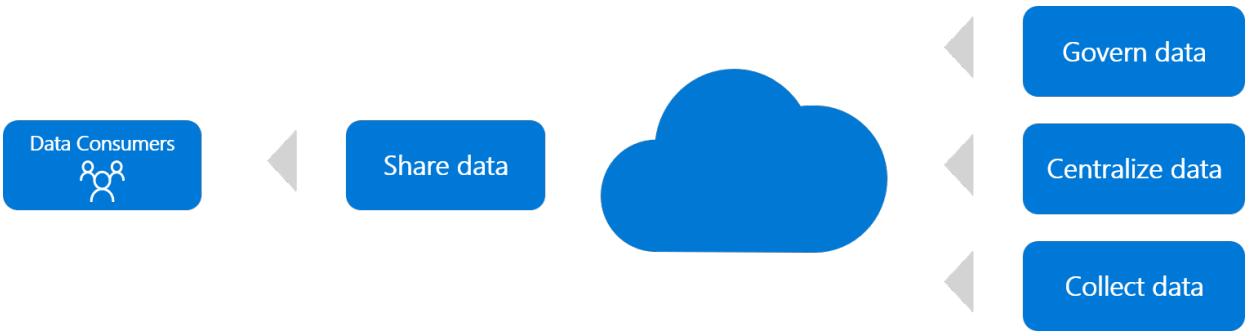
- Democratize data: Tools for sharing data to solve information-related customer needs.
- Engage via applications: Tools to create applications that engage customers beyond raw data.
- Empower adoption: Tools to accelerate customer adoption through digital support for your build-measure-learn cycles.
- Interact with devices: Tools to create different levels of ambient experiences for your customers.
- Predict and influence: Tools for predictive analysis and integration of their output into applications.

Innovation tools to democratize data in Azure

4/22/2021 • 2 minutes to read • Edit Online

As described in the conceptual article on democratizing data, you can deliver many data collection innovations with little technical investment. Many major innovations require little more than raw data. Democratizing data is about investing the least resources needed to engage your customers. The customers then use the data to take advantage of their existing knowledge.

Starting with data democratization is a quick way to test a hypothesis before expanding into broader, more costly digital inventions. As you refine more of the hypothesis and begin to adopt the inventions at scale, the following processes will help you prepare for operational support of the innovation.



Alignment to the methodology

This type of digital invention can be accelerated through each phase of the following processes, as shown in the preceding image. Technical guidance to accelerate digital invention is listed in the table of contents on the left side of this page. Those articles are grouped by phase to align guidance with the overall methodology.

- **Share collected data:** The first step of democratizing data is to share openly.
- **Govern data:** Ensure that any sensitive data is secured, tracked, and governed before sharing.
- **Centralize data:** Sometimes you need to provide a centralized platform for data democratization, sharing, and governance.
- **Collect data:** Migration, integration, ingestion, and virtualization can each collect existing data to be centralized, governed, and shared.

In every iteration, cloud adoption teams should go only as deep into the stack as they require to put the focus on customer needs over architecture. Delaying technical spikes in favor of customer needs accelerates validation of your hypothesis.

All guidance maps to the four preceding processes. Guidance ranges from the highest customer effect to the highest technical effect. Across each process, you'll see guidance on different potential ways that Azure can accelerate your ability to build with customer empathy.

Toolchain

In Azure, the following innovation tools are commonly used to accelerate digital invention across the preceding phases:

- Power BI
- Azure Data Catalog
- Azure Synapse Analytics

- Azure Cosmos DB
- Azure Database for PostgreSQL
- Azure Database for MySQL
- Azure Database for MariaDB
- Azure Database for PostgreSQL hyperscale
- Azure Data Lake Storage
- Azure Database Migration Service
- Azure SQL Database, with or without Azure SQL Managed Instance
- Azure Data Factory
- Azure Stream Analytics
- SQL Server Integration Services
- Azure Stack
- SQL Server Stretch Database
- Azure StorSimple
- Azure Files
- Azure File Sync
- PolyBase

As the invention approaches adoption at scale, the aspects of each solution require refinement and technical maturity. As that happens, more of these services are likely to be required. Use the table of contents on the left side of this page for Azure tools guidance relevant to your hypothesis-testing process.

Get started

The table of contents on the left side of this page outlines many articles. These articles help you get started with each of the tools in this toolchain.

> **NOTE**
> Some links might leave the Cloud Adoption Framework to help you go beyond the scope of this framework.

Classify your organization's data

3/6/2021 • 2 minutes to read • Edit Online

Data classification allows you to determine and assign value to your organization's data and provides a common starting point for governance. The data classification process categorizes data by sensitivity and business impact in order to identify risks. When data is classified, you can manage it in ways that protect sensitive or important data from theft or loss.

Understand data risks, then manage them

Before any risk can be managed, it must be understood. In the case of data breach liability, that understanding starts with data classification. Data classification is the process of associating a metadata characteristic to every asset in a digital estate, which identifies the type of data associated with that asset.

Any asset identified as a potential candidate for migration or deployment to the cloud should have documented metadata to record the data classification, business criticality, and billing responsibility. These three points of classification can go a long way to understanding and mitigating risks.

Classifications Microsoft uses

The following is a list of classifications Microsoft uses. Depending on your industry or existing security requirements, data classification standards might already exist within your organization. If no standard exists, you might want to use this sample classification to better understand your own digital estate and risk profile.

- **Non-business:** Data from your personal life that doesn't belong to Microsoft.
- **Public:** Business data that is freely available and approved for public consumption.
- **General:** Business data that isn't meant for a public audience.
- **Confidential:** Business data that can cause harm to Microsoft if overshared.
- **Highly confidential:** Business data that would cause extensive harm to Microsoft if overshared.

Tagging data classification in Azure

Resource tags are a good approach for metadata storage, and you can use these tags to apply data classification information to deployed resources. Although tagging cloud assets by classification isn't a replacement for a formal data classification process, it provides a valuable tool for managing resources and applying policy. Azure Information Protection is an excellent solution to help you classify data itself, regardless of where it resides (on-premises, in Azure, or somewhere else). Consider it as part of an overall classification strategy.

Take action

Take action by defining and tagging assets with a defined data classification.

- Choose one of the actionable governance guides for examples of applying tags across your portfolio.
- Review the recommended naming and tagging conventions to define a more comprehensive tagging standard.
- For additional information on resource tagging in Azure, see Use tags to organize your Azure resources and management hierarchy.

Next steps

Continue learning from this article series by reviewing the article on securing sensitive data. The next article contains applicable insights if you are working with data that is classified as confidential or highly confidential.

Secure sensitive data

Collect data through the migration and data modernization of existing sources

4/22/2021 • 2 minutes to read • Edit Online

Companies often have different kinds of existing data that they can democratize. When a customer hypothesis requires the use of existing data to build modern solutions, a first step might be data migration and data modernization to prepare for inventions and innovations. To align with existing migration efforts within a cloud adoption plan, do your migration and modernization by using the Migrate methodology.

Use of this article

This article outlines a series of approaches that align with the Migrate methodology. You can best align these approaches to the standard database migration toolchain.

During the assess workloads phase, a cloud adoption team assesses the current state and future state for the migrated asset. When that process is part of an innovation effort, both cloud adoption teams can use this article to help make those assessments.

Primary toolset

When you migrate and modernize on-premises data, the most common Azure tool choice is the Azure Database Migration Service. This service is part of the broader Azure Migrate hub. For existing SQL Server data sources, Data Migration Assistant helps you assess and migrate data structures.

To support Oracle and NoSQL data migrations, you can also use Azure Database Migration Service for certain types of source-to-target databases. For example, migrating Oracle databases to PostgreSQL or migrating MongoDB databases to Azure Cosmos DB. More commonly, adoption teams use partner tools or custom scripts to migrate to Azure Cosmos DB, Azure HDInsight, or virtual machine options based on infrastructure as a service (IaaS).

Considerations and guidance

When you use Azure Database Migration Service for data migration and data modernization, it's important to understand:

- The current platform for hosting the data source.
- The current version.
- The future platform and version that best supports the customer hypothesis or target.

Data migration type

With an offline migration, application downtime starts when the migration starts. For an online migration, downtime is limited to the time to cut over at the end of migration.

Test an offline migration to determine whether the downtime is acceptable. If the restoration time is unacceptable, do an online migration.

The following data migration type table shows source and target pairs to review with the migration team. Each pair includes a tool choice and a link to a related tutorial.

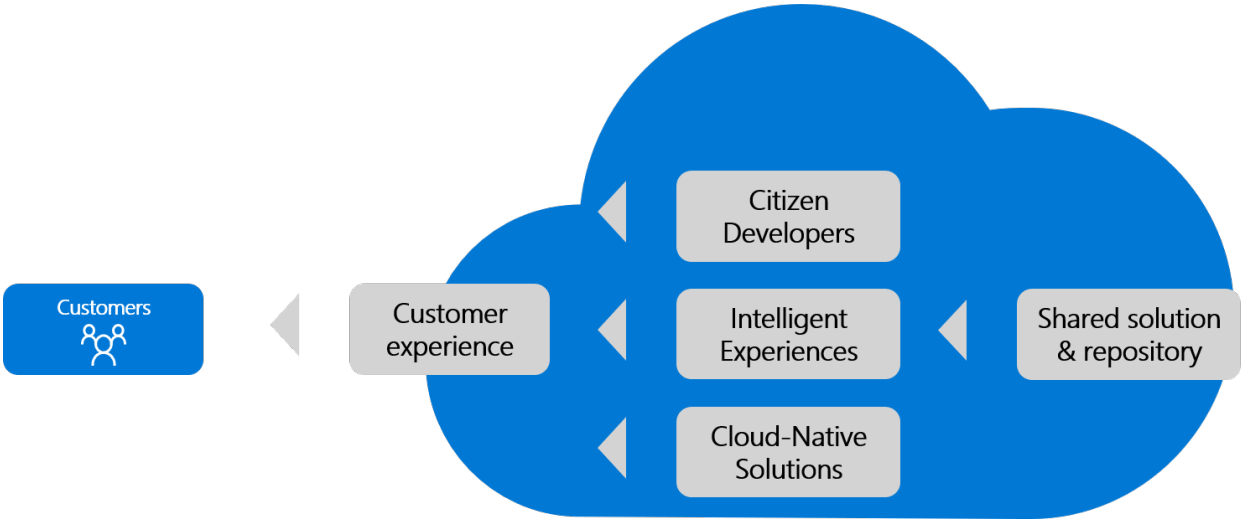
SOURCE	TARGET	TOOL	MIGRATION TYPE	GUIDANCE
SQL Server	Azure SQL Database	Database Migration Service	Offline	Tutorial
SQL Server	Azure SQL Database	Database Migration Service	Online	Tutorial
SQL Server	Azure SQL Managed Instance	Database Migration Service	Offline	Tutorial
SQL Server	Azure SQL Managed Instance	Database Migration Service	Online	Tutorial
RDS SQL Server	Azure SQL Database or Azure SQL Managed Instance	Database Migration Service	Online	Tutorial
MySQL	Azure Database for MySQL	Database Migration Service	Online	Tutorial
PostgreSQL	Azure Database for PostgreSQL	Database Migration Service	Online	Tutorial
MongoDB	Azure Cosmos DB API for MongoDB	Database Migration Service	Offline	Tutorial
MongoDB	Azure Cosmos DB API for MongoDB	Database Migration Service	Online	Tutorial

Tools to engage via applications in Azure

4/22/2021 • 2 minutes to read • Edit Online

Build cloud-native applications to connect customers in new ways. Cloud-native applications are built from the ground up, optimized for cloud scale and performance. They are based on microservices architecture, use managed services, and take advantage of continuous delivery to achieve reliability and faster time to market.

As described in Engage via applications, applications can be an important aspect of a minimal viable product (MVP) solution. For example, applications are often required for testing a hypothesis. This article helps you learn the application development tools that Azure provides to accelerate development of those applications.



Alignment to the innovation methodology

You can accelerate this type of digital invention through each of the following approaches. Technical guidance for accelerating digital invention is listed in the table of contents on the left side of this page. Those articles are grouped by their approaches to aligning guidance with the overall Innovate methodology.

For this article, assume all inventions that result in an application stem from a shared solution as described in Empower adoption. Also assume each application results in some type of customer experience for both internal and external customers.

Based on these assumptions, the following three paths are the most common for cloud adoption teams who are developing digital inventions:

- **Citizen developers:** Before engaging professional developers, business subject matter experts use citizen developer tools. These tools rapidly test and validate that a customer hypothesis can meet the needs of that customer.
- **Intelligent experiences:** Create modern experiences by using cloud platforms to drive rapid deployment and short feedback loops. Expand on web applications to infuse intelligence or even integrate bots.
- **Cloud-native:** Build a new invention that naturally takes advantage of cloud capabilities.

Each path results in advantages and disadvantages that are both short-term and long-term. When the cloud governance team, the cloud operations team, and the cloud center of excellence team are ready to support every approach, you can accelerate adoption with minimal effect on sustainable business operations.

Toolchain

Depending on the path that the cloud adoption team takes, Azure provides application development services and tools to accelerate the team's ability to build with customer empathy in mind. The following list of Azure offerings is grouped based on the preceding decision paths. These offerings include:

- Azure App Service
- Azure Kubernetes Service (AKS)
- Azure Migrate
- Azure Stack
- Power Apps
- Power Automate
- Power BI

Get started

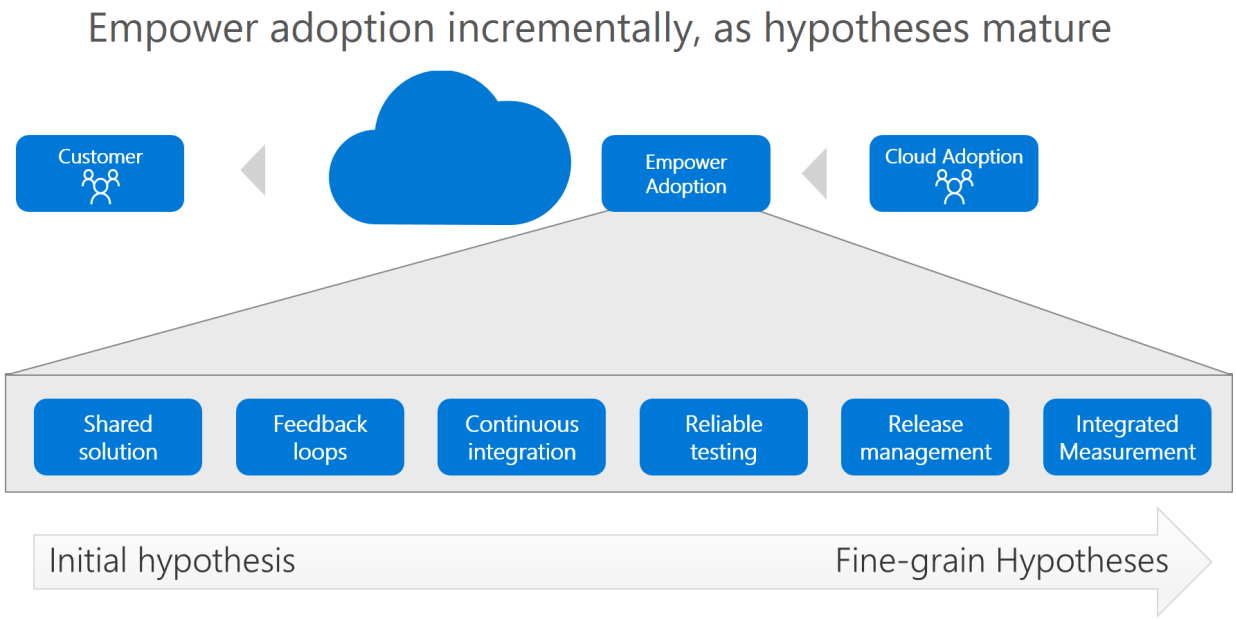
The table of contents on the left side of this page outlines many articles. These articles help you get started with each of the tools in this toolchain.

> **NOTE**
> Some links might leave the Cloud Adoption Framework to help you go beyond the scope of this framework.

Tools to empower adoption in Azure

4/22/2021 • 2 minutes to read • Edit Online

As described in Empower adoption, building true innovation at scale requires an investment in removing friction that could slow adoption. In the early stages of testing a hypothesis, a solution is small. The investment in removing friction is likely small as well. As hypotheses prove true, the solution and the investment in empowering adoption grows. This article provides key continuous integration links to help you get started with each stage of the maturity model.



Alignment to the innovation methodology

You can accelerate this type of digital invention through the following levels of maturity. Technical guidance to accelerate digital invention is listed in the table of contents on the left side of this page. Those articles are grouped by maturity model level.

- **Shared solution:** Establish a centralized repository for all aspects of the solution.
- **Feedback loops:** Ensure feedback loops can be managed consistently throughout iterations.
- **Continuous integration:** Regularly build and consolidate a continuous integration and continuous deployment (CI/CD) solution.
- **Reliable testing:** Validate solution quality and expected changes to drive ensuring measurements.
- **Solution deployment:** Deploy a solution to allow a team to quickly share changes with customers.
- **Integrated measurement:** Add learning metrics to the feedback loop for clear analysis by the full team.

Toolchain

For adoption teams that are mature professional development teams with many contributors, the Azure toolchain starts with GitHub and Azure DevOps.

As your need grows, you can expand this foundation to use other tool features. The expanded foundation might involve tools like:

- Azure Blueprints
- Azure Policy

- Azure Resource Manager templates
- Azure Monitor

The table of contents on the left side of this page lists guidance for each tool and aligns with the previously described maturity model.

Get started

The table of contents on the left side of this page outlines many articles. These articles help you get started with each of the tools in this toolchain.

> **NOTE**
> Some links might leave the Cloud Adoption Framework to help you go beyond the scope of this framework.

MLOps with Azure Machine Learning

5/4/2021 • 2 minutes to read • Edit Online

MLOps (machine learning operations) is based on DevOps principles and practices that increase workflow efficiencies like continuous integration, delivery, and deployment. MLOps applies these principles to the machine learning process in order to:

- Experiment and develop models more quickly.
- Deploy models to production more quickly.
- Practice and refine quality assurance.

Azure Machine Learning provides the following MLOps capabilities:

- **Create reproducible pipelines.** Machine Learning pipelines enable you to define repeatable and reusable steps for your data preparation, training, and scoring processes.
- **Create reusable software environments** for training and deploying models.
- **Register, package, and deploy models from anywhere.** You can track the associated metadata required to use the model.
- **Capture the governance data for the end-to-end lifecycle.** The logged information can include who is publishing models, why changes were made, and when models were deployed or used in production.
- **Notify and alert on events in the lifecycle.** For example, you can get alerts for experiment completion, model registration, model deployment, and data drift detection.
- **Monitor applications for operational and machine learning-related issues.** Compare model inputs between training and inference, explore model-specific metrics, and provide monitoring and alerts on your machine learning infrastructure.
- **Automate the end-to-end machine learning lifecycle with Azure Machine Learning and Azure Pipelines.** With pipelines, you can frequently update models, test new models, and continuously roll out new machine learning models alongside your other applications and services.

Best practices for MLOps with Azure Machine Learning

Models differ from code because they have an organic shelf life and will deteriorate unless maintained. After they're deployed, they can add real business value, and this gets easier when data scientists are given the tools to adopt standard engineering practices.

MLOps with Azure helps you:

- Create reproducible models and reusable training pipelines.
- Simplify model packaging, validation, and deployment for quality control and a/B testing.
- Explain and observe model behavior, and automate the retraining process.

MLOps improves the quality and consistency of your machine learning solutions. To learn more about how to use Azure Machine Learning to manage the lifecycle of your models, see MLOps: model management, deployment, and monitoring with Azure Machine Learning.

Next steps

Learn more by reading and exploring the following resources:

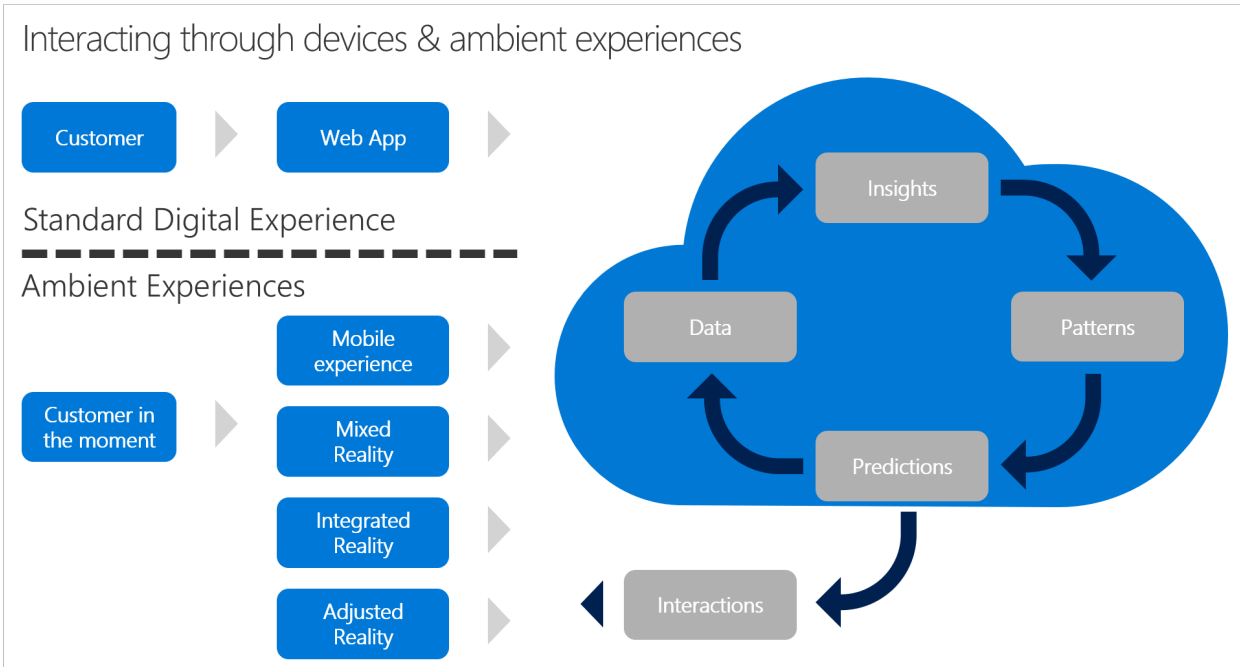
- MLOps: model management, deployment, and monitoring with Azure Machine Learning

- How and where to deploy models with Azure Machine Learning
- Tutorial: deploy an image classification model in Azure Container Instances
- End-to-end MLOps examples repo
- CI/CD of machine learning models with Azure Pipelines
- Create clients that consume a deployed model
- Machine learning at scale
- Azure AI reference architectures and best practices repo

Ambient experience tools to interact with devices in Azure

4/22/2021 • 2 minutes to read • Edit Online

As described in interacting with devices, the devices used to interact with a customer depend on how much ambient user experience is required to deliver the customer's need and empower adoption. Speed from the trigger that prompts the customer's need and your solution's ability to meet that need are determining factors in repeat usage. Ambient experiences help accelerate that response time and create a better experience for your customers by embedding your solution in the customers' immediate surroundings.



Alignment to the methodology

This type of digital invention can be delivered through any of the following levels of ambient experience. The table of contents on the left side of this page lists technical guidance to accelerate digital invention. Those device interaction articles are grouped by level of ambient user experience to align with the methodology.

- **Mobile experience:** Mobile apps are commonly part of a customer's surroundings. In some scenarios, a mobile device might provide enough interactivity to make a solution ambient.
- **Mixed reality:** Sometimes a customer's natural surroundings must be altered through mixed reality. Engaging a customer within that mixed reality can provide a form of ambient user experience.
- **Integrated reality:** Moving closer to true ambience, integrated reality solutions focus on the use of a customer's physical device to integrate the solution into natural behaviors.
- **Adjusted reality:** When any of the preceding solutions use predictive analysis to provide an interaction with a customer within that customer's natural surroundings, that solution creates the highest form of ambient experience.

Toolchain

In Azure, you commonly use the following tools to accelerate digital invention across each level of ambient user experience solutions. These tools are grouped based on the amount of experience required to reduce complexity in aligning tools with those experiences.

CATEGORY	TOOLS
Mobile experiences	• Azure App Service • Power Apps • Power Automate • Intune
Mixed reality	• Unity • Azure Spatial Anchors • HoloLens
Integrated reality	• Azure IoT Hub • Azure Sphere • Azure Kinect DK
Adjusted reality	• IoT cloud to device • Azure Digital Twins + HoloLens

Get started

The table of contents on the left side of this page outlines many articles. These articles help you get started with each of the tools in this toolchain.

> **NOTE**
> Some links might leave the Cloud Adoption Framework to help you go beyond the scope of this framework.

Use innovation tools with AI to predict and influence

3/6/2021 • 2 minutes to read • Edit Online

AI means *artificial intelligence*, in which a computer detects patterns of data to produce insights that can help businesses understand the behavior of their customers. AI predictions can predict customer needs and automate business processes. Using AI applications and digital innovation tools, companies can discover information lying latent in unstructured data and deliver new ways to engage with customers to deliver better experiences.

You can accelerate this type of digital innovation through each of the following solution areas. Best practices and technical guidance to accelerate digital invention are listed in the table of contents on the left side of this page. Those articles are grouped by solution area:

- Machine learning
- AI applications and AI agents
- Knowledge mining

Considerations when starting your digital innovation journey with AI applications

AI strategy, AI culture, responsible and scalable AI, and AI for each of the different business personas are crucial elements for any business to consider when planning your AI prediction and innovation journey with AI.

- **AI strategy:** Industry leaders follow a framework for creating an AI vision that can be applied strategically across an organization. Questions around the value creation perspective, organization and execution perspective, and industry environment perspective define the AI environment.
- **AI culture:** To successfully develop an AI culture, key changes might be required to become AI-ready. This includes the ability to explore AI potential across the business, recognition and fostering of the skills and roles required to make AI success, providing examples of successful implementations of AI with relevant scenarios in finance, marketing, sales, and customer service and AI application evangelism.
- **Responsible AI:** Responsible AI is a commitment to the advancement of AI, driven by principles that put people first. Microsoft AI principles and resources support trusted AI within infrastructure and business frameworks. To successfully build an AI practice, responsible AI should be incorporated into your digital innovation approach.
- **Scalable AI:** Scalable AI enables fueling innovation tools at all levels, evaluating AI investments, and establishing technical processes for AI throughout your organization. AI patterns and best practices allow scaling AI horizontal and vertical through the enterprise.
- **AI for business users:** AI can empower everyone within an organization to achieve more, not just developers and data scientists. Over time, all users should understand the AI applications, concepts, tools, and technologies that make it possible.
- **AI for business leaders:** Understanding state-of-the-art AI technology by holistically exploring machine learning concepts and how they can be used to optimize business across industries, and develop an understanding on how these innovative tools and technologies can benefit your business.

Get started

The articles shown in the table of contents will help you get started with each of the solution areas.

> **NOTE**
>
> Some links might leave the Cloud Adoption Framework to help you go beyond the scope of this framework.

What is machine learning?

3/6/2021 • 6 minutes to read • Edit Online

Machine learning is a data science technique that allows computers to use existing data to forecast future behaviors, outcomes, and trends. By using machine learning, computers learn without being explicitly programmed. Machine learning tools use AI systems which provide the ability to identify patterns and create associations from experience with the data.

Automated machine learning forecasts or predictions can make applications and devices smarter. For example, when you shop online, machine learning helps recommend other products you might want based on what you've bought. Or when your credit card is swiped, machine learning compares the transaction to a database of transactions and helps detect fraud. And when your robot vacuum cleaner vacuums a room, machine learning helps it decide whether the job is done.

Machine learning tools to fit each task

Azure Machine Learning provides all the tools developers and data scientists need for their machine learning workflows, including:

- The Azure Machine Learning designer (preview): Drag-n-drop modules to build your experiments and then deploy pipelines
- Jupyter notebooks: Use our example notebooks or create your own notebooks to use our SDK for Python samples.
- R scripts or notebooks in which you use the SDK for R to write your own code, or use the R modules in the designer.
- The many models solution accelerator (preview) builds on Azure Machine Learning and enables you to train, operate, and manage hundreds or even thousands of machine learning models.
- Visual Studio Code extension.
- Machine learning CLI.
- Open-source frameworks such as PyTorch, TensorFlow, and scikit-learn and many more
- Reinforcement learning with Ray RLlib.

You can even use MLflow to track metrics and deploy models or kubeflow to build end-to-end workflow pipelines.

Build machine learning models in Python or R

Start training on your local machine using the Azure Machine Learning Python SDK or R SDK. Then, you can scale out to the cloud. With many available compute targets, like Azure Machine Learning compute and Azure Databricks, and with advanced hyperparameter tuning services, you can build better models faster by using the power of the cloud. You can also automate model training and tuning using the SDK.

Build machine learning models with no-code tools

For code-free or low-code training and deployment, try:

- Azure Machine Learning designer (preview)

  Use the designer to prep data, train, test, deploy, manage, and track machine learning models without writing any code. There is no programming required, you visually connect datasets and modules to

construct your model. Try out the designer tutorial.

Learn more in the Azure Machine Learning designer overview article.

- Automated machine learning (AutoML) UI

  Learn how to create AutoML experiments in the easy-to-use interface.

MLOps: Deploy and lifecycle management

Machine learning operations (MLOps) is based on DevOps principles and practices that increase the efficiency of workflows. For example, continuous integration, delivery, and deployment. MLOps applies these principles to the machine learning process, with the goal of:

- Faster experimentation and development of models
- Faster deployment of models into production
- Quality assurance

When you have the right model, you can easily use it in a web service, on an IoT device, or from Power BI. For more information, see Deploy models with Azure Machine Learning.

Then you can manage your deployed models by using the Azure Machine Learning SDK for Python, Azure Machine Learning studio, or the Machine learning CLI.

These models can be consumed and return predictions in Real time or asynchronously on large quantities of data.

And with advanced machine learning pipelines, you can collaborate on each step from data preparation, model training and evaluation, through deployment. Pipelines allow you to:

- Automate the end-to-end machine learning process in the cloud
- Reuse components and only rerun steps when needed
- Use different compute resources in each step
- Run batch scoring tasks

If you want to use scripts to automate your machine learning workflow, the Machine learning CLI provides command-line tools that perform common tasks, such as submitting a training run or deploying a model.

To get started using Azure Machine Learning, see Next steps.

Automated Machine Learning

Data scientists spend an inordinate amount of time iterating over models during the experimentation phase. The whole process of trying out different algorithms and hyperparameter combinations until an acceptable model is built is extremely taxing for data scientists, due to the monotonous and non-challenging nature of work. While this is an exercise that yields massive gains in terms of the model efficacy, it sometimes costs too much in terms of time and resources and thus may have a negative return on investment (ROI).

This is where automated machine learning (AutoML) comes in. It uses the concepts from the research paper on probabilistic matrix factorization and implements an automated pipeline of trying out intelligently-selected algorithms and hypermeter settings, based on the heuristics of the data presented, keeping into consideration the given problem or scenario. The result of this pipeline is a set of models that are best suited for the given problem and dataset.

For more information on AutoML, see AutoML and MLOps with Azure Machine Learning.

Responsible ML

Throughout the development and use of AI systems, trust must be at the core. Trust in the platform, process, and models. As AI and autonomous systems integrate more into the fabric of society, it's important to proactively make an effort to anticipate and mitigate the unintended consequences of these technologies.

- **Understand your models and build for fairness:** Explain model behavior and uncover features that have the most impact on predictions. Use built-in explainers for both glass-box and black-box models during model training and inference. Use interactive visualizations to compare models and perform what-if analysis to improve model accuracy. Test your models for fairness using state-of-the-art algorithms. Mitigate unfairness throughout the machine learning lifecycle, compare mitigated models, and make intentional fairness versus accuracy trade-offs as desired.
- **Protect data privacy and confidentiality:** Build models that preserve privacy using the latest innovations in differential privacy, which injects precise levels of statistical noise in data to limit the disclosure of sensitive information. Identify data leaks and intelligently limit repeat queries to manage exposure risk. Use encryption and confidential machine learning (coming soon) techniques specifically designed for machine learning to securely build models using confidential data.
- **Control and govern through every step of the machine learning process:** Access built-in capabilities to automatically track lineage and create an audit trial across the machine learning lifecycle. Obtain full visibility into the machine learning process by tracking datasets, models, experiments, code, and more. Use custom tags to implement model data sheets, document key model metadata, increase accountability, and ensure responsible process.

Learn more about how to implement Responsible ML.

Integration with other services

Azure Machine Learning works with other services on the Azure platform, and also integrates with open-source tools such as Git and MLflow.

- **Compute targets such as Azure Kubernetes Service, Azure Container Instances, Azure Databricks, Azure Data Lake Analytics, and Azure HDInsight:** For more information on compute targets, see What are compute targets?.
- **Azure Event Grid:** For more information, see Consume Azure Machine Learning events.
- **Azure Monitor:** For more information, see Monitoring Azure Machine Learning.
- **Data stores such as Azure Storage accounts, Azure Data Lake Storage, Azure SQL Database, Azure Database for PostgreSQL, and Azure open datasets:** For more information, see Access data in Azure Storage services and create datasets with Azure open datasets.
- **Azure Virtual Network:** For more information, see Secure experimentation and inference in a virtual network.
- **Azure Pipelines:** For more information, see Train and deploy machine learning models.
- **Git repository logs:** For more information, see Git integration.
- **MLflow:** For more information, see MLflow to track metrics and deploy models.
- **Kubeflow:** For more information, see Build end-to-end workflow pipelines.
- **Secure communications:** Your Azure Storage account, compute targets, and other resources can be used securely inside a virtual network to train models and perform inference. For more information, see Secure experimentation and inference in a virtual network.

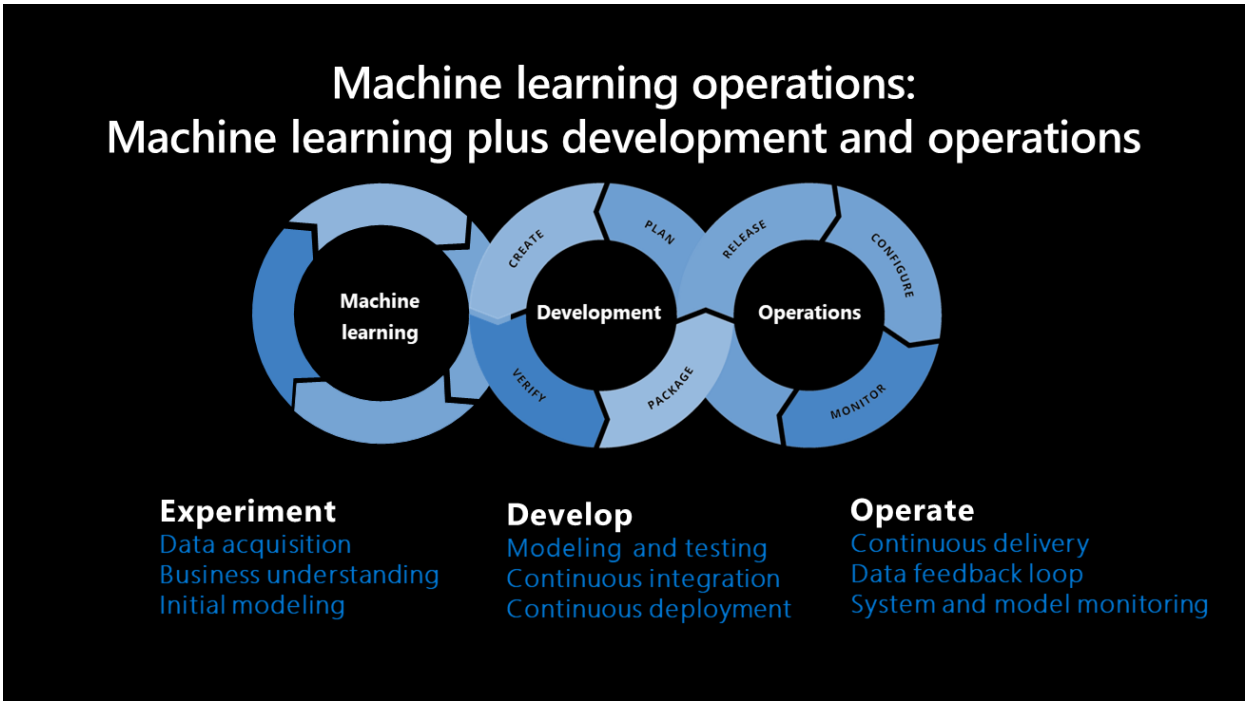
Next steps

- Review machine learning white papers and e-books on Machine Learning studio and Machine Learning service.
- Review AI + Machine Learning architectures.

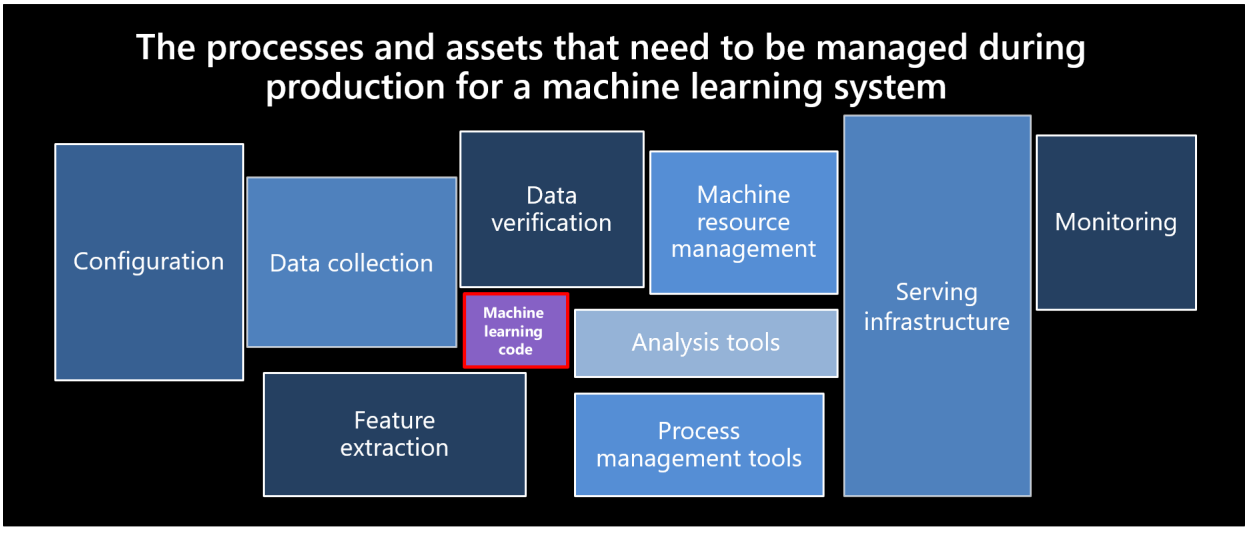
How to approach machine learning operations

4/22/2021 • 4 minutes to read • Edit Online

Machine learning operations consist of principles and best practices about how to organize and standardize machine model development, deployment, and maintenance in a scalable way.



The main components of how a machine learning system develops are outlined below:



Sculley, et al. 2015. Hidden technical debt in machine learning systems. Proceedings of the 28th International Conference on Neural Information Processing Systems, Volume 2 (NIPS 2015).

Machine learning operations vs. development operations

While development operations (DevOps) influence machine learning operations, there are differences between their processes. In addition to DevOps practices, machine learning operations address the following concepts not covered in DevOps:

- **Data versioning:** There must be code versioning and dataset versioning, as the schema and actual data

can change over time. This allows data to be reproduced, makes the data visible other team members, and helps use cases to be audited.

- **Model tracking:** Model artifacts are often stored in a model registry that should identify storage, versioning, and tagging capabilities. These registries need to identify the source code, its parameters, and the corresponding data used to train the model, all of which indicate where a model was created.
- **Digital audit trail:** When working with code and data, all changes need to be tracked.
- **Generalization:** Models are different than code for reuse, as models must be tuned based on the input data or scenario. You might need to fine-tune the model for the new data to use it for a new scenario.
- **Model retraining:** Model performance can decrease over time, and it's important to retrain models for them to remain useful.

Approaches to machine learning operations

Data scientists within an organization apply a broad spectrum of experience, maturity, skills, and tools to experimenting with machine learning operations. Since it's important to encourage as many participants as possible to embrace AI, a consensus about how all organizations should approach machine learning operations isn't likely or desirable. In light of this variety, a clear starting point for your organization is to understand how its size and resources will influence its approach to machine learning operations.

A company's size and maturity indicate if data science teams or individuals with unique roles will own the machine learning lifecycle and navigate each phase. The following approaches to the lifecycle are the most common to each scenario:

A centralized approach

Data science teams will likely monitor the machine learning lifecycle in small companies with limited resources and specialists. This team applies their technical expertise to cleaning and aggregating data, developing and deploying models, and monitoring and maintaining deployed models.

One advantage of this method is that it progresses the model quickly to production, but it increases costs because of the specialized skill levels that need to be maintained on the data science team. Quality suffers when those required levels of expertise aren't present.

A decentralized approach

Individuals with specialized roles will likely monitor and be responsible for the machine learning lifecycle in companies with dedicated operations teams. Once a use case is approved, a machine learning engineer is assigned to assess its current state and the amount of work needed to elevate it to format that can be supported.

The data scientist will need to gather information for the following questions:

- Who will be the business owner?
- How will the model be consumed?
- What level of service availability will be needed?
- How will the model be retrained?
- How often will the model be retrained?
- Who will decide the conditions for model promotion?
- How sensitive is the use case, and is the code shareable?
- How will the model and code be modified in the future?
- How much of the current experiment is reusable?
- Are there existing project workflows that can assist?
- How much work will be required to advance the model to production?

Next, a machine learning engineer designs the workflow and estimates the amount of work required. One best practice is to involve the data scientist(s) in building out the workflow; this time presents a key opportunity to cross-train and familiarize them with the final repo since it's common for the data scientist to work on the use case in the future.

How machine learning operations benefit business

Machine learning operations connect traditional development operations, data operations, and data science/AI. Understanding how machine learning operations can benefit your business will help your AI journey.

Integrating machine learning operations with your business can create the following benefits:

- Enterprise model management streamlines and automates the lifecycle for model development, training, deployment, and operationalization. This allows businesses to be agile and respond to immediate needs and business changes in a repeatable and managed way.
- Model versioning and data realization allow the enterprise to generate iterated and versioned models to adjust to the nuances of the data or the particular use case. This provides flexibility and agility in responding to business challenges and changes.
- When organizations monitoring and manage their models, this helps them to quickly respond to significant changes in the data or the scenario. For example, an implemented model might experience extreme data drift because of an external factor or a change in the underlying data. This would make the previous models unusable and require the current model to be retrained as soon as possible. Machine learning models to be tracked for accuracy and performance. It alerts stakeholders when changes impact model reliability and performance, which leads to quick retraining and deployment.
- Applied machine learning operations processes support business outcomes by allowing rapid auditing, compliance, governance, and access control throughout the development lifecycle. The visibility of model generation, data usage, and regulatory compliance is clear as changes take place in the business.

Next steps

- The Microsoft AI Business School is a resource that outlines AI, including how to approach implementation holistically, understand dependencies beyond the technology, and drive lasting business impact.
- Read further about the machine learning operations process to explore this in more detail.

The machine learning operations process

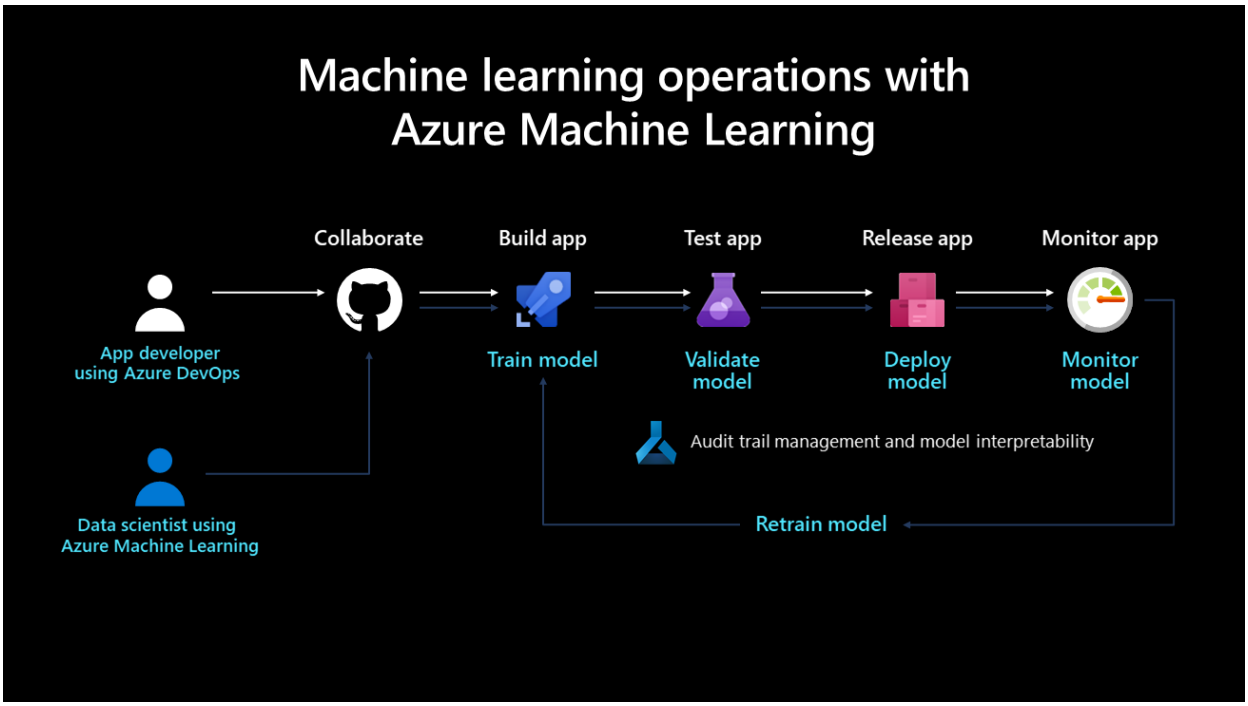
5/18/2021 • 6 minutes to read • Edit Online

The model development process

The development process should produce the following outcomes:

- Training is automated, and models are validated, which includes testing functionality and performance (for example, using accuracy metrics).
- Deployment to the infrastructure used for inferencing (including monitoring) is automated.
- Mechanisms create an end-to-end data audit trail. Automatic model retraining occurs when data drifts over time, which is relevant to large-scale, machine-learning-infused systems.

The following diagram depicts the deployment lifecycle of a machine learning system:



Once developed, a machine learning model is trained, validated, deployed, and monitored. From an organizational perspective and on the managerial and technical level, it's important to define who owns and implements this process. In larger enterprises, a data scientist might own the model training and validation steps, and a machine learning engineer might tend to the remaining steps. In smaller companies, a data scientist might own all steps.

Train the model

In this step, a training dataset trains the machine learning model. The training code is version-controlled and reusable, and this feature optimizes button clicks and event triggers (such as a new version of the data becoming available) to automate how the model is trained.

Validate the model

This step uses established metrics like an accuracy metric to automatically validate the newly trained model and compare it to older ones. Did its accuracy increase? If yes, this model might be registered in the model registry for ensuring that next steps can consume it. If the new model performs worse, then a data scientist can be alerted to investigate why or discard the newly trained model.

Deploy the model

Deploy the model as an API service for web applications in the deployment step. This approach enables the model to be scaled and updated independently of the applications. Alternatively, the model can be used to perform batch scoring where it's used once or periodically to calculate predictions on new data points. This is useful when large amounts of data need to be processed asynchronously. More details on deployment models can be found on the machine learning inference during deployment page.

Monitor the model

It's necessary to monitor the model for two key reasons. First, monitoring the model helps to ensure that it's technically functional; for example, able to generate predictions. This is important if an organization's applications depend on the model and use it in real time. Monitoring the model also helps organizations to gauge if it continuously generates useful predictions. This might not be useful when data drift occurs, such as when the data used to train the model significantly differs from the data that's sent to the model during prediction phase. For example, a model trained for recommending products to young people might produce undesirable results when recommending products to people from a different age group. Model monitoring with data drift can detect this type of mismatch, alert machine learning engineers, and automatically retrain the model with more relevant or newer data.

How to monitor models

Since data drift, seasonality, or newer architecture tuned for better performance can all cause model performance to wane over time, it's important to establish a process to continuously deploy models. Some best practices include:

- **Ownership:** An owner should be assigned to the model performance monitoring process to actively manage its performance.
- **Release pipelines:** Set up a release pipeline in Azure DevOps first, and set the trigger to the model registry. When a new model is registered in the registry, the release pipeline triggers and signs off on a deployment process.

Prerequisites for retraining models

Collecting data from models in production is one prerequisite to retraining models in a continuous integration/continuous development framework, and this process uses input data from scoring requests. This capability is currently limited to tabular data that can be parsed as JSON with minimal formatting and manipulation; video, audio, and images are excluded. This capability is available for models on the Azure Kubernetes Service (AKS). The collected data is stored in an Azure blob.

To prepare for retraining a model:

1. **Monitor data drift from the input data collected.** Setting up a monitoring process requires extracting the timestamp from the production data. This is required to compare the production data and the baseline data (the training data used to build the model). The preferred way to monitor data drift is through Azure Monitor Application Insights. This feature provides an alert that can trigger actions like email, SMS text, push, or Azure Functions. You need to enable Application Insights to log data.
2. **Analyze the collected data.** Make sure to collect data from models in production, and include the results in the model scoring script. Collect all features used for model scoring, as this ensures that all necessary features are present and can be used as training data.
3. **Decide whether retraining with the collected data is necessary.** Many things cause data drift, including sensor issues to seasonality, changes in user behavior, and data quality issues related to the data source. Model retraining isn't required in all cases, so it's recommended to investigate and understand the cause of the data drift before pursuing this.
4. **Retrain the model.** Model training should already be automated, and this step involves triggering the

current training step. This could be for when data drift has been detected (and it isn't related to a data issue), or when a data engineer has published a new version of a dataset. Depending on the use case, these steps can be fully automated or supervised by a human. For example, while some use cases like product recommendations could run autonomously in the future, others in finance would factor standards like model fairness and transparency and require a human to approve newly trained models.

At first, it's common for an organization to only automate a model's training and deployment but not the validation, monitoring, and retraining steps, which are performed manually. Eventually, automation steps for these tasks can progress until the desired state is achieved. DevOps and machine learning operations are concepts that develop over time, and organizations should be aware of their evolution.

The Team Data Science Process lifecycle

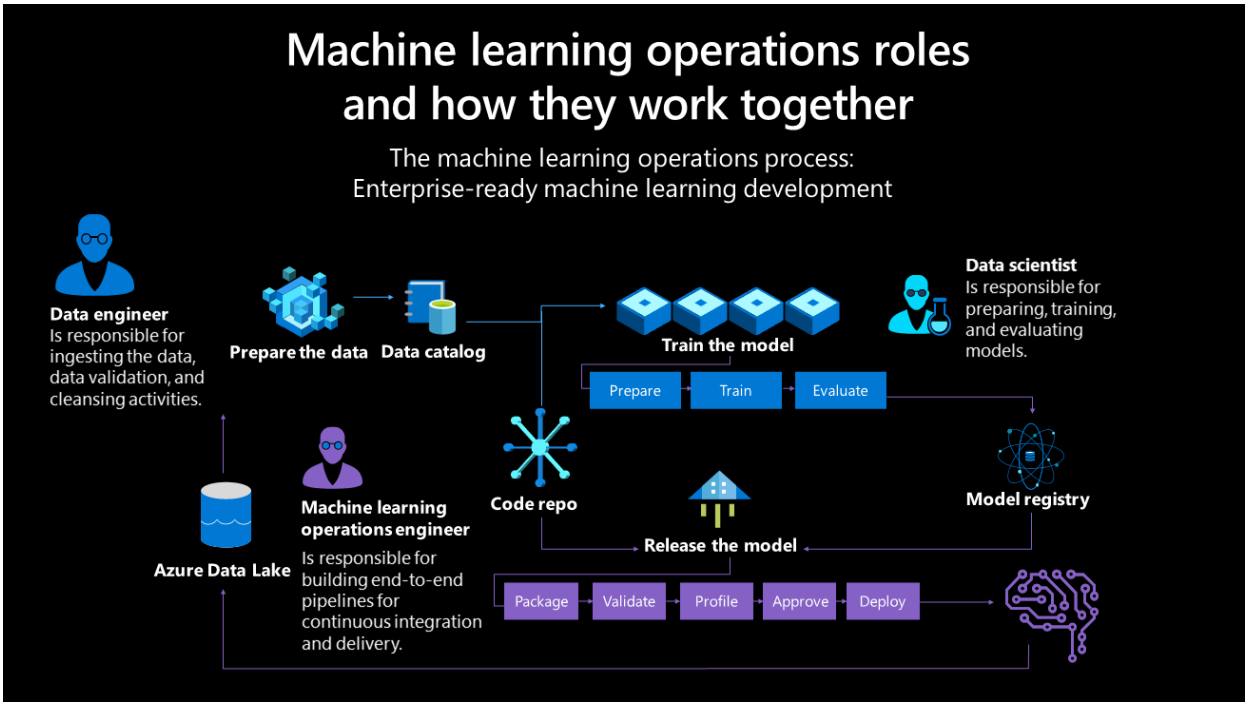
The Team Data Science Process (TDSP) provides a lifecycle to structure the development of your data science projects. The lifecycle outlines the major stages that projects typically execute, often iteratively:

- Business understanding
- Data acquisition and understanding
- Modeling
- Deployment

The goals, tasks, and documentation artifacts for each stage of the TDSP lifecycle are described in the Team Data Science Process lifecycle.

The roles and activities within machine learning operations

Per the TDSP lifecycle, the key roles in the AI project are data engineer, data scientist, and machine learning operations engineer. These roles are critical to your project's success and must work together toward accurate, repeatable, scalable, and production-ready solutions.



- **Data engineer:** This role ingests, validates, and cleans the data. Once the data is refined, it's cataloged and made available for data scientists to use. At this stage, it's important to explore and analyze duplicate data, remove outliers, and identify missing data. These activities should be defined in the pipeline steps and are executed as the train pipeline is preprocessed. Unique and specific names should be assigned to core and generated features.

- **Data scientist (or AI engineer):** This role navigates the training pipeline process and evaluates models. A data scientist receives data from the data engineer and identifies patterns and relationships within it, possibly selecting or generating features for the experiment. Since feature engineering plays a major role in building a sound generalized model, it's key for this phase to be completed as thoroughly as possible. Various experiments can be performed with different algorithms and hyperparameters. Azure tools like automated machine learning can automate this task, which can also help with underfitting or overfitting a model. A successfully trained model is then registered in the model registry. The model should have a unique and specific name, and a version history should be retained for traceability.
- **Machine learning operations engineer:** This role builds end-to-end pipelines for continuous integration and delivery. This includes packing the model in a Docker image, validating and profiling the model, awaiting approval from a stakeholder, and deploying the model in a container orchestration service such as AKS. Various triggers can be set during continuous integration, and the model's code can trigger the train pipeline and the release pipeline afterward.

Machine learning security

3/6/2021 • 2 minutes to read • Edit Online

Machine learning presents unique security considerations for businesses, and companies should consider several security principles when designing and assessing machine learning architectures.

- **Resilience:** Machine learning systems should identify abnormal behavior and prevent manipulation or coercion.
- **Discretion:** Any data access scenarios involving AI should limit access to the data to the minimal extent.
- **Malicious data:** Machine learning algorithms must protect against maliciously introduced data.
- **Transparency and accountability:** AI must have built-in forensic capabilities that provide transparency and accountability. These capabilities function as an early form of AI intrusion detection to help engineers understand the exact point in time a decision was made and the data that was used.
- **Secure environments:** Access to development, training, and inference environments must be secured to protect data and model results/predictions.

Deploy Azure Kubernetes Service to secure an inference environment

Azure Kubernetes Service (AKS) is recommended for inference in a production environment. Two options are available in your virtual network:

- Deploy or attach an AKS cluster with a public IP address to your virtual network.
- Attach a private AKS cluster to your virtual network.

Default AKS clusters have a control plane with a public IP address. If you need a private IP address for your AKS cluster, use the private AKS cluster. For inference, the AKS cluster and Azure Machine Learning workspace must be in the same virtual network.

For securing the default AKS inference cluster inside your virtual network, specify three IP addresses:

- CIDR notation specifies an **AKS address range**. It must not overlap with any other subnet range.
- An **AKS DNS service IP address** is assigned for the DNS service within AKS. It must be within the AKS address range.
- A **Docker bridge address** is assigned to the Docker bridge, which executes your scoring script as a container. This address must not be within your subnet IP or AKS address range.

You can configure AKS to use an internal and private load balancer with a private AKS cluster. Only private IPs are allowed for this scenario, and you can use a Python SDK or Azure command-line extension but not Azure Machine Learning studio for this task. When using a private load balancer, you have to grant the Network Contributor role to the AKS cluster resource group that contains the virtual network.

Next steps

- Reference how to secure an Azure Machine Learning inferencing environment with virtual networks to see an IP address range and the steps to perform inference in a virtual network.
- See the Network Contributor role section to learn how to configure a private load balancer and set up a Network Contributor role.

Machine learning inference during deployment

5/18/2021 • 7 minutes to read • Edit Online

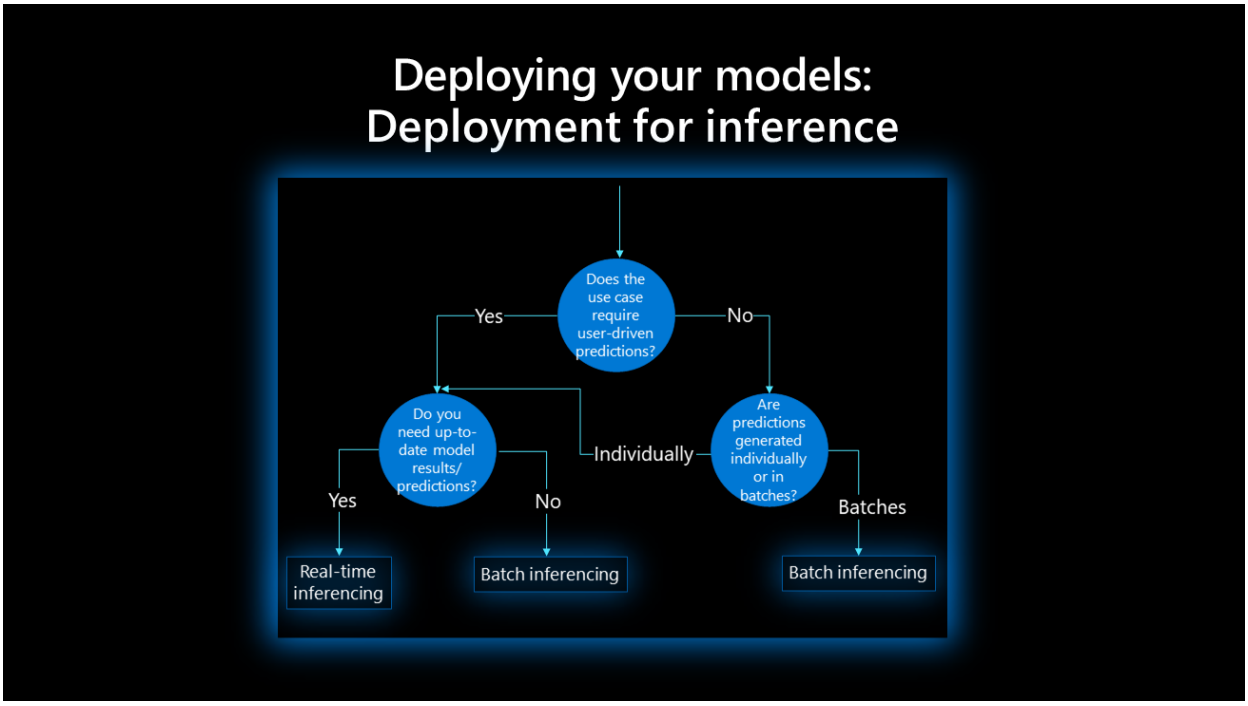
When deploying your AI model during production, you need to consider how it will make predictions. The two main processes for AI models are:

- **Batch inference:** An asynchronous process that bases its predictions on a batch of observations. The predictions are stored as files or in a database for end users or business applications.
- **Real-time (or interactive) inference:** Frees the model to make predictions at any time and trigger an immediate response. This pattern can be used to analyze streaming and interactive application data.

Consider the following questions to evaluate your model, compare the two processes, and select the one that suits your model:

- How often should predictions be generated?
- How soon are the results needed?
- Should predictions be generated individually, in small batches, or in large batches?
- Is latency to be expected from the model?
- How much compute power is needed to execute the model?
- Are there operational implications and costs to maintain the model?

The following decision tree can help you to determine which deployment model best fits your use case:



Batch inference

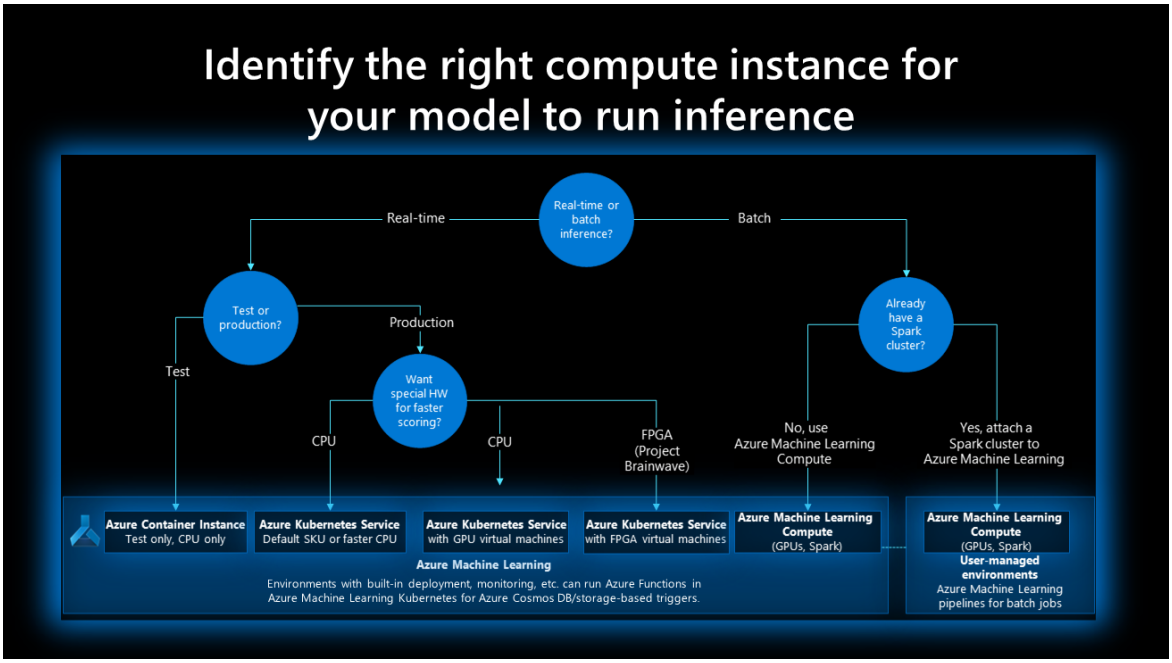
Batch inference, sometimes called offline inference, is a simpler inference process that helps models to run in timed intervals and business applications to store predictions.

Consider the following best practices for batch inference:

- **Trigger batch scoring:** Use Azure Machine Learning pipelines and the `ParallelRunStep` feature in Azure Machine Learning to set up a schedule or event-based automation. Navigate to the AI show to

perform batch inference using Azure Machine Learning `ParallelRunStep` and learn more about the process.

- **Compute options for batch inference:** Since batch inference processes don't run continuously, it's recommended to automatically start, stop, and scale reusable clusters that can handle a range of workloads. Different models require different environments, and your solution needs to be able to deploy a specific environment and remove it when inference is over for the compute to be available for the next model. See the following decision tree to identify the right compute instance for your model:



- **Implement batch inference:** Azure supports multiple features for batch inference. One feature is `ParallelRunStep` in Azure Machine Learning, which allows customers to gain insights from terabytes of structured or unstructured data stored in Azure. `ParallelRunStep` provides out-of-the-box parallelism and works within Azure Machine Learning pipelines.

- **Batch inference challenges:** While batch inference is a simpler way to use and deploy your model in production, it does present select challenges:

  - Depending on the frequency at which inference runs, the data produced could be irrelevant by the time it's accessed.

  - A variation of the cold-start problem; results might not be available for new data. For example, if a new user creates and account and starts shopping with a retail recommendation system, product recommendations won't be available until after the next batch inference run. If this is an obstacle for your use case, consider real-time inference.

  - Deploying to many regions and high availability aren't critical concerns in a batch inference scenario. The model doesn't need to be deployed regionally, and the data store might need to be deployed with a high-availability strategy in many locations. This will normally follow the application HA design and strategy.

Real-time inference

Real-time, or interactive, inference is architecture where model inference can be triggered at any time, and an immediate response is expected. This pattern can be used to analyze streaming data, interactive application data, and more. This mode allows you to take advantage of your machine learning model in real time and resolves the cold-start problem outlined above in batch inference.

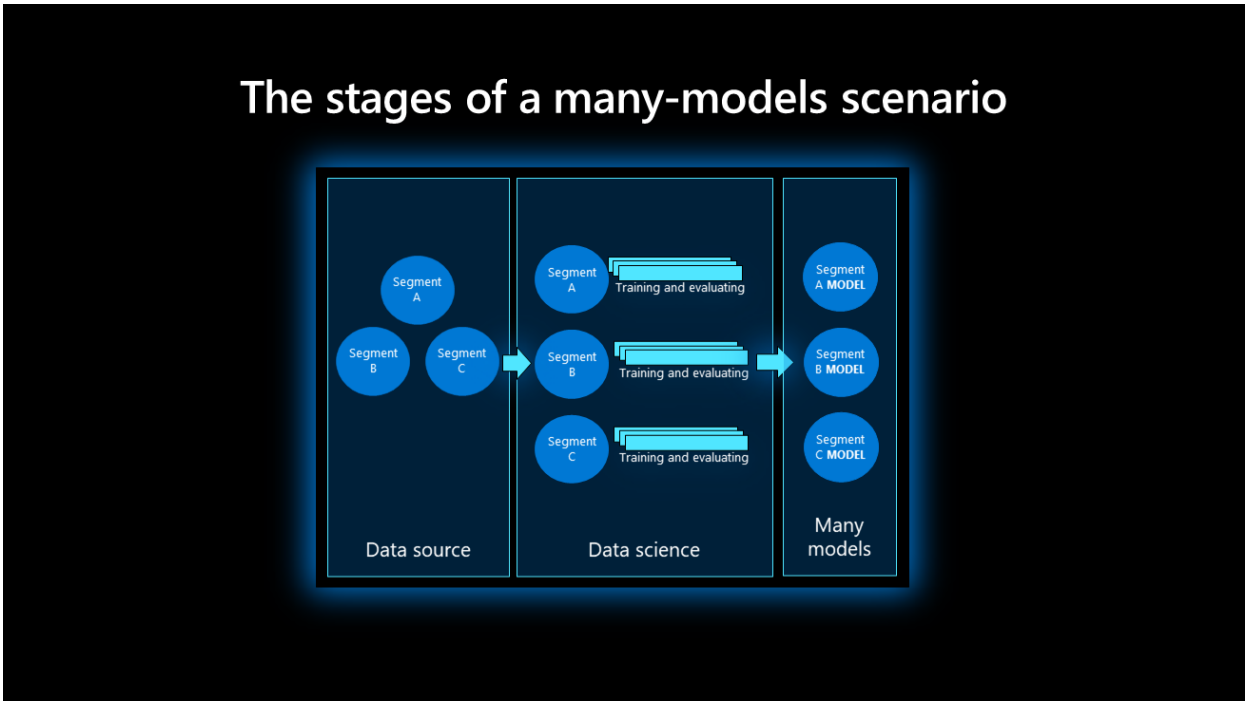
The following considerations and best practices are available if real-time inference is right for your model:

- **The challenges of real-time inference:** Latency and performance requirements make real-time inference architecture more complex for your model. A system might need to respond in 100 milliseconds or less, during which it needs to retrieve the data, perform inference, validate and store the model results, run any required business logic, and return the results to the system or application.
- **Compute options for real-time inference:** The best way to implement real-time inference is to deploy the model in a container form to Docker or AKS cluster and expose it as a web-service with REST API. This way, the model executes in its own isolated environment and can be managed like any other web service there. Docker/AKS capabilities can then be used for management, monitoring, scaling, and more. The model can be deployed on-premises, in the cloud, or on the edge. The preceding compute decision outlines real-time inference.
- **Multiregional deployment and high availability:** Regional deployment and high availability architectures need to be considered in real-time inference scenarios, as latency and the model's performance will be critical to resolve. To reduce latency in multiregional deployments, it's recommended to locate the model as close as possible to the consumption point. The model and supporting infrastructure should follow the business' high availability and DR principles and strategy.

Many-models scenario

A singular model might not be able to capture the complex nature of real-world problems, such as predicting sales for a supermarket where demographics, brand, SKUs, and other features could cause customer behavior to vary significantly. Regions could cause developing predictive maintenance for smart meters to also vary significantly. Having many models for these scenarios to capture regional data or store-level relationships could produce higher accuracy than a single model. This approach assumes that enough data is available for this level of granularity.

At a high level, a many-models scenario occurs in three stages: data source, data science, and many models.



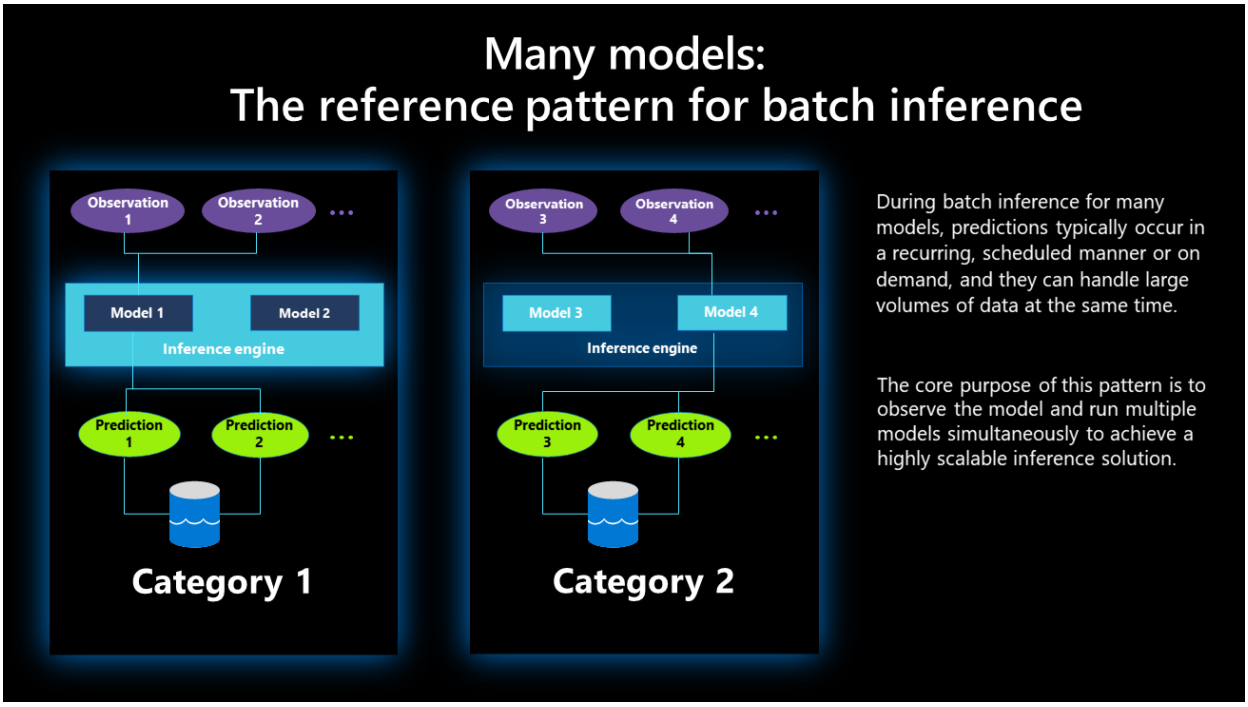
Data source: It's important to segment data without too many cardinalities in the data source stage. The product ID or barcode shouldn't be factored into the main partition, as this will produce too many segments and could inhibit meaningful models. The brand, SKU, or locality could be more fitting features. It's also important to homogenize the data by removing anomalies that would skew data distribution.

Data science: Several experiments run parallel to each data partition in the data science stage. This is a typically iterative process where models from the experiments are evaluated to determine the best one.

Many models: The best models for each segment or category are registered in the model registry. Assign meaningful names to the models, which will make them more discoverable for inference. Use tagging where necessary to group the model into specific categories.

Batch inference for many models

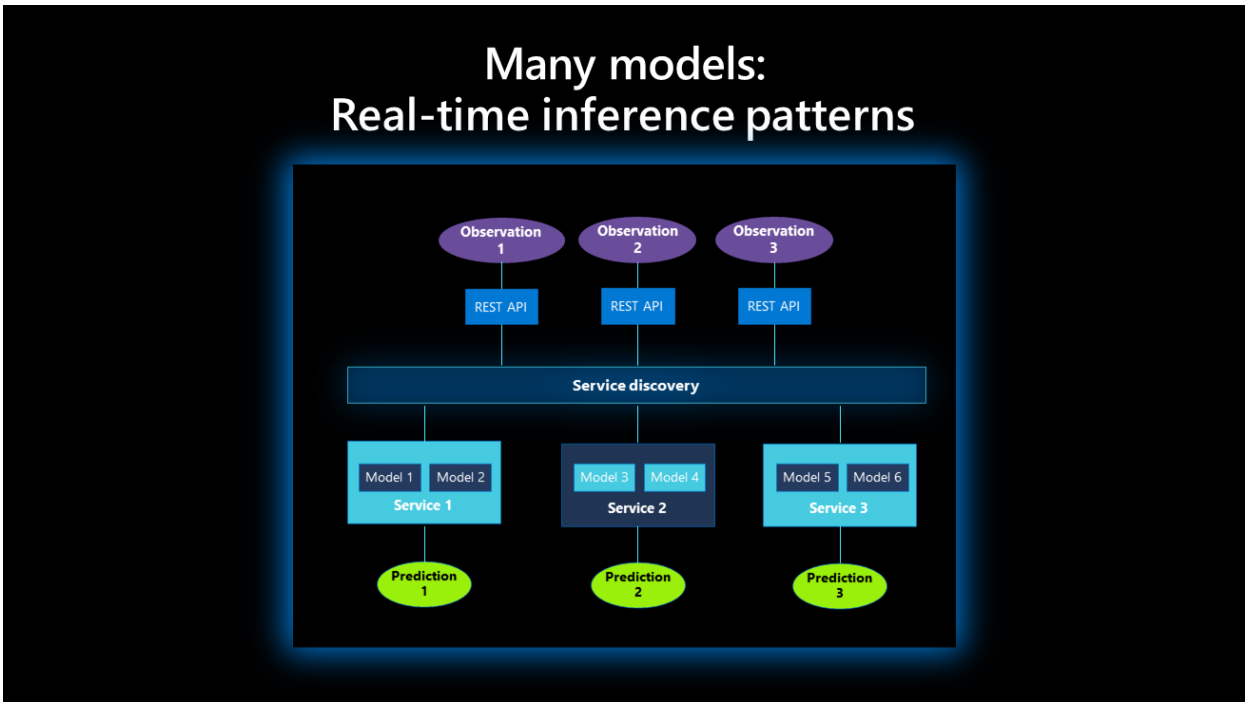
During batch inference for many models, predictions are typically scheduled, recurring, and they can handle large volumes of data running at the same time. Unlike in a single-model scenario, many models inference at the same time, and it's important to select the correct ones. The following diagram shows the reference pattern for many-models batch inference:



The core purpose of this pattern is to observe the model and run multiple models simultaneously to achieve a highly scalable inference solution that can handle large data volumes. To achieve hierarchical model inference, many models can be split into categories. Each category can have its own inference storage, like an Azure data lake. When implementing this pattern, one needs to balance scaling the models horizontally and vertically, as this would impact the cost and performance. Running too many model instances might increase performance but impact the cost. Too few instances with high spec nodes might be more cost effective but could cause issues with scaling.

Real-time inference for many models

Real-time many-models inference requires low latency and on-demand requests, typically via a REST endpoint. This is useful when external applications or services require a standard interface to interact with the model, typically via a REST interface with a JSON payload.



The core purpose of this pattern is to use the discovery service to identify a list of services and their metadata. This can be implemented as an Azure function and enables clients to obtain relevant service details of service, that can be invoked with a secure REST URI. A JSON payload be sent to the service, which would summon the relevant model and provide a JSON response back to the client.

Each service is stateless microservice that can handle multiple requests simultaneously and is limited to the physical virtual machine resource. The service can deploy multiple models if multiple groups are selected; homogeneous groupings like the category, SKU, and more are recommended for this. The mapping between the service request and model selected for a given service needs to be baked into the inference logic, typically via the score script. If the size of models is relatively small (a few megabytes), it's recommended to load them in memory for performance reasons; otherwise, each model can be loaded dynamically per request.

Next steps

Explore the following resources to learn more about inference in Azure Machine Learning:

- Build an Azure Machine Learning pipeline for batch scoring
- Run batch prediction using Azure Machine Learning designer
- Batch inference in Azure Machine Learning

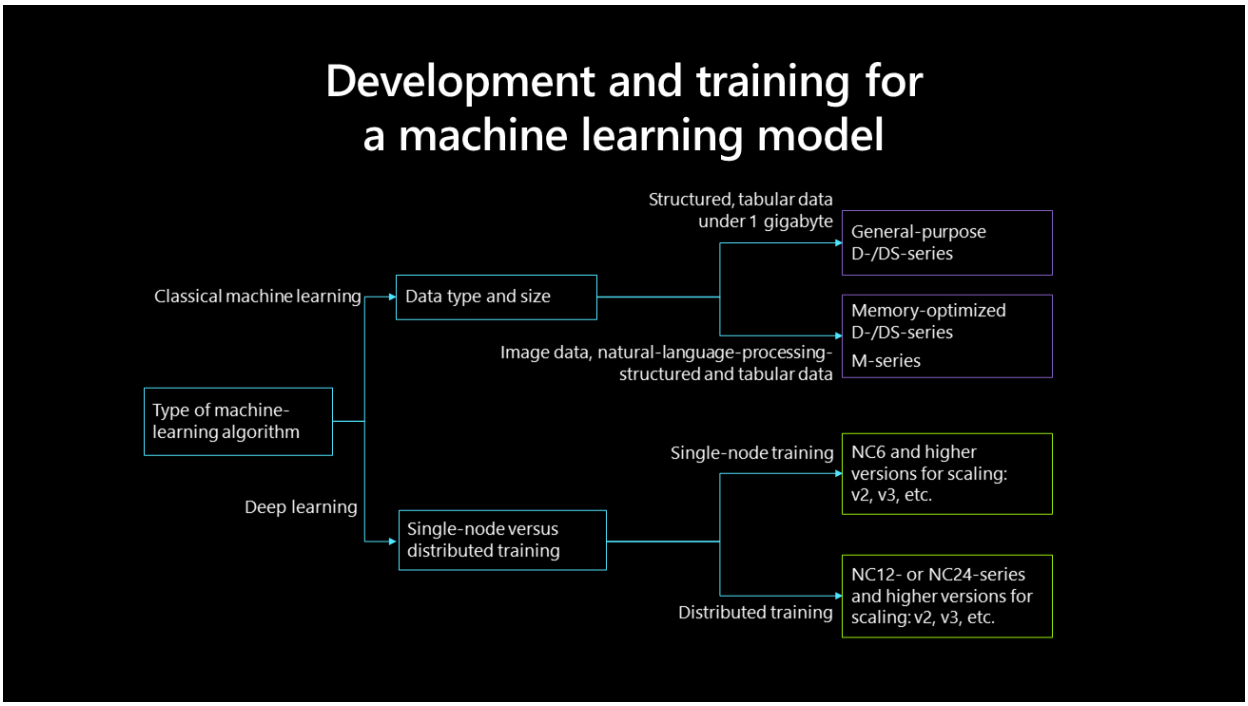
Determine development, training, and compute instances for your machine learning model

3/6/2021 • 2 minutes to read • Edit Online

When determining development, training, and compute instances for your machine learning model, consider the algorithm that you're using, the data type, data sizes, and if you'll perform distributed training.

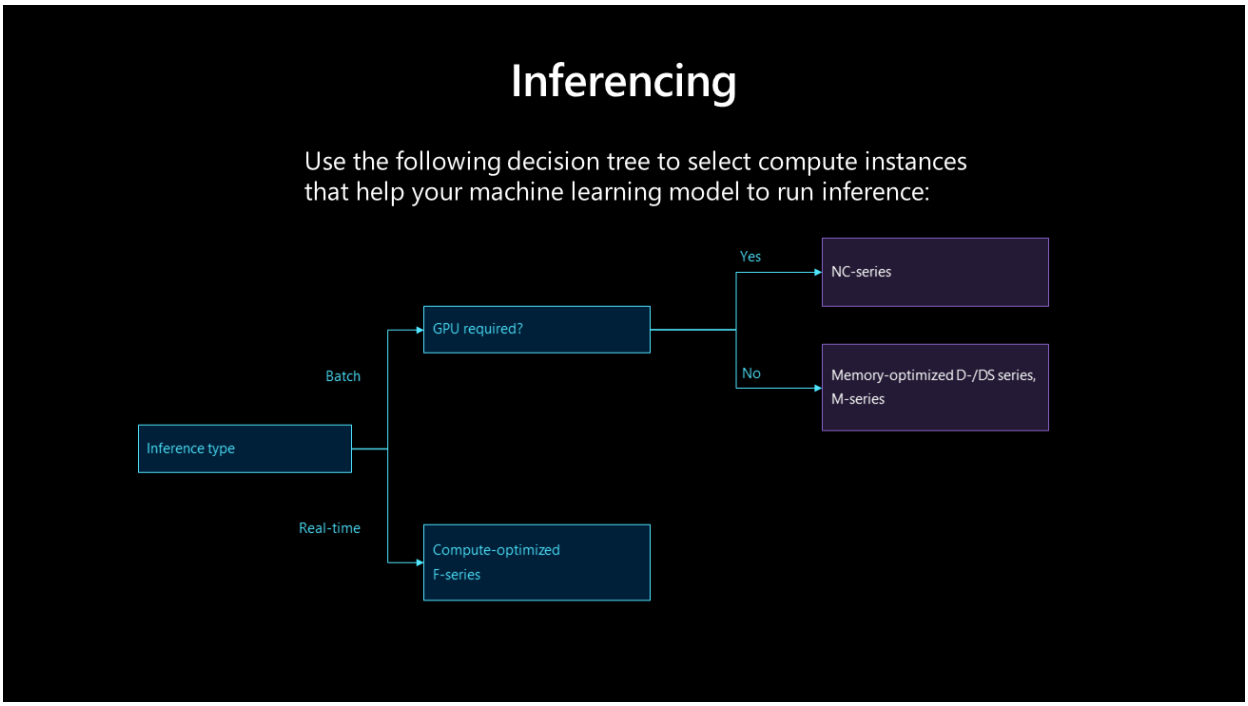
Development and training for your machine learning model

Reference the following diagram to understand development and training for your machine learning model:



Compute instances for your machine learning model

The following diagram can help you to choose compute instances that help your machine learning model to run inference:



Next steps

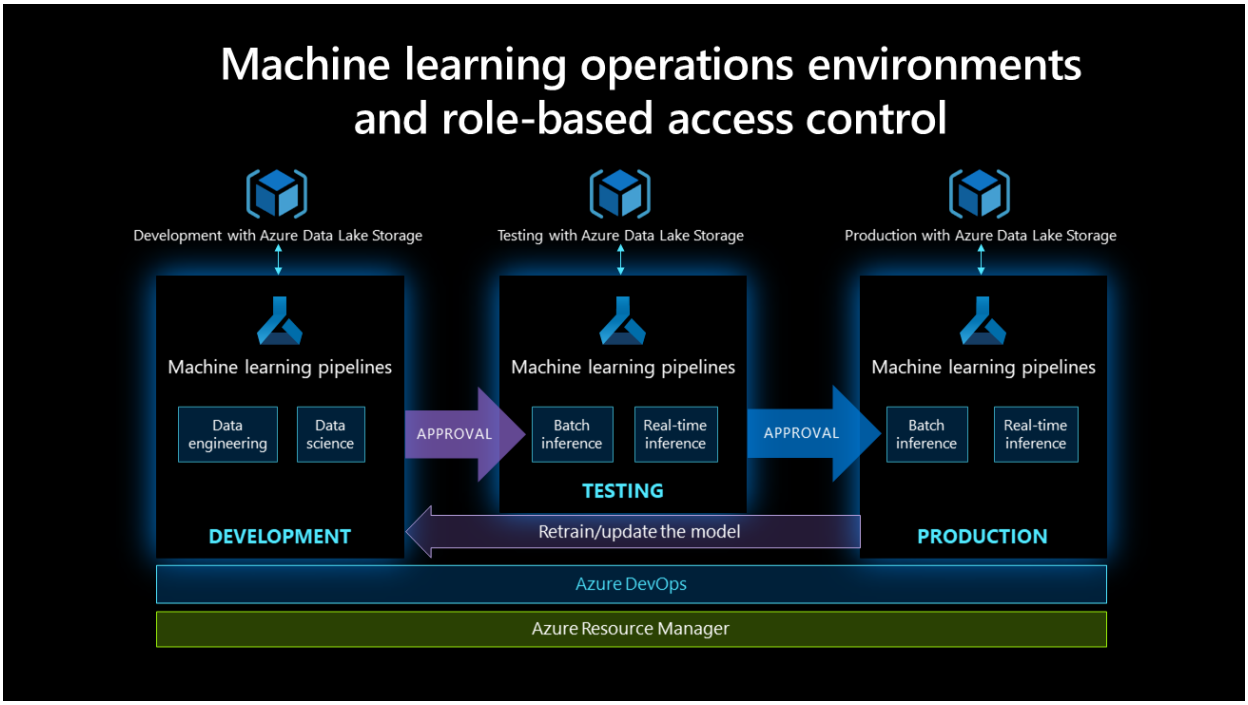
- Create and manage a compute instance in your Azure Machine Learning workspace to understand the process in more detail.
- Read about compute targets in Azure Machine Learning to learn how these environments can help you to train and deploy your model during the development lifecycle.

How to set up machine learning workspaces

3/6/2021 • 2 minutes to read • Edit Online

Machine learning environments and role-based access control

Development, testing, and production environments support machine learning operations processes.

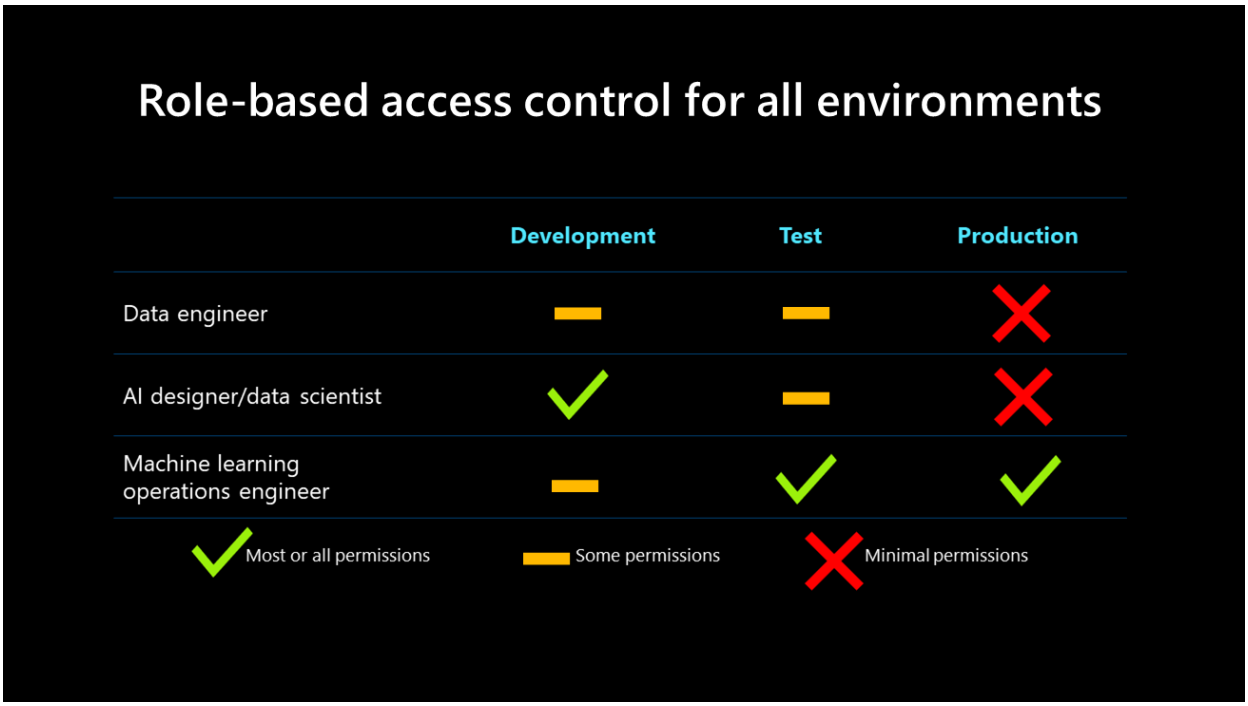


In a development environment: Machine learning pipelines should support the data science and engineering activities carried out by data scientists and data engineers. They should have full permissions related to running experiments, such as provisioning training clusters or building models. However, they should not have permission for activities like deleting or creating workspaces, or adding or removing workspace users.

In a test environment: Various tests are performed on the model's environment, and either champion/challenger or A/B testing should be used for the model. The test environment should mimic the deployment environment, and should run tests such as load testing, model response time, and others. Data scientists and engineers have limited access to this environment, primarily read-only access and some configuration rights. A DevOps engineer hand has full access to the environment. Automate as many tests as possible. Once all the tests are completed, stakeholder approval is required for deployment in the production environment.

In a production environment: A model is deployed during batch or real-time inference. A production environment is typically read-only; however, a DevOps engineer has full access to this environment and is responsible for continually supporting and maintaining it. Data scientists and data engineers have limited access to the environment, and it's read-only.

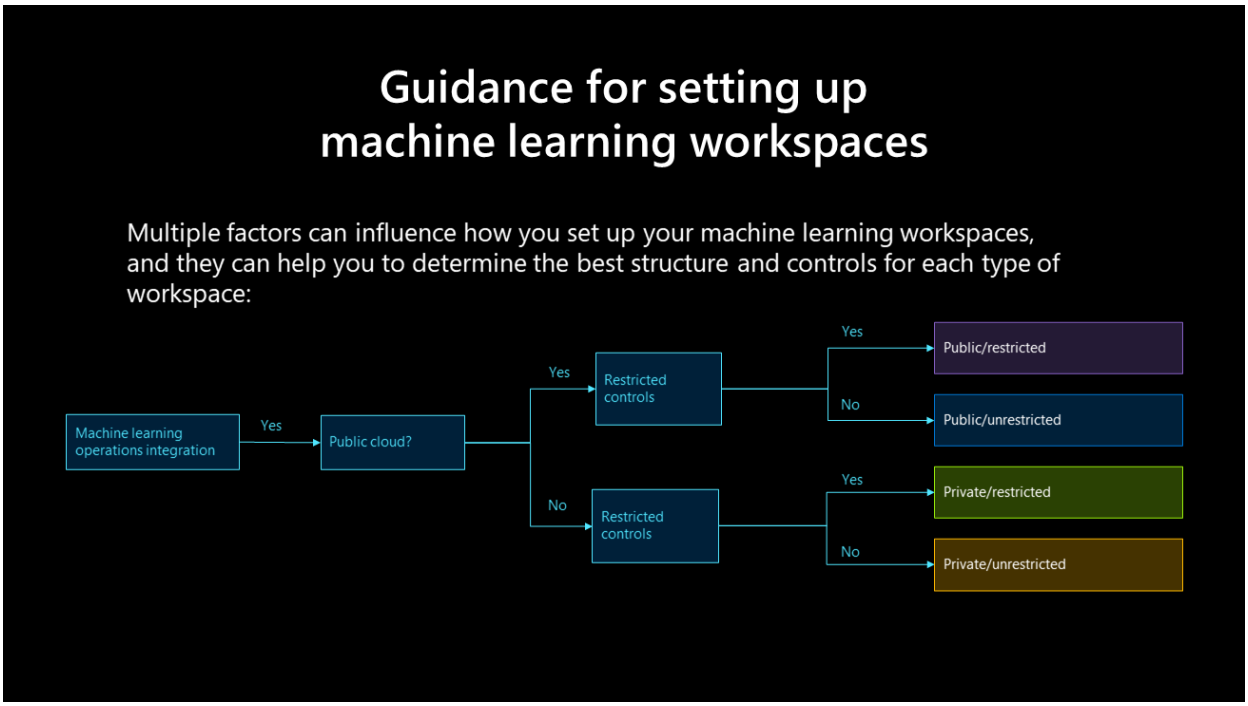
Role-based access control for all environments is shown in the following diagram:



This table shows that the data engineer and data scientist's access levels decrease within higher environments while the DevOps engineer's access increases. This is because a machine learning operations engineer builds the pipeline, glues things together, and deploys models in production. This level of granularity is recommended for each role.

Factors that influence machine learning workspaces

Multiple factors can influence how you set up your machine learning workspaces, and they can help you to determine the best structure and controls for each type of workspace:



- **Public, restricted:**
  - Development, test, and production workspace
  - Custom role: Data scientist
  - Git integration for version control and continuous integration/continuous development (CI/CD)
- **Public, unrestricted:**

    - Development, test, and production workspace
    - Role: Contributor
    - Git integration for version control and CI/CD
- **Private, restricted:**
    - Development, test, and production workspace
    - Private Link enabled
    - Custom role: Data scientist
    - Git integration for version control and CI/CD
- **Private, unrestricted:**
    - Development, test, and production workspace
    - Private Link enabled
    - Role: Contributor
    - Git integration for version control and CI/CD
- **All workspaces:**
    - One Azure Machine Learning studio workspace per project
    - One compute instance per data scientist
    - One compute cluster per virtual machine size shared with data scientists for development
    - One compute cluster per production pipeline
    - Setting the compute cluster minimum node size to 0 to reduce costs
    - Instructing users to shut down compute instances manually after use
    - A 'workspace admin' custom role with access to create compute instances and clusters
    - A 'data scientist' custom role requiring all infrastructure to be set up by another user before the data scientist can begin work

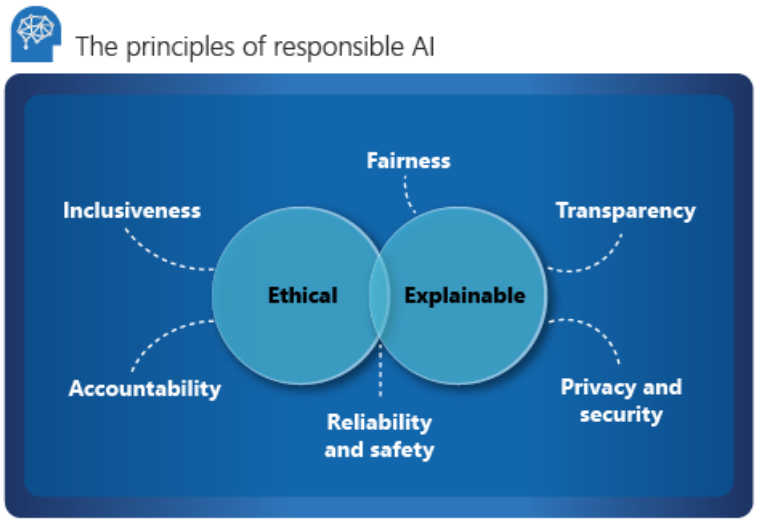
Next steps

- Learn more about how to create and manage a workspace in Azure Machine Learning.
- Use the Python SDK to create a workspace in your development environment.
- Use the Azure portal and Jupyter notebooks during the development lifecycle to set up your Azure Machine Learning development environment and train your model.

Responsible and trusted AI

5/18/2021 • 8 minutes to read • Edit Online

Microsoft outlines six key principles for responsible AI: accountability, inclusiveness, reliability and safety, fairness, transparency, and privacy and security. These principles are essential to creating responsible and trustworthy AI as it moves into more mainstream products and services. They are guided by two perspectives: ethical and explainable.



Ethical

From an ethical perspective, AI should be fair and inclusive in its assertions, be accountable for its decisions, and not discriminate or hinder different races, disabilities, or backgrounds.

Microsoft established an ethical committee for AI, ethics, and effects in engineering and research Aether, in 2017. The core responsibility of the committee is to advise on responsible issues, technologies, processes, and best practices. Learn more about Aether at this Microsoft Learn module.

Accountability

Accountability is an essential pillar of responsible AI. The people who design and deploy the AI system need to be accountable for its actions and decisions, especially as we progress toward more autonomous systems. Organizations should consider establishing an internal review body that provides oversight, insights, and guidance about developing and deploying AI systems. While this guidance might vary depending on the company and region, it should reflect an organization's AI journey.

Inclusiveness

Inclusiveness mandates that AI should consider all human races and experiences, and inclusive design practices can help developers to understand and address potential barriers that could unintentionally exclude people. Where possible, speech-to-text, text-to-speech, and visual recognition technology should be used to empower people with hearing, visual, and other impairments.

Reliability and safety

AI systems need to be reliable and safe in order to be trusted. It's important for a system to perform as it was originally designed and for it to respond safely to new situations. Its inherent resilience should resist intended or unintended manipulation. Rigorous testing and validation should be established for operating conditions to ensure that the system responds safely to edge cases, and A/B testing and champion/challenger methods should be integrated into the evaluation process.

An AI system's performance can degrade over time, so a robust monitoring and model tracking process needs to be established to reactively and proactively measure the model's performance and retrain it, as necessary, to modernize it.

Explainability

Explainability helps data scientists, auditors, and business decision makers to ensure that AI systems can reasonably justify their decisions and how they reach their conclusions. This also ensures compliance with company policies, industry standards, and government regulations. A data scientist should be able to explain to the stakeholder how they achieved certain levels of accuracy and what influenced this outcome. Likewise, in order to comply with the company's policies, an auditor needs a tool that validates the model, and a business decision maker needs to be able to provide a transparent model in order to gain trust.

Explainability tools

Microsoft has developed InterpretML, an open-source toolkit that helps to achieve model explainability and supports glass-box and black-box models.

- Glass-box models are interpretable because of their structure. For these models, use Explainable Boosting Machine, which is the state of the algorithm based on a decision tree or linear models, provides lossless explanations, and is editable by domain experts.
- Black-box models are more challenging to interpret because of a complex internal structure, the neural network. Explainers like LIME or SHapley Additive exPlanations (SHAP) interpret these models by analyzing the relationship between the input and output.
- Fairlearn is an Azure Machine Learning integration and an open-source toolkit for the SDK and the AutoML graphical user interface. Use explainers to understand what mainly influences the model and domain experts to validate these influences.

Explore model interpretability in Azure Machine Learning to learn more about explainability.

Fairness

Fairness is a core ethical principle that all humans aim to understand and apply. This principle is even more important when AI systems are being developed. Key checks and balances need to make sure that the system's decisions don't discriminate or run a gender, race, sexual orientation, or religion bias toward a group or individual.

- Microsoft provides an AI fairness checklist that offers guidance and solutions for AI systems. These solutions are loosely categorized into five stages: envision, prototype, build, launch, and evolve. Each stage lists recommended due diligence activities that help to minimize the impact of unfairness in the system.
- Fairlearn integrates with Azure Machine Learning and supports data scientists and developers to assess and improve the fairness of their AI systems. The toolbox provides various unfairness mitigation algorithms and an interactive dashboard that visualizes the fairness of the model. Use the toolkit and closely assess the fairness of the model while it's being built; this should be an integral part of the data science process.

Learn how to mitigate fairness in machine learning models.

Transparency

Achieving transparency helps the team to understand the data and algorithms used to train the model, what transformation logic was applied to the data, the final model generated, and its associated assets. This information offers insights about how the model was created, which allows it to be reproduced in a transparent way. Snapshots within Azure Machine Learning workspaces support transparency by recording or retraining all training-related assets and metrics involved in the experiment.

Privacy and security

A data holder is obligated to protect the data in an AI system, and privacy and security are an integral part of this system. Personal needs to be secured, and it should be accessed in a way that doesn't compromise an individual's privacy. Azure differential privacy protects and preserves privacy by randomizing data and adding noise to conceal personal information from data scientists.

Human AI guidelines

Human AI design guidelines consist of 18 principles that occur over four periods: initially, during interaction, when wrong, and over time. These principles are designed to produce a more inclusive and human-centric AI system.

Initially

- **Clarify what the system can do.** If the AI system uses or generates metrics, it's important to show them all and how they're tracked.
- **Clarify how well the system can do what it can do.** Help users to understand that AI will not be completely accurate, and set expectations for when the AI system might make mistakes.

During interaction

- **Show contextually relevant information.** Provide visual information related to the user's current context and environment, such as nearby hotels and return details close to the target destination and date.
- **Mitigate social biases.** Make sure that the language and behavior don't introduce unintended stereotypes or biases. For example, an autocomplete feature needs to acknowledge both genders.

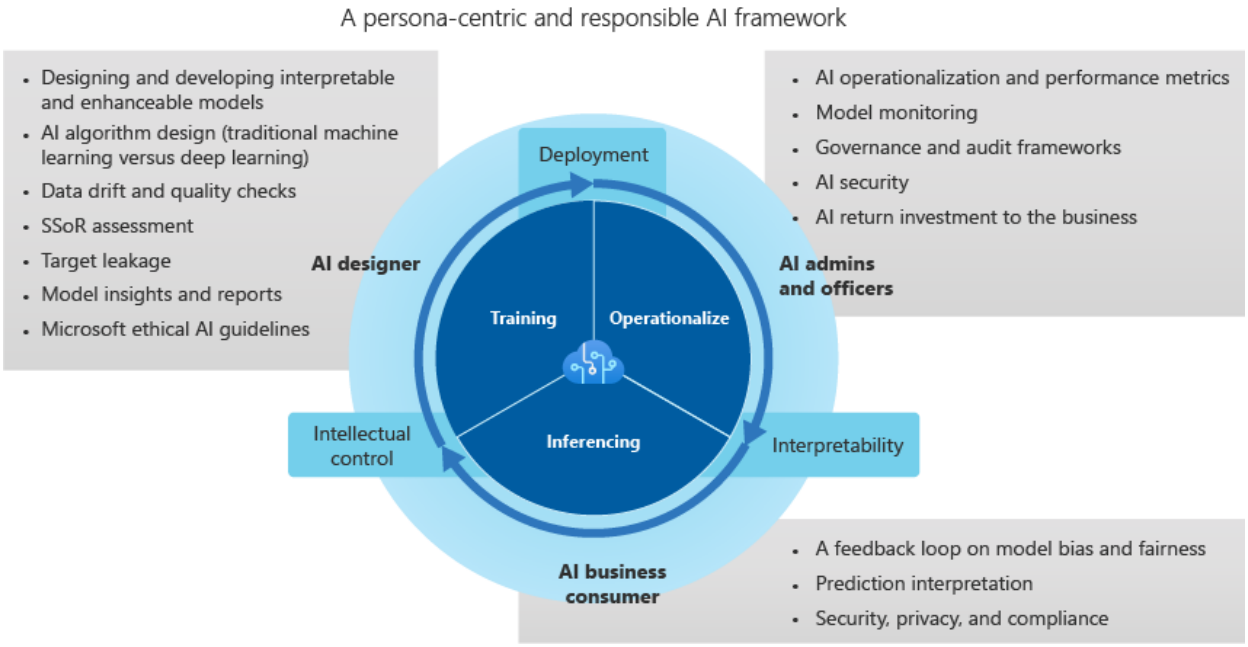
When wrong

- **Support efficient dismissal.** Provide an easy mechanism to ignore or dismiss undesirable features/services.
- **Support efficient correction.** Provide an intuitive way of making it easier to edit, refine, or recover.
- **Make clear why the system did what it did.** Optimize explainable AI to offer insights about the AI system's assertions.

Over time

- **Remember recent interactions.** Retain a history of interactions for future reference.
- **Learn from user behavior.** Personalize the interaction based on the user's behavior.
- **Update and adapt cautiously.** Limit disruptive changes, and update based on the user's profile.
- **Encourage granular feedback.** Gather user feedback from their interactions with the AI system.

A persona-centric, trusted AI framework



AI designer

The AI designer builds the model and is responsible for:

- Data drift and quality checks. They detect outliers and perform data quality checks to identify missing values, standardize distribution, scrutinize data, and produce use case and project reports.
- Assessing data in the system's source to identify potential bias.
- Designing AI algorithms to minimize data biases, such as discovering how binning, grouping, and normalization (especially in traditional machine learning models like tree-based ones) can eliminate minority groups from data. Categorical AI design reiterates data biases by grouping social, racial, and gender classes in industry verticals that rely on protected health information (PHI) and personally identifiable information (PII).
- Optimizing monitoring and alerts to identify target leakage and strengthen the model's development.
- Establishing best practices for reporting and insights that offer a granular understanding of the model and avoiding black-box approaches that use feature or vector importance, UMAP clustering, Friedman's H-statistic, feature effects, and others. Identification metrics help to define predictive influence, relationships, and dependencies between correlations in complex and modern datasets.

AI administrators and officers

The AI administrator and officers oversee AI, governance, and audit framework operations and performance metrics, plus how AI security is implemented and the business' return on investment.

- Monitoring a tracking dashboard that assists model monitoring, combines model metrics for production models, and focuses on accuracy, model degradation, data drift, deviation, and changes in speed/error of inference.
- Implementing flexible deployment and redeployment (preferably, REST API) allows models to be implemented into open, agnostic architecture, which integrates the model with business processes and generates value for feedback loops.
- Working toward building model governance and access sets boundaries and mitigates negative business and operational impact. Role-based access control standards determine security controls, which preserve restricted production environments and the IP.
- Using AI audit and compliance frameworks to track how models develop and change to uphold industry-specific standards. Interpretable and responsible AI is founded on explainability measures, concise

features, model visualizations, and industry-vertical language.

AI business consumer

AI business consumers (business experts) close the feedback loop and provide input for the AI designer. Predictive decision-making and potential bias implications like fairness and ethical measures, privacy and compliance, and business efficiency help to evaluate AI systems.

- Feedback loops belong to a business' ecosystem. Data showing a model's bias, errors, prediction speed, and fairness establish trust and balance between the AI designer, administrator, and officers. Human-centric assessment should gradually improve AI over time, and minimizing AI learning from multidimensional, complex data (LO-shot learning) can help to prevent biased learning.
- Using interpretability design and tools hold AI systems accountable for potential biases. Model bias and fairness issues should be flagged and fed to an alerting and anomaly detection system that learns from this behavior and automatically addresses biases.
- Each predictive value should be broken down into individual features or vectors by importance or impact and deliver thorough prediction explanations that can be exported into a business report for audit and compliance reviews, customer transparency, and business readiness.
- Due to increasing global security and privacy risks, best practices for resolving data violations during inference require complying with regulations in individual industry verticals; for example, alerts about noncompliance with PHI and PII, violation of national security laws, and more.

Next steps

Explore human AI guidelines to learn more about responsible AI.

What are AI applications?

8/11/2021 • 4 minutes to read • Edit Online

AI applications are things like speech recognition APIs, computer vision APIs, decision logic APIs, and other kinds of intelligent systems that mimic human reason. These are essential functions of many software products in the market today, since AI applications can communicate with end users in a more natural manner and make for a better user experience. Azure can help you and your team save time by being able to roll out AI applications anywhere.

In Azure, you can build intelligent applications faster by using the tools and technologies of your choice and built-in AI.

- **Easily build and deploy anywhere:** Use your team's existing skill set and the tools you know to build intelligent applications and deploy them without a change in code. You can build once and then deploy in the cloud, on-premises, and to edge devices. You can be confident of global distribution to more datacenters than with any other provider.
- **Create an impact by using an open platform:** Choose your favorite technologies, which can be open source. Azure supports a range of deployment options, popular stacks and languages, and a comprehensive set of data engines. Capitalize on this flexibility, plus the performance, scale, and security delivered by Microsoft technologies to build applications for any scenario.
- **Develop applications with built-in intelligence:** Building intelligent applications using Azure is easy, because no other platform brings analytics and native AI to your data wherever it lives and in the languages you use. You can take advantage of a rich set of cognitive APIs to easily build new experiences into your applications for human-like intelligence.

What is Azure Cognitive Services?

Azure Cognitive Services can simplify how you integrate AI capabilities and breakthroughs into your applications with a few simple lines of code. It supports you to create applications that see, hear, speak, understand, and even start to reason between your business processes. Cognitive Services provides AI intelligence in a form that's easy to use and incorporate into your applications.

Cognitive Services is made up of APIs, SDKs, and services available to help developers build intelligent applications without having direct AI or data science skills or knowledge. Cognitive Services enables developers to easily add cognitive features into their applications. The catalog of services within Cognitive Services can be categorized into five main parts: vision, speech, language, web search, and decision.

Computer Vision APIs

SERVICE NAME	SERVICE DESCRIPTION
Computer Vision	Computer Vision APIs provides you with access to advanced algorithms for processing images and returning information.
Custom Vision	Custom Vision allows you to build custom image classifiers.
Face	The Face service provides access to advanced face algorithms that detect and recognize facial attributes.

SERVICE NAME	SERVICE DESCRIPTION
Form Recognizer (preview)	Form Recognizer identifies and extracts key-value pairs and table data from form documents. It then outputs structured data, which includes the relationships, in the original file.
Video Indexer	Video Indexer enables you to extract insights from your videos.

Speech recognition APIs

SERVICE NAME	SERVICE DESCRIPTION
Speech	Speech service adds speech-enabled features to applications.
Speaker Recognition (preview)	The Speaker Recognition API provides algorithms for Speaker Identification and verification.
Bing Speech (retired)	You can migrate existing Bing Speech API applications to the Speech service.
Translator Speech (retired)	You can migrate existing Bing Translator Speech API applications to the Speech service.

Language APIs

SERVICE NAME	SERVICE DESCRIPTION
Language Understanding (LUIS)	The Language Understanding service (LUIS) allows your application to understand what a person wants in their own words.
QnA Maker	QnA Maker allows you to build a question-and-answer service from your semistructured content.
Text Analytics	Text Analytics provides natural language processing over raw text for sentiment analysis, key phrase extraction, and language detection.
Translator	Translator provides machine-based text translation in near real time.

Decision logic APIs

SERVICE NAME	SERVICE DESCRIPTION
Anomaly Detector (preview)	Anomaly Detector allows you to monitor and detect abnormalities in your time series data.
Content Moderator	Content Moderator provides monitoring for possible offensive, undesirable, and risky content.
Personalizer	Personalizer allows you to learn from users' real-time behavior in order to choose the most tailored experience for them.

Supported cultural languages

Cognitive Services supports a wide range of cultural languages at the service level. You can find the language availability for each API in the supported languages list.

Secure resources

Cognitive Services provides a layered security model, which includes authentication via Azure Active Directory credentials, a valid resource key, and Azure Virtual Network.

Container support

Cognitive Services provides containers for deployment in the cloud or on-premises. Learn more about Cognitive Services containers.

Certifications and compliance

Cognitive Services has been awarded certifications such as CSA STAR certification, FedRAMP Moderate, and HIPAA BAA.

You can download certifications for your own audits and security reviews.

To understand privacy and data management, go to the Microsoft Trust Center.

How are Cognitive Services and Azure Machine Learning similar?

Cognitive Services and Azure Machine Learning have the common goal of applying AI to enhance business operations. How each one provides this capability in the respective offerings is different. Generally, the audiences are different:

- Cognitive Services is for developers without machine learning experience.
- Machine learning is tailored for data scientists.

How is a cognitive service different from machine learning?

A cognitive service provides a trained model for you. This model brings data and an algorithm together and is available from a REST API or SDK. You can implement this service within minutes depending on your scenario. A cognitive service provides answers to general problems such as key phrases in text or item identification in images.

Machine learning is a process that generally requires a longer period of time to implement successfully. This time is spent on data collection, cleaning, transformation, algorithm selection, model training, and deployment to get to the same level of functionality provided by a cognitive service. With machine learning, it's possible to provide answers to highly specialized or specific problems. Machine learning problems require familiarity with the specific subject matter and data of the problem under consideration and expertise.

Next steps

- Learn more about Cognitive Services.
- Find best practices for AI architectures.

What are AI agents?

3/6/2021 • 6 minutes to read • Edit Online

AI agents are code or mechanisms which act to achieve predetermined goals. Examples of AI agents can be found in the code for things like chat bots, smart homes, and the programmatic trading software used in finance. The Azure Bot Service and Bot Framework are examples of platforms that can be used to build these AI agents and integrate them into larger software applications.

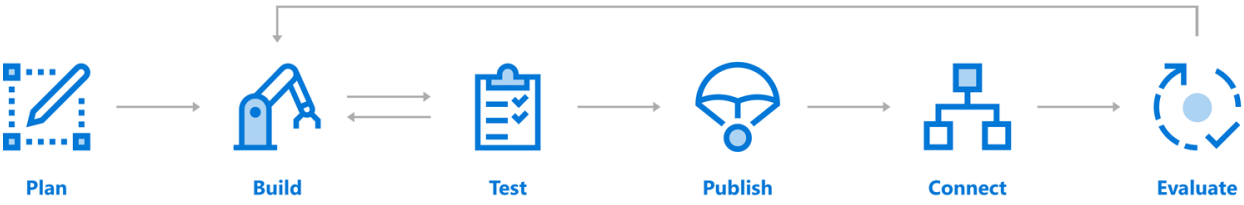
Users are engaging more and more with conversational interfaces, which can present a more natural experience where humans express their needs through natural language and quickly complete tasks. For many companies, conversational AI applications are becoming a competitive differentiator. Many organizations are strategically making bots available within the same messaging platforms in which their customers spend time.

Organizations around the world are transforming their businesses with conversational AI, which can promote more efficient and natural interactions with both their customers and their employees. Here are a few common use cases:

- Customer support
- Enterprise assistant
- Call center optimization
- In-car voice assistant

Build a bot

Azure Bot Service and Bot Framework offer an integrated set of tools and services to help create the AI agent you need. Choose your favorite development environment or command-line tools to create your bot. SDKs exist for C#, JavaScript, TypeScript, and Python. The SDK for Java is under development. We provide tools for various stages of bot development to help you design and build bots.



Plan

Having a thorough understanding of the goals, processes, and user needs is important to the process of creating a successful bot. Before you write code, review the bot design guidelines for best practices, and identify the needs for your bot. You can create a simple AI agent or include more sophisticated capabilities such as speech, natural language understanding, and question answering.

While you design your AI agent during the planning phase, consider these aspects:

- Define bot personas:
  - What should your bot look like?
    - What should it be named?
    - What's your bot's personality? Does it have a gender?
    - How should your bot handle difficult situations and questions?
- Design conversation flow:
  - What type of conversations can you expect for your use cases?

- Define an evaluation plan:
  - How would you measure success?
  - What measurements do you want to use to improve your service?

To learn more about how to design your bot, see Principles of bot design.

Build

Your bot is a web service that implements a conversational interface and communicates with the Bot Framework Service to send and receive messages and events. The Bot Framework Service is one of the components of Azure Bot Service and Bot Framework. You can create bots in any number of environments and languages. You can start your bot development in the Azure portal or use C#, JavaScript, or Python templates for local development. You also have access to a variety of samples that showcase many of the capabilities available through the SDK. These samples are great for developers who want a more feature-rich starting point.

As part of the Azure Bot Service and Bot Framework, we offer additional components you can use to extend the functionality of your bot. With the Azure Bot Service and Bot Framework, you can build a bot with confidence and speed.

FEATURE	DESCRIPTION	LINK
Add natural language processing	Enable your bot to understand natural language, understand spelling errors, use speech, and recognize the user's intent.	How to use LUIS
Answer questions	Add a knowledge base to answer questions users ask in a more natural, conversational way.	How to use QnA Maker
Manage multiple models	If you use more than one model, such as for LUIS and QnA Maker, intelligently determine when to use which one during your bot's conversation.	Dispatch tool
Add cards and buttons	Enhance the user experience with media other than text, such as graphics, menus, and cards.	How to add cards

> **NOTE**
> This table isn't a comprehensive list. For more information, see the Azure Bot Service documentation.

Test

Bots are complex applications with many different parts that work together. Like any other complex application, this complexity can lead to some interesting bugs or cause your bot to behave differently than expected. Before you publish your bot, test it. We provide several ways to test bots before they're released for use:

- Test your bot locally with the emulator. The Bot Framework Emulator is a stand-alone application that not only provides a chat interface but also debugging and interrogation tools to help you understand how and why your bot does what it does. The emulator can be run locally alongside your in-development bot application.
- Test your bot on the web. After your bot is configured through the Azure portal, it can also be reached through a web chat interface. The web chat interface is a great way to grant access to your bot to testers and other people who don't have direct access to the running code.

- Unit test your bot with the July update of the Bot Framework SDK.

Publish

When you're ready to make your bot available on the web, publish it to Azure or to your own web service or datacenter. Having an address on the public internet is the first step to bringing your bot to life on your site or inside chat channels.

Connect

Connect your bot to channels such as Facebook, Messenger, Kik, Skype, Slack, Microsoft Teams, Telegram, text/SMS, Twilio, Cortana, and Skype. Bot Framework does most of the work necessary to send and receive messages from all of these different platforms. Your bot application receives a unified, normalized stream of messages no matter number and type of channels to which it's connected. For information on how to add channels, see Channels.

Evaluate

Use the data collected in the Azure portal to identify opportunities to improve the capabilities and performance of your bot. You can get service-level and instrumentation data like traffic, latency, and integrations. Analytics also provide conversation-level reporting on user, message, and channel data. For more information, see How to gather analytics.

Patterns for common use cases

There are common patterns used for implementation of a conversational AI application:

- **Knowledge base:** A knowledge bot can be designed to provide information about virtually any subject. For example, one knowledge bot might answer questions about events such as "what bot events are there at this conference?" Or "when is the next reggae show?" Another bot might answer IT-related questions such as "how do I update my operating system?" Yet another bot might answer questions about contacts such as "who is john doe?" Or "what is jane doe's email address?"

  For information on the design elements for knowledge bots, see Design knowledge bots.

- **Hand off to a human:** No matter how much AI a bot possesses, there might still be times when it needs to hand off the conversation to a human being. In such cases, the bot should recognize when it needs to hand off and provide the user with a smooth transition.

  For information on the patterns to hand off, see Transition conversations from bot to human.

- **Embed a bot in an application:** Although bots most commonly exist outside of applications, they can also be integrated with applications. For example, you could embed a knowledge bot within an application to help users find information. You could also embed a bot within a help desk application to act as the first responder to incoming user requests. The bot could independently resolve simple issues and hand off more complex issues to a human agent.

  For information on the ways to integrate your bot within an application, see Embed a bot in an application.

- **Embed a bot in a website:** Like embedding bots in applications, bots can also be embedded within a website to enable multiple modes of communication across channels.

  For information on the ways to integrate your bot within a website, see Embed a bot in a website.

Next steps

- Review machine learning white papers and e-books about Azure Bot Service.
- Review AI + Machine Learning architectures.
- Building intelligent applications with cognitive APIs (e-book).

- FAQ chatbot architecture.

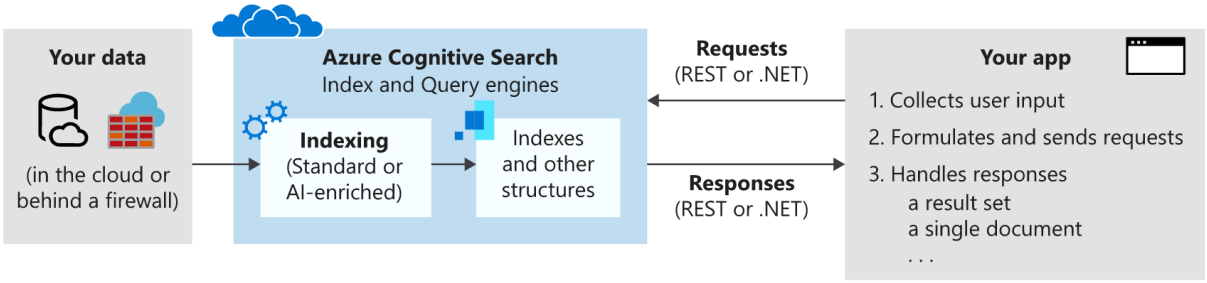
What is Azure Cognitive Search?

3/6/2021 • 4 minutes to read • Edit Online

Formerly known as Azure Search, Azure Cognitive Search is a managed cloud solution that gives developers APIs and tools for adding a rich search experience over private, heterogeneous content in web, mobile, and enterprise applications. Your code or a tool invokes data ingestion (indexing) to create and load an index. Optionally, you can add cognitive skills to apply AI processes during indexing. Using Azure Cognitive Services can add new information and structures that are useful for search and other scenarios.

On the other side of your service, your application code issues query requests and handles responses. The search experience is defined in your client by using functionality from Azure Cognitive Search, with query execution over a persisted index that you create, own, and store in your service.

Azure Cognitive Search is important functionality in applications. The ability to rapidly find relevant data is essential to the end-user experience and results. The Azure Cognitive Search engine uses AI functionality that helps applications work in a more human-like manner and make associations that go beyond mere keyword matching. Azure Cognitive Services can help your end users find what they need to know, faster.



Functionality is exposed through a simple REST API or .NET SDK that masks the inherent complexity of information retrieval. In addition to APIs, the Azure portal provides administration and content management support, with tools for prototyping and querying your indexes. Because the service runs in the cloud, infrastructure and availability are managed by Microsoft.

When to use Azure Cognitive Search

Azure Cognitive Search is well suited for the following application scenarios:

- Consolidation of heterogeneous content types into a private, single, searchable index. Queries are always over an index that you create and load with documents. The index always resides in the cloud on your Azure Cognitive Search instance. You can populate an index with streams of JSON documents from any source or platform. Alternatively, for content sourced on Azure, you can use an indexer to pull data into an index. Index definition and management/ownership is a key reason for using Azure Cognitive Search.
- Raw content includes large undifferentiated text, image files, or application files such as Microsoft Office content types in an Azure data source such as Azure Blob Storage or Azure Cosmos DB. You can apply cognitive skills during indexing to add structure or extract meaning from image and application files.
- Easy implementation of search-related features. Azure Cognitive Search APIs simplify query construction, faceted navigation, filters (including geo-spatial search), synonym mapping, type-ahead queries, and relevance tuning. Using built-in features, you can satisfy user expectations for a search experience similar to commercial web search engines.
- Indexing unstructured text or extracting text and information from image files. The AI enrichment feature of Azure Cognitive Search adds AI processing to an indexing pipeline. Some common use cases include OCR

over scanned documents, entity recognition and key phrase extraction over large documents, language detection and text translation, and sentiment analysis.

- Linguistic requirements satisfied by using the custom and language analyzers of Azure Cognitive Search. If you have non-English content, Azure Cognitive Search supports both Lucene analyzers and Microsoft's natural language processors. You can also configure analyzers to achieve specialized processing of raw content, such as filtering out diacritics.

Use Azure Cognitive Search

Step 1: Provision the service

You can provision an Azure Cognitive Search instance in the Azure portal or through the Azure Resource Manager REST API. You can choose either the free service shared with other subscribers or a paid tier that dedicates resources used only by your service. For paid tiers, you can scale a service in two dimensions:

- Add replicas to grow your capacity to handle heavy query loads.
- Add partitions to grow storage for more documents.

By handling document storage and query throughput separately, you can calibrate resourcing based on production requirements.

Step 2: Create an index

Before you can upload searchable content, you must first define an Azure Cognitive Search index. An index is like a database table that holds your data and can accept search queries. You define the index schema to map to reflect the structure of the documents you want to search, similar to fields in a database.

A schema can be created in the Azure portal or programmatically by using the .NET SDK or REST API.

Step 3: Load data

After you define an index, you're ready to upload content. You can use either a push or pull model.

The pull model retrieves data from external data sources. It's supported through indexers that streamline and automate aspects of data ingestion, such as connecting to, reading, and serializing data. Indexers are available for Azure Cosmos DB, Azure SQL Database, Azure Blob Storage, and SQL Server hosted in an Azure Virtual Machines instance. You can configure an indexer for on-demand or scheduled data refresh.

The push model is provided through the SDK or REST APIs used for sending updated documents to an index. You can push data from virtually any dataset by using the JSON format. For more information, see Add, update, or delete documents or how to use the .NET SDK for guidance on loading data.

Step 4: Search

After populating an index, you can issue search queries to your service endpoint by using simple HTTP requests with REST APIs or the .NET SDK. Step through creating your first search application to build and then extend a web page that collects user input and handles results. You can also use Postman for interactive REST calls or the built-in Search explorer in the Azure portal to query an existing index.

Next steps

- Learn more about Azure Cognitive Search.
- Browse more AI architectures.
- See an example knowledge mining solution in the article JFK Files example architecture and solution.

Innovation in the digital economy

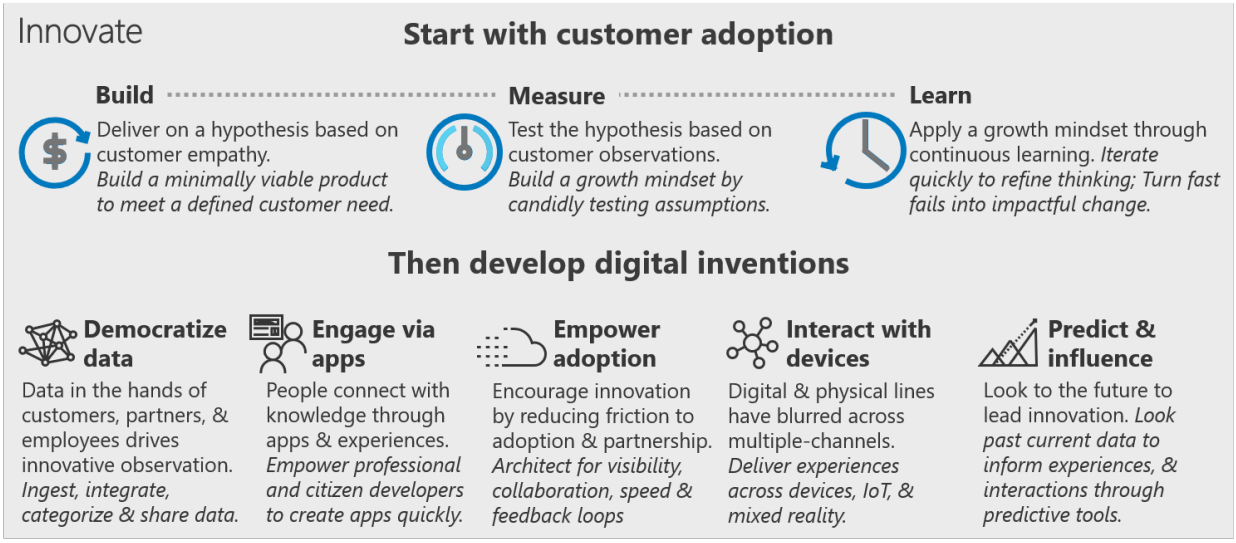
4/22/2021 • 4 minutes to read • Edit Online

The digital economy is an undeniable force in almost every industry. During the industrial revolution, gasoline, conveyor belts, and human ingenuity were key resources for promoting market innovation. Product quality, price, and logistics drove markets as companies sought to deliver better products to their customers more quickly. Today's digital economy shifts the way in which customers interact with corporations. The primary forms of capital and market differentiators have all shifted as a result. In the digital economy, customers are less concerned with logistics and more concerned with their overall experience of using a product. This shift arises from direct interaction with technology in our daily lives and from a realization of the value associated with those interactions.

In the Innovate methodology of the Cloud Adoption Framework, we'll focus on understanding customer needs and rapidly building innovations that shape how your customers interact with your products. We'll also illustrate an approach to delivering on the value of a minimum viable product (MVP). Finally, we'll map decisions common to innovation cycles to help you understand how the cloud can unlock innovation and create partnerships with your customers.

Innovate methodology

The simple methodology for cloud innovation within the Cloud Adoption Framework is illustrated in the following image. Subsequent articles in this section will show how to establish core innovation processes, approaches, and mechanisms for finding and driving innovation within your company.



This article series emphasizes the following aspects of this methodology:

- First, always start with customer adoption to generate feedback that builds customer partnerships through the build-measure-learn feedback loop.
- Second, examine approaches to developing digital inventions that prioritize adoption.

The following section describes what innovation is, the formula for innovation, and the commitments required for success with this approach.

Formula for innovation

Innovation is the development and application of ideas that improve the way things are done or what can be

achieved.

Successful innovation is not a big-bang transformational event or an elusive magical unicorn. Success in innovation is more of a balancing act, illustrated by a simple equation: **innovation = invention + adoption**.

Innovation happens at the intersection of invention and adoption. True innovation stems from slowly adjusting human experiences through new approaches, new processes, and new technologies. In this formula, invention means creating a new solution that meets a customer need. Conversely, adoption means applying the new solution to shape human behaviors and interactions. Finding the right balance between invention and adoption requires iteration, data-driven decision making, constant learning, and a growth mindset. It also requires technologies that can keep pace with the countless opportunities to learn in today's digital economy.

The cloud is often a great platform for invention or the technological aspects of innovation. Unfortunately, most great ideas fail during the hard work of adoption, rather than during the creative or invention processes. To ensure success, development teams should always start with adoption as the test for innovation. To use this methodology, the team should agree on these three commitments:

- Commitment to prioritize customers over technology
- Commitment to transparency
- Commitment to iteration

Cultural commitments

Adopting the Innovate methodology requires some cultural commitments to effectively use the metrics outlined in this article. Before you change your approach to driving innovation, make sure the adoption and leadership teams are ready to make these important commitments.

Commitment to prioritize customers over technology

Every development team has a set of tools or technologies that they're most familiar with. It's wise to play to those strengths and use what you know. However, for innovation to be successful, teams must maintain a focus on customer needs and the hypothesis being tested. At times, this focus may not align with the capabilities of a particular tool or architectural approach. To be successful in innovation, the development team must remain open-minded. During the invention process, focus technical decisions on the needs of the customer over the preferences of your team.

Commitment to transparency

To understand measurement in an innovation approach, you must first understand the commitment to transparency. Innovation can only thrive in an environment that adheres to a *growth mindset*. At the root of a growth mindset is a cultural imperative to learn from experiences. Successful innovation and continuous learning start with a commitment to transparency in measurement. This is a brave commitment for the cloud adoption team. However, that commitment is meaningless if it's not matched by a commitment to preserve transparency within the leadership and cloud strategy teams.

Transparency is important because measuring customer impact doesn't address the question of right or wrong. Nor are impact measurements indicative of the quality of work or the performance of the adoption team. Instead, they represent an opportunity to learn and better meet your customers' needs. Misuse of innovation metrics can stifle that culture. Eventually, such misuse will lead to manipulation of metrics, which in turn causes long-term failure of the invention, the supporting staff, and ultimately the management structure who misused the data. Leaders and contributors alike should avoid using measurements for anything other than an opportunity to learn and improve the MVP solution.

Commitment to iteration

Only one promise rings true across all innovation cycles: you won't get it right on the first try. Measurement helps you understand what adjustments you should make to achieve the desired results. Changes that lead to favorable outcomes stem from iterations of the build-measure-learn process. The cloud adoption team and the cloud strategy team must commit to an iterative mindset before adopting a growth mindset or a build-measure-learn approach.

Next steps

Before building the next great invention, get started with customer adoption by understanding the build-measure-learn feedback loop.

Customer adoption with the build-measure-learn feedback loop

Build consensus on the business value of innovation

4/22/2021 • 5 minutes to read • Edit Online

The first step to developing any new innovation is to identify how that innovation can drive business value. In this exercise, you answer a series of questions that highlight the importance of investing ample time when your organization defines business value.

What is business value?

Business value is an informal term that might vary from business to business. It's the net benefit realized by the customer and includes all forms of value, which determines the health and well-being of the company in the long run.

Qualifying questions to determine business value

Before you develop any business innovation or business value solution (in the cloud or on-premises), validate your business value criteria by answering the following questions:

1. What is the defined customer need that you want to address with this solution?
2. What opportunities would this solution create for your business?
3. Which business outcomes are achieved with this solution?
4. Which of your company's motivations are served with this solution?

If the answers to all four questions are well documented, you might not need to complete the rest of this exercise. Fortunately, you can easily test any documentation. Set up two short meetings to test both the documentation and your organization's internal alignment. Invite committed business stakeholders to one meeting and set up a separate meeting with the engaged development team. Ask the four questions above to each group, and then compare the results.

> **NOTE**
> The existing documentation **should not** be shared with either team before the meeting. If true alignment exists, the guiding hypotheses should be referenced or even recited by members of each group.

> **WARNING**
> Don't facilitate the meeting. This test is to determine alignment; it's not an alignment creation exercise. When you start the meeting, remind the attendees that the objective is to test directional alignment to existing agreements within the team. Establish a five-minute time limit for each question. Set a timer and close each question after five minutes even if the attendees haven't agreed upon an answer.

Account for the different languages and interests of each group. If the test results in answers that are directionally aligned, consider this exercise a victory. You're ready to move on to solution development.

If one or two of the answers are directionally aligned, recognize that your hard work is paying off. You're already better aligned than most organizations. Future success is likely with minor continuing investment in alignment. Review each of the following sections for ideas that might help you build further alignment.

If either team fails to answer all four questions in 30 minutes, then alignment and the considerations in the following sections are likely to have a significant impact on this effort and others. Pay careful attention to each

of the following sections.

Address the big picture first

The Cloud Adoption Framework follows a prescribed path through four phases: strategizing, planning, readiness, and adoption. Cloud and business innovation fits within the adoption phase of this process. The answers to qualifying questions three and four concern outcomes and motivations. When these answers are misaligned, it indicates that your organization missed something during the strategizing phase of the cloud adoption lifecycle. Several of the following scenarios are likely to be at play.

- **Alignment opportunity:** When business stakeholders can't agree on motivations and business outcomes related to a cloud and business innovation effort, it's a symptom of a larger challenge. The exercises in the Strategy methodology can be useful in developing alignment among business stakeholders. Additionally, it's highly recommended that the same stakeholders form a cloud strategy team that meets regularly.
- **Communication opportunity:** When the development team can't agree on motivations and business outcomes, it might be a symptom of strategic communication gaps. You can quickly resolve this issue by reviewing the cloud strategy with the cloud adoption team. Several weeks after the review, the team should repeat the qualifying questions exercise.
- **Prioritization opportunity:** A cloud strategy is essentially an executive-level hypothesis. The best cloud strategies are open to iteration and feedback. If both teams understand the strategy, but still can't align answers to these questions, then priorities might be misaligned. Organize a session with the cloud adoption team and the cloud strategy team. This session can help the efforts of both groups. The cloud adoption team starts by sharing their aligned answers to the qualifying questions. From there, a conversation between the cloud adoption team and cloud strategy team can highlight opportunities to better align priorities.

These big picture opportunities often reveal ways to better align the innovative solution with the cloud strategy. This exercise has two common outcomes:

- These conversations can help your team improve your organization's cloud strategy and better represent important customer needs. Such a change can result in greater executive support for your team.
- Conversely, these conversations might show that your cloud adoption team should invest in a different solution. In this case, consider migrating this solution before continuing to invest in innovation. Alternately, these conversations might indicate that you adopt a citizen developer approach to test the business value first. In either case, they will help your team avoid making a large investment with limited business returns.

Address solution alignment

It's fairly common for the answers to questions one and two to be misaligned. During the early stages of ideation and development, customer need and business opportunity often get out of alignment. Many development teams find it challenging to achieve a balance between too much and too little definition. The Cloud Adoption Framework recommends lean approaches like build-measure-learn feedback loops to answer these questions. The following list shows opportunities and approaches to create alignment.

- **Hypothesis opportunity:** It's common for various stakeholders and development teams to have too many expectations for a solution. Unrealistic expectations can be a sign that the hypothesis is too vague. Follow the guidance on building with customer empathy to construct a clearer hypothesis.
- **Build opportunity:** Teams might be misaligned because they disagree on the way to solve the customer need. Such disagreement typically indicates that the team is being delayed by a premature technical spike. To keep the team focused on the customer, start the first iteration and build a small minimum viable product (MVP) to address part of the hypothesis. For more information, see Develop digital inventions.
- **Training opportunity:** Either team can be misaligned because they need deep technical requirements and

extensive functional requirements. This need can lead to an opportunity for training in agile methodologies. When the team culture isn't ready for agile processes, you might find innovation and keeping pace with the market to be a challenge. For training resources about DevOps and agile practices, see:

- Evolve your DevOps practices
- Build applications with Azure DevOps
- Deploy applications with Azure DevOps

By following the Innovate methodology and the backlog management tools in each section of this article, you can help create solution alignment.

Next steps

After you've aligned your business value proposition and communicated it, you're ready to start building your solution.

Return to the innovate exercises for next steps

Create customer partnerships through the build-measure-learn feedback loop

4/22/2021 • 2 minutes to read • Edit Online

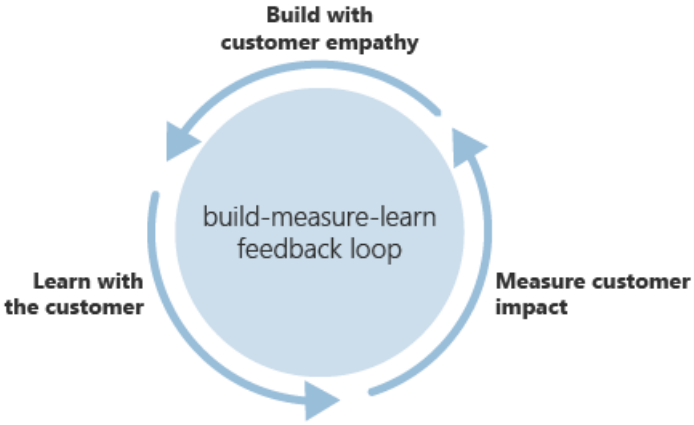
True innovation comes from the hard work of building solutions that demonstrate customer empathy, from measuring the impact of those changes on the customer, and from learning with the customer. Most importantly, it comes from feedback over multiple iterations.

If the past decade has taught us anything about innovation, it's that the old rules of business have changed. Large, wealthy incumbents no longer have an unbreakable hold on the market. The first or best players to market aren't always the winners. Having the best idea doesn't lead to market dominance. In a rapidly changing business climate, market leaders are the most agile.

Large or small, the companies that thrive in the digital economy as innovative leaders are those who listen to their customer base. That skill can be cultivated and managed. At the core of all good customer partnerships is a clear feedback loop. The process for building customer partnerships within the Cloud Adoption Framework is the build-measure-learn feedback loop.

What is the build-measure-learn feedback loop?

The build-measure-learn feedback loop is a description of the process for building empathy with your customers, measuring their reactions, and learning what adjustments to make that improve customer interactions. As described in Innovation in the digital economy, innovation requires a balance between invention and adoption. Customer feedback and partnership drive adoption. By turning your customers into strong, loyal partners during innovation cycles, you can realize better products and gain quicker traction in the market.



This process for managing customer partnerships and integrating them into your innovation efforts includes three phases of development:

- Build with customer empathy
- Measure for customer impact
- Learn with customers

Each phase of the build-measure-learn feedback loop process helps you build better solutions with your customers.

Next steps

Learn how to build with customer empathy to begin your build-measure-learn cycle.

Build with customer empathy

Build with customer empathy

4/22/2021 • 11 minutes to read • Edit Online

"Necessity is the mother of invention." This proverb captures the indelibility of the human spirit and our natural drive to invent. As explained in the Oxford English Dictionary, "When the need for something becomes imperative, you're forced to find ways of getting or achieving it." Few would deny these universal truths about invention. However, as described in Innovation in the digital economy, cloud innovation requires a balance between invention and adoption.

Continuing with the analogy, innovation comes from a more extended family. *Customer empathy is the proud parent of innovation.* Creating a customer empathy solution that drives innovation requires a legitimate customer need that keeps the customer coming back to solve critical challenges. These solutions are based on what a customer needs rather than on their wants or whims. To find their true needs, we start with empathy, a deep understanding of the customer's experience. Empathy is an underdeveloped skill for many engineers, product managers, and even business leaders. Fortunately, the diverse interactions and rapid pace of the cloud architect role have already started fostering this skill.

How do we build empathy, and why is customer empathy so important? Customer empathy helps us understand and share in the experience of the customer. From the first release of a minimum viable product (MVP) to the general availability of a market-grade solution, customer empathy helps us build a better solution. More importantly, it better positions us to invent solutions that will encourage adoption. In a digital economy, those who can most readily empathize with customer needs can build a brighter future that redefines and leads the market.

How to build with customer empathy

Defining assumptions is a fundamental part of planning. The more we plan, the more we see assumptions creep into the foundation of a great idea. Assumptions are typically the product of self-empathy. In other words, *what would I want if I were in this position?* Starting with the build phase minimizes the period in which assumptions can invade a solution. This approach also accelerates the feedback loop with real customers, triggering earlier opportunities to learn and sharpen empathy.

Caution

Properly defining what to build can be tricky and requires some practice. If you build something too quickly, if might not reflect customer needs. If you spend too much time trying to understand initial customer needs and solution requirements, the market might meet them before you have a chance to build anything at all. In either scenario, the opportunity to learn can be significantly delayed or reduced. Sometimes the data can even be corrupted.

The most innovative solutions in history began with an intuitive belief. That gut feeling comes from both existing expertise and firsthand observation. We start with the build phase because it allows for a rapid test of that intuition. From there, we can cultivate deeper understanding and clearer degrees of empathy. At every iteration or release of a solution, balance comes from building MVPs that demonstrate customer empathy.

To steady this balancing act, the following two sections discuss the concepts of how to build with empathy and defining an MVP.

Define a customer focused-hypothesis

Building with empathy means creating a solution based on defined hypotheses that illustrate a specific customer need. The following steps aim to formulate a hypothesis that will encourage building with empathy.

1. When you build with empathy, the customer is always the focus. This intention can take many shapes. You

could reference a customer archetype, a specific persona, or even a picture of a customer in the midst of the problem you want to solve. And keep in mind that customers can be internal (employees or partners) or external (consumers or business customers). This definition is the first hypothesis to be tested: can we help this specific customer?

2. Understand the customer experience. Building with empathy means you can relate to the customer's experience and understand their challenges. This mindset indicates the next hypothesis to be tested: can we help this specific customer with this manageable challenge?
3. Define a simple solution to a single challenge. Relying on expertise across people, processes, and subject matter experts will lead to a potential solution. This is the full hypothesis to be tested: can we help this specific customer with this manageable challenge through the proposed solution?
4. Arrive at a value statement. What long-term value do you hope to provide to these customers? The answer to this question creates your full hypothesis: how will these customers' lives be improved by using the proposed solution to address this manageable challenge?

This last step is the culmination of a customer empathy-driven hypothesis. It defines the audience, the problem, the solution, and the metric by which improvement is to be made, all of which center on the customer. During the measure and learn phases, each hypothesis should be tested. Changes in the customer, problem statement, or solution are anticipated as the team develops greater empathy for the addressable customer base.

Caution

The goal is to *build* with customer empathy, not to *plan* with it. It's all too easy to get stuck in endless cycles of planning and tweaking to hit upon the perfect customer empathy statement. Before you try to develop such a statement, review the following sections on defining and building an MVP.

After core assumptions are proven, later iterations will focus on growth tests in addition to empathy tests. After empathy is built, tested, and validated, you can begin to understand the addressable market at scale. This can be done through an expansion of the standard hypothesis formula described earlier. Based on available data, estimate the size of the total market (the number of potential customers).

From there, estimate the percentage of that total market that experiences a similar challenge and that might therefore be interested in this solution. This is your addressable market. The next hypothesis to be tested is: how will *x%* of customers' lives be improved by using the proposed solution to address this manageable challenge? A small sampling of customers will reveal leading indicators that suggest a percentage impact on the pool of customers engaged.

Define a solution to test the hypothesis

During each iteration of a build-measure-learn feedback loop, your attempt to build with empathy is defined by an MVP.

An MVP is the smallest unit of effort (invention, engineering, application development, or data architecture) required to create enough of a solution to learn *with the customer*. The goal of every MVP is to test some or all of the prior hypotheses and to receive feedback directly from the customer. The output is not a beautiful application with all the features required to change your industry. The desired output of each iteration is a learning opportunity, a chance to more deeply test a hypothesis.

Timeboxing is a standard way to make sure a product remains lean. For example, make sure your development team thinks the solution can be created in a single iteration to allow for rapid testing. To better understand using velocity, iterations, and releases to define what minimal means, see Planning velocity, iterations, release, and iteration paths.

Reduce complexity and delay technical spikes

The disciplines of invention found in the Innovate methodology describe the functionality that's often required to deliver a mature innovation or scale-ready MVP solution. Use these disciplines as a long-term guide for feature inclusion. Likewise, use them as a cautionary guide during early testing of customer value and empathy in your solution.

Feature breadth and the different disciplines of invention can't all be created in a single iteration. It might take several releases for an MVP solution to include the complexity of multiple disciplines. Depending on the investment in development, there might be multiple parallel teams working within different disciplines to test multiple hypotheses. Although it's smart to maintain architectural alignment between those teams, it's unwise to try to build complex, integrated solutions until value hypotheses can be validated.

Complexity is best detected in the frequency or volume of *technical spikes*. Technical spikes are efforts to create technical solutions that can't be easily tested with customers. When customer value and customer empathy are untested, technical spikes represent a risk to innovation and should be minimized. For the types of mature tested solutions found in a migration effort, technical spikes can be common throughout adoption. However, they delay the testing of hypotheses in innovation efforts and should be postponed whenever possible.

A relentless simplification approach is suggested for any MVP definition. This approach means removing anything that doesn't add to your ability to validate the hypothesis. To minimize complexity, reduce the number of integrations and features that aren't required to test the hypothesis.

Build an MVP

At each iteration, an MVP solution can take many different shapes. The common requirement is only that the output allows for measurement and testing of the hypothesis. This simple requirement initiates the scientific process and allows the team to build with empathy. To deliver this customer-first focus, an initial MVP might rely on only one of the disciplines of invention.

In some cases, the fastest path to innovation means temporarily avoiding these disciplines entirely, until the cloud adoption team is confident that the hypothesis has been accurately validated. Coming from a technology company like Microsoft, this guidance might sound counterintuitive. However, this simply emphasizes that customer needs, not a specific technology decision, are the highest priority in an MVP solution.

Typically, an MVP solution consists of a simple application or data solution with minimal features and limited polish. For organizations that have professional development expertise, this path is often the fastest one to learning and iteration. The following list includes several other approaches a team might take to build an MVP:

- A predictive algorithm that's wrong 99 percent of the time but that demonstrates specific desired outcomes.
- An IoT device that doesn't communicate securely at production scale but that demonstrates the value of nearly real-time data within a process.
- An application built by a citizen developer to test a hypothesis or meet smaller-scale needs.
- A manual process that re-creates the benefits of the application to follow.
- A wireframe or video that's detailed enough to allow the customer to interact.

Developing an MVP shouldn't require massive amounts of development investment. Preferably, investment should be as constrained as possible to minimize the number of hypotheses being tested at one time. Then, in each iteration and with each release, the solution is intentionally improved toward a scale-ready solution that represents multiple disciplines of invention.

Accelerate MVP development

Time to market is crucial to the success of any innovation. Faster releases lead to faster learning. Faster learning leads to products that can scale more quickly. At times, traditional application development cycles can slow this process. More frequently, innovation is constrained by limits on available expertise. Budgets, headcount, and availability of staff can all create limits to the number of new innovations a team can handle.

Staffing constraints and the desire to build with empathy have spawned a rapidly growing trend toward citizen developers. These developers reduce risk and provide scale within an organization's professional development community. Citizen developers are subject matter experts where the customer experience is concerned, but they're not trained as engineers. These individuals use prototyping tools or lighter-weight development tools that might be frowned upon by professional developers. These business-aligned developers create MVP solutions and test theories. When aligned well, this process can create production solutions that provide value

but don't pass a sufficiently effective scale hypothesis. They can also be used to validate a prototype before scale efforts begin.

Within any innovate plan, cloud adoption teams should diversify their portfolios to include citizen developer efforts. By scaling development efforts, more hypotheses can be formed and tested at a reduced investment. When a hypothesis is validated and an addressable market is identified, professional developers can harden and scale the solution by using modern development tools.

Final build gate: Customer pain

When customer empathy is strong, a clearly existing problem should be easy to identify. The customer's pain should be obvious. During build, the cloud adoption team is building a solution to test a hypothesis based on a customer pain point. If the hypothesis is well-defined but the pain point is not, the solution is not truly based on customer empathy. In this scenario, build is not the right starting point. Instead, invest first in building empathy and learning from real customers. The best approach for building empathy and validating pain is simple: listen to your customers. Invest time in meeting with and observing them until you can identify a pain point that occurs frequently. After the pain point is well-understood, you're ready to test a hypothesized solution for addressing that pain.

When not to apply this approach

There are many legal, compliance, and industry requirements that might necessitate an alternate approach. If public releases of a developing solution create risk to patent timing, intellectual property protection, customer data leaks, or violation of established compliance requirements, this approach might not be suitable. When perceived risks like these exist, consult legal counsel before adopting any guided approach to release management.

References

Some of the concepts in this article build on topics discussed in The Lean Startup by Eric Ries.

Next steps

After you've built an MVP solution, you can measure the empathy value and scale value. Learn how to measure for customer impact.

Measure for customer impact

How to measure for customer impact

5/18/2021 • 4 minutes to read • Edit Online

There are several ways to measure for customer impact. This article helps you define business metrics to validate hypotheses that arise out of an effort to build with customer empathy.

Strategic metrics

The Strategy methodology examines motivations and business outcomes. These practices provide a set of business metrics to test customer impact. When innovation is successful, results align with your strategic goals.

What are metrics in business? Business metrics are quantifiable metrics used to track and assess a specific business goal. Before establishing customer impact learning metrics, define strategic business metrics that you want this innovation to affect. Generally, those strategic metrics align with one or more of the following outcome areas:

- Business agility
- Customer engagement
- Customer reach
- Financial impact
- Solution performance, in the case of operational innovation.

Document your business metrics and track their impact frequently. Don't expect results in any of these metrics to emerge for several iterations. For more information about setting and aligning expectations across the parties involved, see Commitment to iteration.

Aside from motivation and business outcome metrics, this article focuses on learning metrics designed to guide transparent discovery and customer-focused iterations. For more information, see Commitment to transparency.

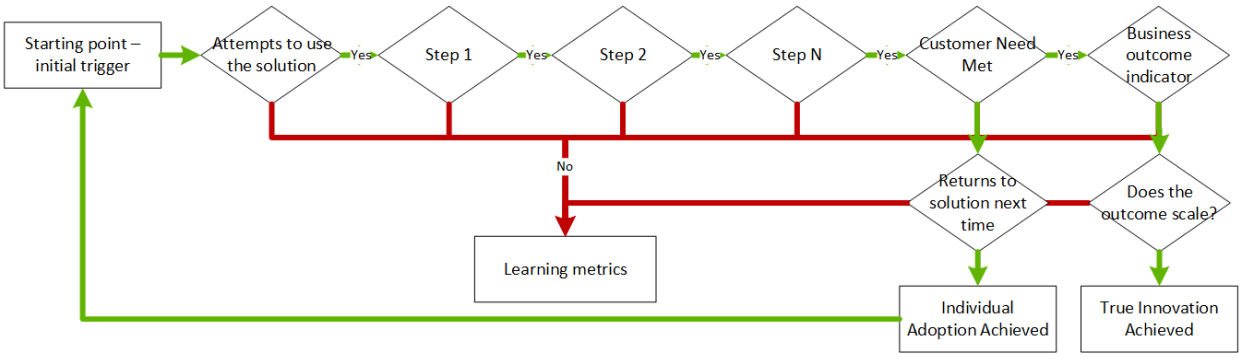
Learning metrics

When you share the first version of a minimum viable product (MVP) with a customer, there won't be an impact on strategic metrics. Several iterations later, the team may still struggle to change behaviors enough to affect strategic metrics. During learning processes, such as build-measure-learn cycles, consider adopting learning metrics. These metrics enhance tracking and learning opportunities.

Customer flow and learning metrics

If an MVP solution validates a customer-focused hypothesis, the solution drives some change in customer behavior. Those behavior changes across customer cohorts should improve business outcomes. Changing customer behavior is a process. Each step provides an opportunity to measure impact. The adoption team can keep learning along the way and build a better solution.

Learning about changes to customer behavior starts by mapping the flow that you hope to see from an MVP solution.



In most cases, a customer flow has an easily defined starting point and no more than two endpoints. Between the start and endpoints are various learning metrics to be used as measures in the feedback loop. Here are the steps to measure customer impact using the customer flow:

- **Starting point (initial trigger):** The starting point is the scenario that triggers the need for this solution. For a solution built with customer empathy, that initial trigger inspires a customer to try the MVP solution.
- **Solution steps:** Steps required to move the customer from the initial trigger to a successful outcome. Each step produces a learning metric based on a customer decision to move on to the next step.
- **Customer need met:** A solution that meets a customer need validates the hypothesis.
- **Individual adoption achieved:** If the customer returns to the solution the next time they find the trigger, individual adoption has been achieved.
- **Business outcome indicator:** When a customer behaves in a way that contributes to the defined business outcome, a business outcome indicator is observed.
- **True innovation achieved:** When *business outcome indicators* and *individual adoption* both occur at the desired scale, you've realized true innovation.

Each step of the customer flow generates learning metrics. After each iteration or release, a new version of the hypothesis is tested. At the same time, tweaks to the solution are tested to reflect adjustments in the hypothesis. When customers follow the prescribed path in any given step, a positive metric is recorded. When customers deviate from the prescribed path, a negative metric is recorded.

These alignment and deviation counters create learning metrics. Each should be recorded and tracked as the cloud adoption team progresses toward business outcomes and true innovation. In Learn with customers, we'll discuss ways to apply these business metrics to learn and build better solutions.

Group and observe customer partners

The first measurement in defining learning metrics is the customer partner definition. Any customer who participates in innovation cycles qualifies as a customer partner. To accurately measure behavior, you should use a cohort model to define customer partners. In this model, customers are grouped to sharpen your understanding of their responses to changes in the MVP. These customer impact groups typically resemble the following groups:

- **Experiment or focus group:** Grouping customers based on their participation in a specific experiment designed to test changes over time.
- **Segment:** Grouping customers by the size of the company.
- **Vertical:** Grouping customers by the *industry vertical* they represent.
- **Individual demographics:** Grouping customers based on personal demographics like age and physical location.

These groupings help you validate learning metrics across cross-sections of those customers who choose to partner with you during your innovation efforts. All other metrics should be derived from definable customer groupings.

Next steps

As learning metrics accumulate, the team can begin to learn with customers.

Learn with customers

Some of the concepts in this article build on topics first described in The Lean Startup, written by Eric Ries.

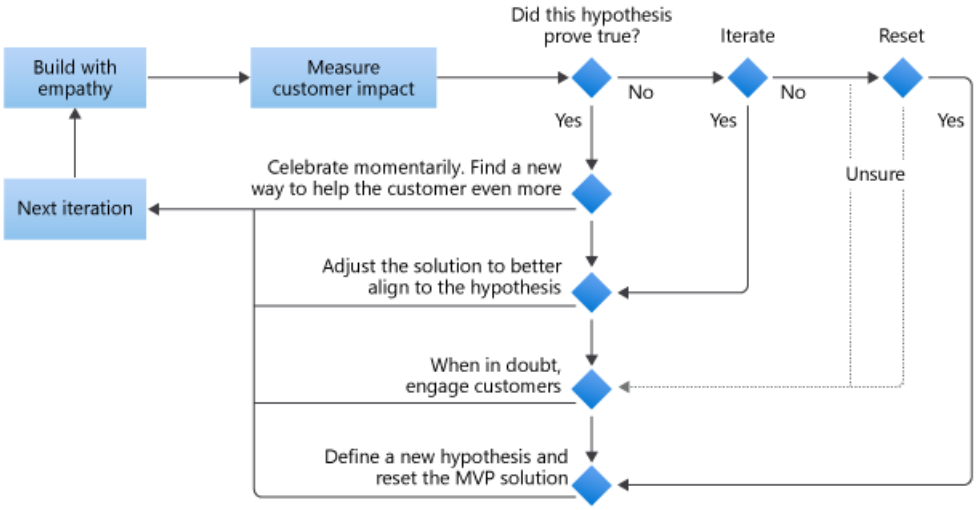
Learn with customers

5/20/2021 • 5 minutes to read • Edit Online

Our current customers represent our best resource for continuous learning. By partnering with us, customers help us build with customer empathy to find the best solution to their needs. They also help create a minimum viable product (MVP) solution by generating metrics from which we measure customer impact. In this article, we'll describe how to drive innovation learning with and from our customer-partners.

Continuous learning

Continuous learning is the process of learning new skills and abilities continuously. This continuous learning can come from both formal training and taking on challenges. At the end of every iteration, we have an opportunity to learn from the build and measure cycles. The following image offers an overview of the continuous learning process flow.



Continuous learning is a method for responding to learning metrics and assessing their impact on customer needs. Here are the primary learning decisions to make at the end of each iteration:

- **Did the hypothesis prove true?** When the answer is yes, celebrate for a moment and then move on. There are always more things to learn, more hypotheses to test, and more ways to help the customer in your next iteration. When a hypothesis proves true, it's a good time to decide on a new feature that enhances the solution's utility for the customer.
- **Can you get closer to a validated hypothesis by iterating on the current solution?** The answer is usually yes. Learning metrics typically suggest points in the process that lead to customer deviation. Use these data to find the root of a failed hypothesis. At times, the metrics may also suggest a solution.
- **Is a reset of the hypothesis required?** Sometimes you learn that the hypothesis or underlying requirement was flawed. When you find a flaw, an iteration alone isn't necessarily the right answer. If a reset is required, rewrite the hypothesis and review the solution in light of the new hypothesis. The sooner this type of learning occurs, the easier it is to pivot. Early hypotheses should focus on testing the riskiest aspects of the solution to avoid pivots later in development.
- **Unsure?** The second most common response after "iterate" is "we're not sure." Embrace this response. This response represents an opportunity to engage the customer, foster customer empathy, and to look beyond the data.

The answers to these questions shape the iteration to follow. Companies that demonstrate an ability to apply continuous learning and boldly make the right decisions for their customers are more likely to emerge as

leaders in their markets.

The practice of continuous learning requires a great deal of trial and error. It also requires some science and data-driven decision-making. Perhaps the most difficult part of adopting continuous learning concerns the cultural requirements. To effectively adopt continuous learning, your business culture must be open to a fail-first, customer-focused approach. The following section provides more details about this approach.

Growth mindset

Few could deny the radical transformation within Microsoft culture that's occurred over the last several years. This multifaceted transformation, led by Satya Nadella, has been hailed as a surprising business success story. At the heart of this story is the growth mindset. Here are a few key points of the growth mindset that inform learning with customers:

- **Customer first:** To design a hypothesis to improve the experience of customers, you have to meet real customers where they are. Don't just rely on metrics. Compare and analyze metrics based on firsthand observation of customer experiences.
- **Continuous learning:** Customer focus and customer empathy stem from a learn-it-all mindset. Strive for a learn-it-all, not know-it-all, mindset.
- **Beginner's mindset:** Demonstrate customer empathy by approaching every conversation with a beginner's mindset. Whether you're new to your field or a 30-year veteran, assume you know little, and you'll learn a lot.
- **Listen more:** Customers want to partner with you. An ego-driven need to be right blocks that partnership. To learn beyond the metrics, speak less and listen more.
- **Encourage others:** Don't just listen. Use the things you **do** say to encourage others. In every meeting, find ways to pull in diverse perspectives from those people who may not be quick to share.
- **Share the code:** If our obligation is to own a code base, we lose sight of the true power of innovation. Focus on owning and driving outcomes for your customers. Share your code, publicly with the world or privately within your company, to invite diverse perspectives into the solution and the code base.
- **Challenge what works:** Success doesn't mean you're demonstrating true customer empathy. Avoid having a fixed mindset and a bias toward doing what's worked before. Look for learning in positive and negative metrics by engaging your customers.
- **Be inclusive:** Work hard to invite diverse perspectives into the mix. There are many variables that can divide humans into segregated groups. Cultural norms, past behaviors, gender, religion, sexual preference, even physical abilities. True innovation comes when we challenge ourselves to see past our differences and consciously strive to include all customers, partners, and coworkers.

The build-measure-learn feedback loop

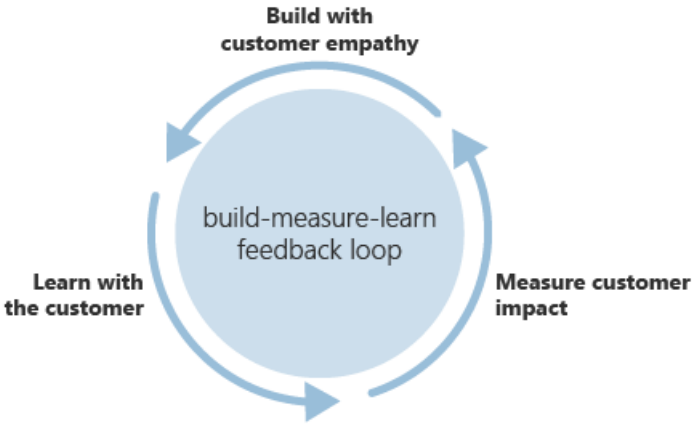
True innovation comes from the hard work of building solutions that demonstrate customer empathy, from measuring the impact of those changes on the customer, and from learning with the customer. Most importantly, it comes from feedback over multiple iterations.

If the past decade has taught us anything about innovation, it's that the old rules of business have changed. Large, wealthy incumbents no longer have an unbreakable hold on the market. The first or best players to market are not always the winners. Having the best idea doesn't lead to market dominance. In a rapidly changing business climate, market leaders are the most agile. Those who can adapt to changing conditions lead.

Large or small, the companies that thrive in the digital economy as innovative leaders are those organizations with the greatest ability to listen to their customer base. That skill can be cultivated and managed. At the core of all good partnerships is a clear feedback loop. The process for building customer partnerships within the Cloud Adoption Framework is the build-measure-learn feedback loop.

Innovation requires a balance between invention and adoption. Customer feedback and partnership drive adoption. By turning your customers into strong, loyal partners during innovation cycles, you can realize better

products and gain quicker traction in the market.



This process for managing customer partnerships and integrating them into your innovation efforts includes three phases of development:

- Build with customer empathy
- Measure for customer impact
- Learn with customers

Each phase of the process helps you build better solutions with your customers.

Next steps

As a next step to understanding this methodology, see Common blockers and challenges to innovation to prepare for the changes ahead.

Some of the concepts in this article build on topics first described in The Lean Startup, written by Eric Ries.

Common technology adoption blockers and challenges to innovation

4/22/2021 • 5 minutes to read • Edit Online

As described in Innovation in the digital economy, innovation requires a balance between invention and adoption. This article expands on the common cloud adoption challenges and blockers to innovation, as it aims to help you understand how this approach can add value during your innovation cycles.

Formula for innovation: **innovation = invention + adoption**

Knowing how to overcome innovation challenges takes some time to discover the right methods. This article delves into overcoming technology adoption challenges in the workplace.

Cloud technology adoption challenges

Although cloud technology advances have reduced some of the friction related to adoption, technology adoption is more people-centric than technology-centric. And unfortunately, the cloud can't fix people.

The following list describes some of the most common adoption challenges related to innovation. As you progress through the Innovate methodology, each of the challenges in the following sections are identified and addressed. Before you apply this methodology, evaluate your current innovation cycles to determine which are the most important challenges or blockers for you. Then, use the methodology to address or remove those blockers.

Types of external challenges

- **Time to market:** In a digital economy, time to market is one of the most crucial indicators of market domination. Surprisingly, time to market impact has little to do with positioning or early market share. Both of these factors are fickle and temporary. The time to market advantage comes from the simple truth that the more time your solution has on the market, the more time you have to learn, iterate, and improve. To shorten time to market and accelerate learning opportunities, focus on quick definition and rapid build of an effective minimum viable product.
- **Competitive challenges:** Dominant incumbents reduce opportunities to engage and learn from customers. Competitors also create external pressure to deliver more quickly. Build fast, but invest heavily in understanding the proper *measures*. Well-defined niches produce more actionable feedback measures and enhance your ability to partner and learn, resulting in better overall solutions.
- **Understand your customer:** Customer empathy starts with an understanding of the customer and customer base. One of the biggest challenges for innovators is the ability to rapidly categorize measurements and learning within the build-measure-learn cycle. It's important to understand your customer through the lenses of market segmentation, channels, and types of relationships. Throughout the build-measure-learn cycle, these data points help create empathy and shape the lessons learned.

Types of internal challenges

- **Choosing innovation candidates:** When investing in innovation, healthy companies spawn an endless supply of potential inventions. Many of these create compelling business cases that suggest high returns and generate enticing business justification spreadsheets. As described in the build article, *building with customer empathy* should be prioritized over invention that's based only on gain projections. If customer empathy isn't visible in the proposal, long-term adoption is unlikely.
- **Balancing the portfolio:** Most technology implementations don't focus on changing the market or improving the lives of customers. In the average IT department, more than 80% of workloads are maintained

for basic process automation. With the ease of innovation, it's tempting to innovate and rearchitect those solutions. Most of the times, those workloads can experience similar or better returns by migrating or modernizing the solution, with no change to core business logic or data processes. Balance your portfolio to favor innovation strategies that can be *built* with clear empathy for the customer (internal or external). For all other workloads, follow a migrate path to financial returns.

- **Maintaining focus and protecting priorities:** When you've made a commitment to innovation, it's important to maintain your team's focus. During the first iteration of a build phase, it's relatively easy to keep a team excited about the possibilities of changing the future for your customers. However, that first MVP release is just the beginning. True innovation comes with each build-measure-learn cycle, by learning from the feedback loops to produce a better solution. As a leader in any innovation process, concentrate on keeping the team focused and on maintaining your innovation priorities through the subsequent, less-glamorous build iterations.

Invention challenges

Before the widespread adoption of the cloud, invention cycles that depended on information technology were laborious and time-consuming. Procurement and provisioning cycles frequently delayed the crucial first steps toward any new solutions. The cost of DevOps solutions and feedback loops delayed teams' abilities to collaborate on early stage ideation and invention. Costs related to developer environments and data platforms prevented anyone but highly trained professional developers from participating in the creation of new solutions.

The cloud has overcome many of these invention challenges by providing self-service automated provisioning, light-weight development and deployment tools, and opportunities for professional developers and citizen developers to cooperate in creating rapid solutions. Using the cloud for innovation dramatically reduces customer challenges and blockers to the invention side of the innovation equation.

Invention and innovation challenges in a digital economy

The invention challenges of today are different than challenges of the past. The endless potential of cloud technologies also produces more implementation options and deeper considerations about how those implementations might be used.

The Innovate methodology uses the following innovation disciplines to help align your implementation decisions with your invention and adoption goals:

- **Data platforms:** New sources and variations on data are available. Previously, much of this data couldn't be integrated into legacy or on-premises applications to create cost-effective solutions. Understanding the change you hope to drive in customers will inform your data platform decisions. Those decisions will be an extension of selected approaches to ingest, integrate, categorize, and share data. Microsoft refers to this decision-making process as the democratization of data.
- **Device interactions:** IoT, mobile, and augmented reality blur the lines between digital and physical, accelerating the digital economy. Understanding the real-world interactions surrounding customer behavior will drive decisions about device integration.
- **Applications:** Applications are no longer the exclusive domain of professional developers. Nor do they require traditional server-based approaches. Empowering professional developers, enabling business specialists to become citizen developers, and expanding compute options for API, micro-services, and PaaS solutions expand application interface options. Understanding the digital experience required to shape customer behavior will improve your decision-making about application options.
- **Source code and deployment:** Collaboration between developers of all walks improves both quality and speed to market. Integration of feedback and a rapid response to learning shape market leaders. Commitment to the build, measure, and learn processes help accelerate tool adoption decisions.
- **Predictive solutions:** In a digital economy, it's seldom sufficient to just meet the current needs of your customers. Customers expect businesses to anticipate their next steps and predict their future needs. Continuous learning often evolves into prediction tooling. The complexity of customer needs and the

availability of data will help define the best tools and approaches to predict and influence.

In a digital economy, the greatest challenge architects face is to clearly understand their customers' invention and adoption needs and to then determine the best cloud-based toolchain to deliver on those needs.

Next steps

With the knowledge you've gained about the build-measure-learn model and a growth mindset, you're ready to develop digital inventions within the Innovate methodology.

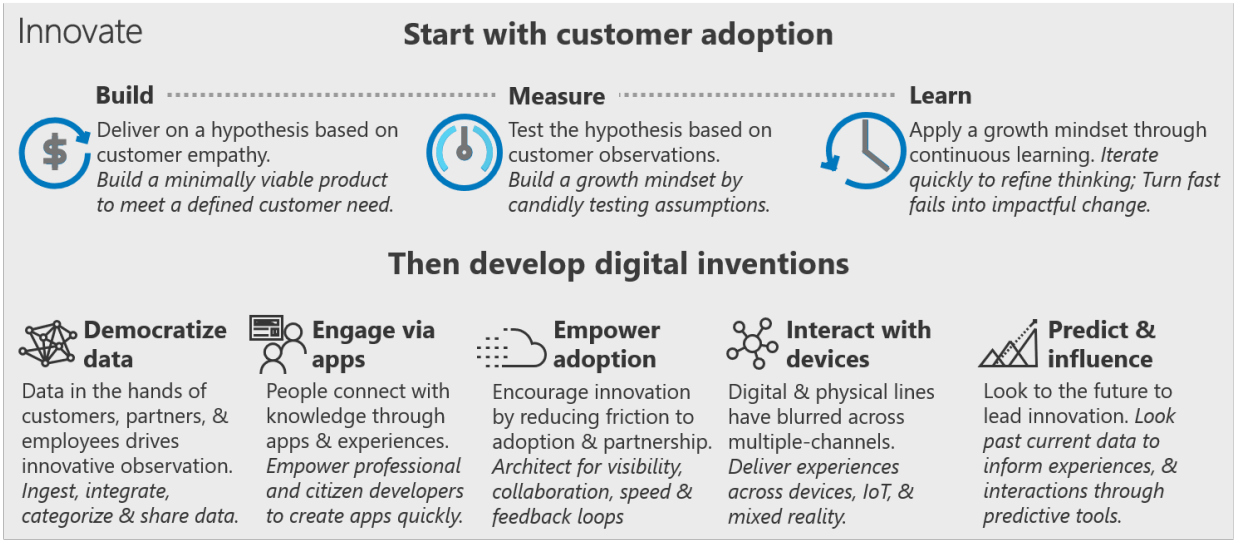
Develop digital inventions

Develop digital inventions

5/18/2021 • 2 minutes to read • Edit Online

What are digital inventions? As described in Innovation in the digital economy, innovation requires a balance between invention and adoption. Digital inventions are the products of technological innovation that solve customer's needs and provide innovative solutions. Customer feedback and partnership are required to drive adoption, preferably using the build-measure-learn feedback loop. For more information, see Create customer partnerships through the build-measure-learn feedback loop.

The types of innovation and disciplines described in the next section introduce a series of approaches to developing digital inventions while keeping adoption and customer empathy in mind. Each discipline is briefly described, along with links to each process.



Summary of each discipline of digital invention

There are different types of innovation for product development. Not every discipline is required to drive innovation for each specific case. By following the guidance in Build with customer empathy, you'll test a hypothesis in every iteration. Defining the output of each iteration as a minimum viable product (MVP) should enable you to create innovation with the smallest number of disciplines.

- Democratize data: By getting data into the hands of customers, partners, and employees, you encourage innovative observation. Ingest, centralize, govern, and share data.
- Engage via applications: People connect with knowledge through applications and experiences. Empower professional and citizen developers to create applications quickly.
- Empower adoption: Encourage innovation by reducing friction to adoption and partnership. Architect for visibility, collaboration, speed, and feedback loops.
- Interact with devices: Digital and physical lines have blurred across multiple-channels. Deliver experiences across devices, IoT, and mixed reality.
- Predict and influence: Look to the future to lead innovation. Look past current data to inform experiences and interactions through predictive tools.

Next steps

Democratization of data is the first discipline of innovation to consider and evaluate.

Democratize data with digital invention

5/14/2021 • 6 minutes to read • Edit Online

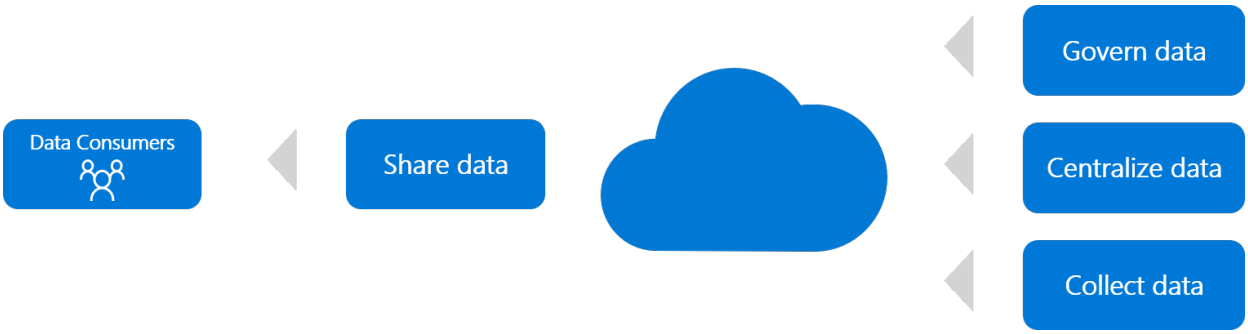
Coal, oil, and human potential were the three most consequential assets during the industrial revolution. These assets built companies, shifted markets, and ultimately changed nations. In the digital economy, there are three equally important assets for innovation—data, devices, and human potential. Each of these assets holds great innovation potential. For any innovation effort in the modern era, data is the new oil.

In every company, there is data that can be used to find and meet customer needs. Unfortunately, the process of mining that data to drive innovation can be costly and time-consuming, so needs aren't discovered and solutions aren't created. Data democratization can solve this problem.

What is data democratization? It's the process of getting data into the the right hands to drive innovation. This democratization process can take several forms, but they generally include solutions for ingested or integrated raw data, centralization of data, sharing data, and securing data. When data is democratized, experts around the company can use it to form and test hypotheses. In many cases, cloud adoption teams can build with customer empathy using only data, to rapidly meet customer needs.

Ways to democratize data

There are various ways to democratize data, but most include methods of collecting, centralizing, governing, and sharing the data. The following sections describe some of these methods. When you build a solution to a customer hypothesis, you should assess whether to democratize data, to what extent, and how to do it.



Share data

When you build with customer empathy, customer needs guide the solution. If the need is data, the solution enables the customer to interrogate, analyze, and report on the data directly, with no support from IT staff.

Many successful innovations start as a minimum viable product (MVP) that delivers data to the customer. An MVP is a version of the product that has just enough features to be usable by the customer. It shows the possible potential of the product in order to gather feedback from the customer. In this concierge model, an employee is the data consumer. That employee uses data to aid the customer. Each time the customer engages manual support, a hypothesis can be tested and validated. This approach is often a cost effective means of testing a customer-focused hypothesis before you invest heavily in integrated solutions.

The primary tools for sharing data directly with data consumers include self-service reporting or data embedded within other experiences, using tools like Power BI.

> **NOTE**
> Before you share data, make sure you've read the following sections. Sharing data might require governance to provide protection for the data. Also, if the data spans multiple clouds it might require centralization. If data resides within applications, you must collect it in order to share it.

Govern data

Sharing data can quickly produce a minimum viable product to use in customer conversations. However, to turn that shared data into useful and actionable knowledge, more is generally required.

After a hypothesis has been validated through data sharing, the next phase of development is typically data governance.

Data governance is a broad topic that can require its own dedicated framework, a matter that's outside the scope of the Cloud Adoption Framework.

There are several aspects of data governance to consider as soon as you validate the customer hypothesis. For example:

- **Is the shared data sensitive?** Data should be classified before being shared publicly to protect the interests of customers and the company.
- **If the data is sensitive, has it been secured?** Protection of sensitive data is a must for democratized data. The example workload discussed in securing data solutions provides some references for securing data.
- **Is the data cataloged?** Identifying the nature of the shared data aids in long-term data management. Tools for documenting data, like Azure Data Catalog, make this process much easier in the cloud. Guidance regarding the annotation of data and the documentation of data sources can accelerate the process.

When democratization of data is important to a customer-focused hypothesis, make sure the governance of shared data is in the release plan. This protects customers, data consumers, and the company.

Centralize data

Data centralization leads to more meaningful reporting, ensures that the same data is available across the organization, and increases your ROI. When data is dispersed across an IT environment, opportunities to innovate can be extremely constrained, expensive, and time-consuming. The cloud provides new opportunities to centralize data. When centralization of multiple data sources is required to build with customer empathy, the cloud can accelerate the testing of hypotheses.

Caution

Centralization of data represents a risk point in any innovation process. When data centralization is a technical spike, and not a source of customer value, we suggest that you delay centralization until the customer hypotheses have been validated.

When you centralize, you need an appropriate data store for the centralized data. It's a good practice to establish a data warehouse in the cloud. This scalable option provides a central location for all your data. This type of solution is available in online analytical processing (OLAP) or big data options.

The reference architectures for OLAP and big data solutions can help you choose the most appropriate centralization solution in Azure. If a hybrid solution is required, the reference architecture for extending on-premises data can also help accelerate solution development.

> **IMPORTANT**
> For some customer needs and solutions, a simple approach might be enough. The cloud architect should challenge the team to consider low-cost solutions to validate the customer hypothesis, especially during early development. This section on collecting data discusseses scenarios that might suggest a different solution for your situation.

Collect data

The two primary forms of data collection are *integration* and *ingestion*.

Integration: Data that resides in an existing data store can be integrated into the centralized data store by using traditional data movement techniques. This is especially common for scenarios that involve multicloud data storage. These techniques involve extracting the data from the existing data store and then loading it into the central data store. At some point in this process, the data is typically transformed to be more usable and relevant in the central store.

Cloud-based tools have turned these techniques into pay-per-use tools, reducing the barrier to entry for data collection and centralization. Tools like Azure Database Migration Service and Azure Data Factory are two examples. The reference architecture for Data Factory with an OLAP data store is an example of one such solution.

Ingestion: Some data doesn't reside in an existing data store. When this transient data is a primary source of innovation, you'll want to consider alternative approaches. Transient data can be found in a variety of existing sources like applications, APIs, data streams, IoT devices, a blockchain, an application cache, in media content, or even in flat files.

You can integrate these various forms of data into a central data store on an OLAP or big data solution. However, for early iterations of the build-measure-learn cycle, an online transactional processing (OLTP) solution might be sufficient to validate a customer hypothesis. OLTP solutions aren't the best option for any reporting scenario. However, when you're building with customer empathy, it's more important to focus on customer needs than on technical tooling decisions. After the customer hypothesis is validated at scale, a more suitable platform might be required. The reference architecture on OLTP data stores can help you determine which data store is most appropriate for your solution.

Virtualize: Integration and ingestion of data can sometimes slow innovation. When a solution for data virtualization is already available, it might represent a more reasonable approach. Ingestion and integration can both duplicate storage and development requirements, add data latency, increase attack surface area, trigger quality issues, and increase governance efforts. Data virtualization is a more contemporary alternative that leaves the original data in a single location and creates pass-through or cached queries of the source data.

SQL Server 2017 and Azure SQL Data Warehouse both support PolyBase, which is the approach to data virtualization most commonly used in Azure.

Next steps

With a strategy for democratizing data in place, you'll next want to evaluate approaches to application development.

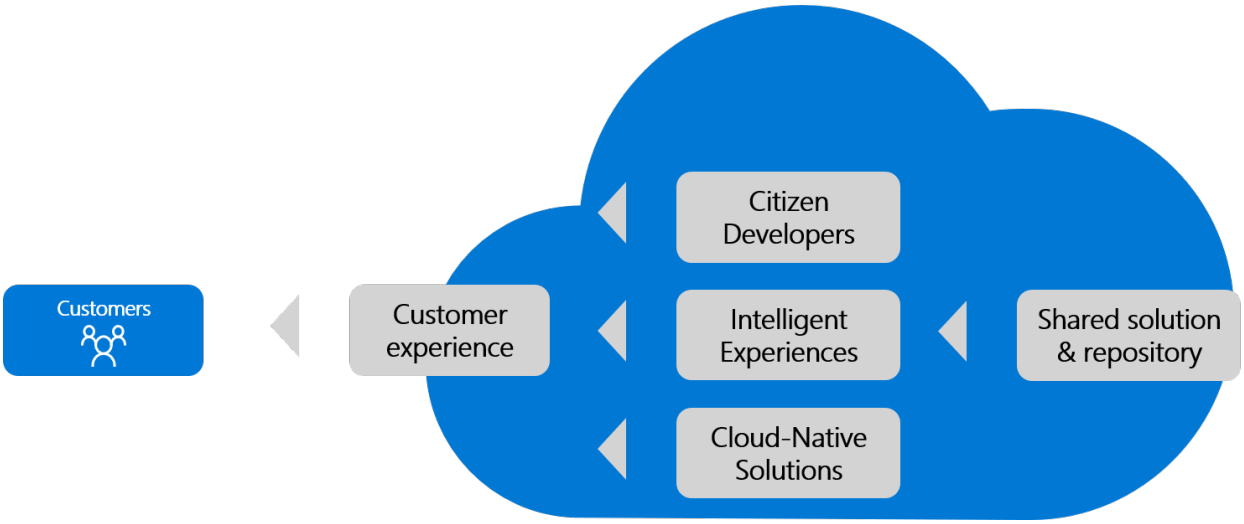
Application development for innovative apps

Application development for innovative apps

4/22/2021 • 7 minutes to read • Edit Online

As discussed in Democratize data with digital invention, data fuels most innovations across the digital economy. Building on that analogy, applications are the fueling stations and infrastructure required to get that fuel into the right hands.

In some cases, data alone is enough to drive change and meet customer needs. More commonly though, solutions to customer needs require applications to shape the data and create an experience. Innovative applications engage and interact with the user, providing information and guidance. This article summarizes several principles that can help you find the right application development solution, based on the hypotheses to be validated.



Shared code

Teams that are quick to respond to customer feedback, market changes, and opportunities typically innovate best. The first principle of innovative applications is an element of the growth mindset: "Share the code." Code sharing invites diverse perspectives and contributions, and spurs innovation. Therefore, application development should start with a shared code repository.

A widely adopted tool for managing code repositories is GitHub, which allows you to create a shared code repository quickly. An alternative is Microsoft Azure Repos, which is an Azure DevOps service that provides unlimited, cloud-hosted private repos for your project. For version control when you use Azure Repos, you can choose either Git, which is a distributed type, or Team Foundation Version Control (TFVC), which is centralized. For more information about Azure Repos, Git, and TFVC, see Azure Repos Documentation.

Citizen developers

Professional developers are important to innovation. When a hypothesis proves accurate at scale, they can stabilize the solution and prepare it for scale. Unfortunately, professional developers may be in short supply, and professional development can increase costs and slow innovation.

Citizen developers are users who create new business applications using development and runtime environments sanctioned by corporate IT. The use of citizen developers can help to scale development efforts and accelerate early hypothesis testing. This strategy is viable and effective when early hypotheses can be validated through tools like Power Apps for application interfaces, AI Builder for processes and predictions,

Power Automate for workflows, and Power BI for data consumption.

> **NOTE**
> When you rely on citizen developers to test hypotheses, it's advisable to also have professional developers to support, review, and guide the work. The professionals can help develop a robust design that accelerates returns on the innovation. By involving professional developers at the right time, you can realize cleaner transitions later.

Intelligent experiences

Intelligent experiences combine the speed and scale of modern web applications with the intelligence of cognitive services and bots. Alone, each of these technologies might be sufficient to meet your customers' needs. When properly combined, they broaden the spectrum of needs that can be met through a digital experience, while helping to contain application development costs.

Modern web apps

Modern web applications can be the fastest way to meet the needs of internal or external customers. The experiences they provide can engage customers quickly and allow for rapid evolution of the solution.

Adding intelligence

It gets easier all the time for professional and citizen developers to add machine learning and AI features to applications that help meet the needs of the customer and create an interactive experience. Some examples of these features are:

- Speech to text
- Text to speech
- Computer vision
- Visual search
- Predictive AI

Innovators should be alert to take advantage of such features to create an interactive and modern experience.

Bots

A bot is a conversational AI application that provides users an experience that is more like dealing with a person, and less like dealing with a conventional computer application. Users converse with a bot through text, interactive cards, and speech. A bot interaction can range from a quick question and answer, such as making a dinner reservation, to a sophisticated conversation that intelligently provides access to services.

Bots can do the same things as other types of software: read and write files, use databases and APIs, and handle regular computational tasks. What makes bots unique is their use of mechanisms generally reserved for human-to-human communication. Bots are a lot like modern web applications: they live on the internet and use APIs to send and receive messages. What's in a bot varies widely depending on what kind of bot it is. Modern bot software relies on a stack of technology and tools to deliver increasingly complex experiences on a variety of platforms. However, a simple bot could just receive a message and echo it back to the user with very little code involved.

Cloud-native solutions

Cloud-native architecture enables you to embrace rapid change, and run resilient and scalable applications more easily. Cloud-native applications are typically built using containers, microservices, managed services, serverless functions, and event-based programming. Most commonly, cloud-native solutions use continuous delivery to achieve faster time to market.

A cloud-native solution allows centralized development teams to maintain control of the business logic without

the need for monolithic, centralized solutions. It also creates an anchor to drive consistency across the input of citizen developers and modern experiences. Finally, cloud-native solutions provide an innovation accelerator by freeing citizen and professional developers to innovate safely and with a minimum of blockers.

Innovate through existing solutions

Many customer hypotheses can best be delivered by a modernized version of an existing solution. This can happen when the current business logic comes close to meeting customer needs.

Most forms of modernization, including refactoring, are included in the Migrate methodology within the Cloud Adoption Framework. That methodology guides cloud adoption teams through the process of migrating a digital estate to the cloud. The Azure migration guide provides a streamlined approach to the same methodology, which is suitable for a small number of workloads or even a single application.

After a solution has been migrated and modernized, there are a variety of ways it can be used to create new, innovative application solutions to meet customer needs. For example, citizen developers could test hypotheses, or professional developers could create intelligent experiences or cloud-native solutions.

Extend an existing solution

Extending a solution is one common form of modernization. This can be the fastest path to innovation when the following are true of the customer hypothesis:

- Existing business logic meets or comes close to customer needs.
- An improved experience, not a new one, best meets the needs of customers.
- The business logic required by the minimum viable product (MVP) solution has been centralized, usually via an n-tier, web services, API, or microservices design. This approach consists of wrapping the existing solution within a new experience hosted in the cloud. In Azure, this solution would likely live in Azure App Service.

Rebuild an existing solution

If an existing solution meets or comes close to meeting customer needs, but can't be easily extended, it may be necessary to refactor it. In this approach, the application is migrated to the cloud. After the application is migrated, parts of it are modified or duplicated, as web services or microservices, which are deployed in parallel with the existing solution. The parallel service-based solution could be treated like an extended solution. This solution would simply wrap the existing solution with a new experience hosted in the cloud. In Azure, this solution would likely live in Azure App Service.

Caution

Refactoring or rearchitecting solutions or centralizing business logic can quickly trigger a time-consuming technical spike instead of a source of customer value. This is a risk to innovation, especially early in hypothesis validation. With a bit of creativity in the design of a solution, there should be a path to MVP that doesn't require refactoring of existing solutions. It's wise to delay refactoring until the initial hypothesis can be validated at scale.

Operating model innovations

In addition to modern innovative approaches to application development, there have been notable innovations in application operations. These approaches have spawned many organizational movements. One of the most prominent is the cloud center of excellence operating model. When fully staffed and mature, business teams have the option to provide their own operational support for a solution.

The type of self-service operational management model found in a cloud center of excellence allows for tighter controls and faster iterations within the solution environment. These goals are accomplished by transferring operational control and accountability to the business team.

If you're trying to scale or meet global demand for an existing solution, this approach might be sufficient to validate a customer hypothesis. After a solution is migrated and slightly modernized, the business team can scale it to test a variety of hypotheses. These typically involve customer cohorts who are concerned with

performance, global distribution, and other customer needs hindered by IT operations.

Reduce overhead and management

The more there is to maintain within an innovative application or solution, the slower that application or solution will iterate. This means you can accelerate innovation by reducing the impact of operations on available bandwidth.

To prepare for the many iterations required to deliver an innovative solution, it's important to think ahead. For example, minimize operational burdens early in the process by favoring serverless options. In Azure, serverless application options could include Azure App Service or containers.

In parallel, consider the serverless transaction data options in Azure that can also reduce overhead. The Azure product catalog provides database options that host data without the need for a full data platform.

Next steps

Depending on the hypothesis and solution, the principles in this article can aid in designing applications that meet MVP definitions and engage users. Up next are the principles for empowering adoption, which offer ways to get the application and data into the hands of customers more quickly and efficiently.

Empower adoption

Empower adoption with digital invention

5/24/2021 • 8 minutes to read • Edit Online

The ultimate test of innovation is customer reaction to your invention. Did the hypothesis prove true? Do customers use the solution? Does it scale to meet the needs of the desired percentage of users? Most importantly, do they keep coming back? None of these questions can be asked until the minimum viable product (MVP) solution has been deployed.

In this article, we'll focus on empowering adoption with continuous integration and continuous deployment (CI/CD) pipeline tools. Continuous integration is the automating of code multiple times per day in order to have an updated single project. Continuous deployment is the automatic delivery of those functions throughout the day.

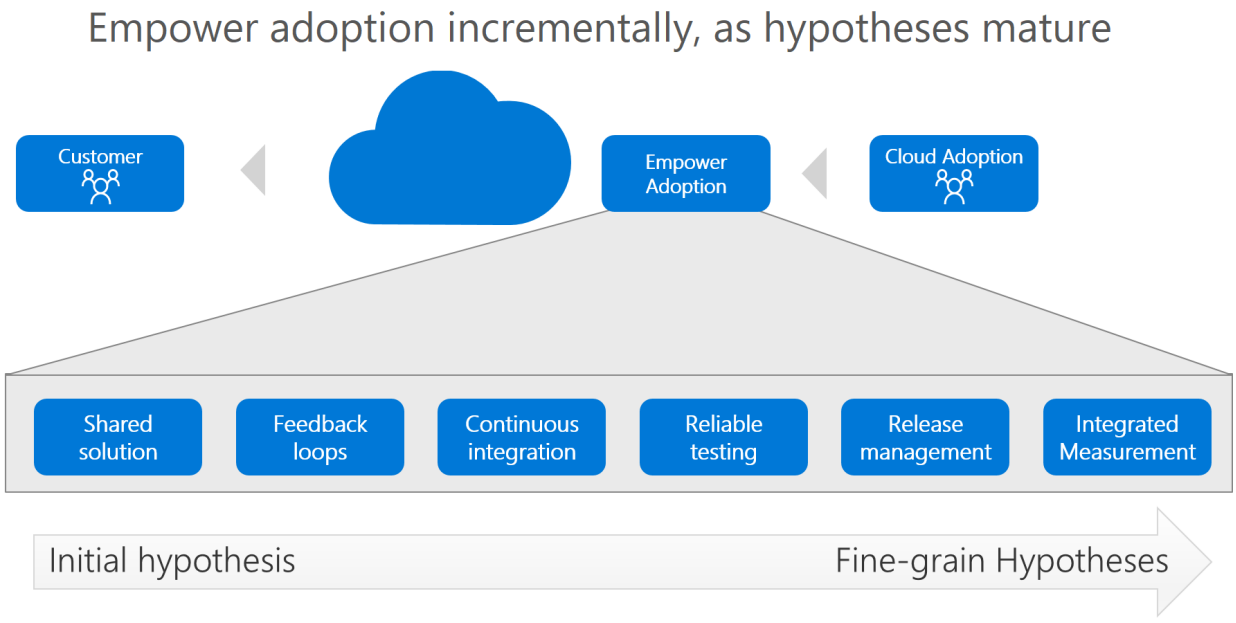
Reduce CI/CD friction that affects adoption

Some obstacles to adoption can be minimized through a combination of technology and processes. For readers with knowledge of CI/CD or DevOps processes, the following CI/CD pipeline processes will be familiar. This article establishes a starting point for cloud adoption teams that fuels innovation and feedback loops. This starting point might foster more robust CI/CD or DevOps approaches as the products and teams mature.

As described in Measure for customer impact, positive validation of any hypothesis requires iteration and determination. This CI/CD article aims to minimize technical spikes that slow innovation, while making sure you keep best practices in place. Doing so will help the team design for future success while delivering on current customer needs.

Empower adoption and digital invention: The maturity model

The primary goal of the Innovate methodology is to build customer partnerships and accelerate feedback loops, which lead to market innovations. The following image and sections describe initial implementations that support this methodology.



- Shared solution: Establish a centralized repository for all aspects of the solution.
- Feedback loops: Make sure that feedback loops can be managed consistently through iterations.
- Continuous integration: Regularly build and consolidate the solution.

- Reliable testing: Validate solution quality and expected changes to ensure the reliability of your testing metrics.
- Solution deployment: Deploy solutions so that the team can quickly share changes with customers.
- Integrated measurement: Add learning metrics to the feedback loop for clear analysis by the full team.

To minimize technical spikes, assume that maturity will initially be low across each of these principles. Plan ahead by aligning to tools and processes that can scale as hypotheses become more fine-grained. In Azure, GitHub and Azure DevOps allow small teams to get started with little friction. These teams might grow to include thousands of developers who collaborate on scale solutions and test hundreds of customer hypotheses. The rest of this article illustrates the "plan big, start small" approach to empowering adoption across each of these principles.

Shared solution

As described in Measure for customer impact, positive validation of any hypothesis requires iteration and determination. You'll experience far more failures than wins during any innovation cycle. This is expected. However, when a customer need, hypothesis, and solution align at scale, the world changes quickly.

When you're scaling digital invention and innovation, there's no more valuable tool than a shared code base for the solution. Unfortunately, there's no reliable way of predicting which iteration or which MVP will yield the winning combination. That's why it's never too early to establish a shared code base or repository. This is the one technical spike that shouldn't be delayed. As the team iterates through various MVP solutions, a shared repo enables easy collaboration and accelerated development. When changes to the solution drag down learning metrics, version control lets you roll back to an earlier, more effective version of the solution.

The most widely adopted CI/CD tool for managing code repositories is GitHub, which lets you create a shared code repository in just a few steps. Additionally, the Azure Repos feature of Azure DevOps can be used to create a Git or TFVC repository.

Feedback loops

Making the customer part of the solution is the key to building customer partnerships during innovation cycles. That's accomplished, in part, by measuring customer impact. It requires conversations and direct testing with the customer. Both generate feedback that must be managed effectively.

Every point of feedback is a potential solution to the customer need. More importantly, every bit of direct customer feedback represents an opportunity to improve the partnership. If feedback makes it into an MVP solution, celebrate that with the customer. Even if some feedback isn't actionable, simply being transparent with the decision to deprioritize the feedback demonstrates a growth mindset and a focus on continuous learning.

Azure DevOps includes ways to request, provide, and manage feedback. Each of these tools centralizes feedback so that the team can take action and provide follow-up in service of a transparent feedback loop.

Continuous integration

Continuous integration is the automating of code multiple times per day to have an updated single project. As adoptions scale and a hypothesis gets closer to true innovation at scale, the number of smaller hypotheses to be tested tends to grow rapidly. For accurate feedback loops and smooth adoption processes, it's important that each of those hypotheses is integrated and supportive of the primary hypothesis behind the innovation. This requires you to move quickly to innovate and grow, which requires multiple developers for testing variations of the core hypothesis. For later stage development efforts, you might even need multiple teams of developers, each building toward a shared solution. Continuous integration is the first step toward management of all the moving parts.

In continuous integration, code changes are frequently merged into the main branch. Automated build and test

processes make sure that code in the main branch is always production quality. This ensures that developers are working together to develop shared solutions that provide accurate and reliable feedback loops.

Azure DevOps and Azure Pipelines provide continuous integration capabilities with just a few steps in GitHub or other repositories. For more information, see What is continuous integration? or try the continuous integration hands-on lab. Solution architectures are available that can accelerate creation of your CI/CD pipelines via Azure DevOps.

Reliable testing

Defects in any solution can create false positives or false negatives. Unexpected errors can easily lead to misinterpretation of user adoption metrics. They can also generate negative feedback from customers that doesn't accurately represent the test of your hypothesis.

During early iterations of an MVP solution, defects are expected. Early adopters might even find them endearing. In early releases, acceptance testing is typically nonexistent. However, one aspect of building with empathy concerns the validation of the need and hypothesis. Both can be completed through unit tests at a code level and manual acceptance tests before deployment. Together, these provide some means of reliability in testing. You should try to automate a well-defined series of build, unit, and acceptance tests. These will ensure reliable metrics related to finer tweaks to the hypothesis and the resulting solution.

The Azure Test Plans feature provides tooling to develop and operate test plans during manual or automated test execution.

Solution deployment

Perhaps the most meaningful aspect of empowering adoption is your ability to control the release of a solution to customers. By providing a self-service or automated pipeline for releasing a solution to customers, you'll accelerate the feedback loop. By allowing customers to quickly interact with changes in the solution, you invite them into the process. This approach also triggers quicker testing of hypotheses, reducing assumptions and potential rework.

There are several methods for solution deployment. The three most common are:

- **Continuous deployment** is the most advanced method, as it automatically deploys code changes into production. For mature teams that are testing mature hypotheses, continuous deployment can be extremely valuable.
- During early stages of development, **continuous delivery** might be more appropriate. In continuous delivery, any code changes are automatically deployed to a production-like environment. Developers, business decision-makers, and others on the team can use this environment to verify that their work is production-ready. You can also use this method to test a hypothesis with customers without affecting ongoing business activities.
- **Manual deployment** is the least sophisticated approach to release management. As the name suggests, someone on the team manually deploys the most recent code changes. This approach is error prone, unreliable, and considered an antipattern by most seasoned engineers.

During the first iteration of an MVP solution, manual deployment is common, despite the preceding assessment. When the solution is extremely fluid and customer feedback is unknown, there's a significant risk in resetting the entire solution (or even the core hypothesis). Here's the general rule for manual deployment: no customer proof, no deployment automation.

Investing early can lead to lost time. More importantly, it can create dependencies on the release pipeline that make the team more resistant to an early pivot. After the first few iterations or when customer feedback suggests potential success, a more advanced model of deployment should be quickly adopted.

At any stage of hypothesis validation, Azure DevOps and Azure Pipelines provide continuous delivery and

continuous deployment capabilities. Learn more about continuous delivery, or check out the hands-on lab. Solution architecture can also accelerate creation of your CI/CD pipelines through Azure DevOps.

Integrated measurements

When you measure for customer impact, it's important to understand how customers react to changes in the solution. This data, known as *telemetry*, provides insights into the actions a user (or cohort of users) took when working with the solution. From this data, it's easy to get a quantitative validation of the hypothesis. Those metrics can then be used to adjust the solution and generate more fine-grained hypotheses. Those subtler changes help mature the initial solution in later iterations, ultimately driving to repeat adoption at scale.

In Azure, Azure Monitor provides the tools and interface to collect and review data from customer experiences. You can apply those observations and insights to refine the backlog by using Azure Boards.

Next steps

After you've gained an understanding of the CI/CD pipeline tools and processes needed to empower adoption, it's time to examine a more advanced innovation discipline: interact with devices. This discipline can help reduce the barriers between physical and digital experiences, making your solution even easier to adopt.

Interact with devices

Ambient user experiences: Interact with devices

5/18/2021 • 9 minutes to read • Edit Online

In Build with customer empathy, we discussed the three tests of true innovation: solve a customer need, keep the customer coming back, and scale across a base of customer cohorts. Each test of your hypothesis requires effort and iterations on the approach to adoption. This article offers insights on advanced approaches to reduce that effort through *ambient user experiences*. By interacting with devices, instead of an application, the customer might be more likely to use your solution first.

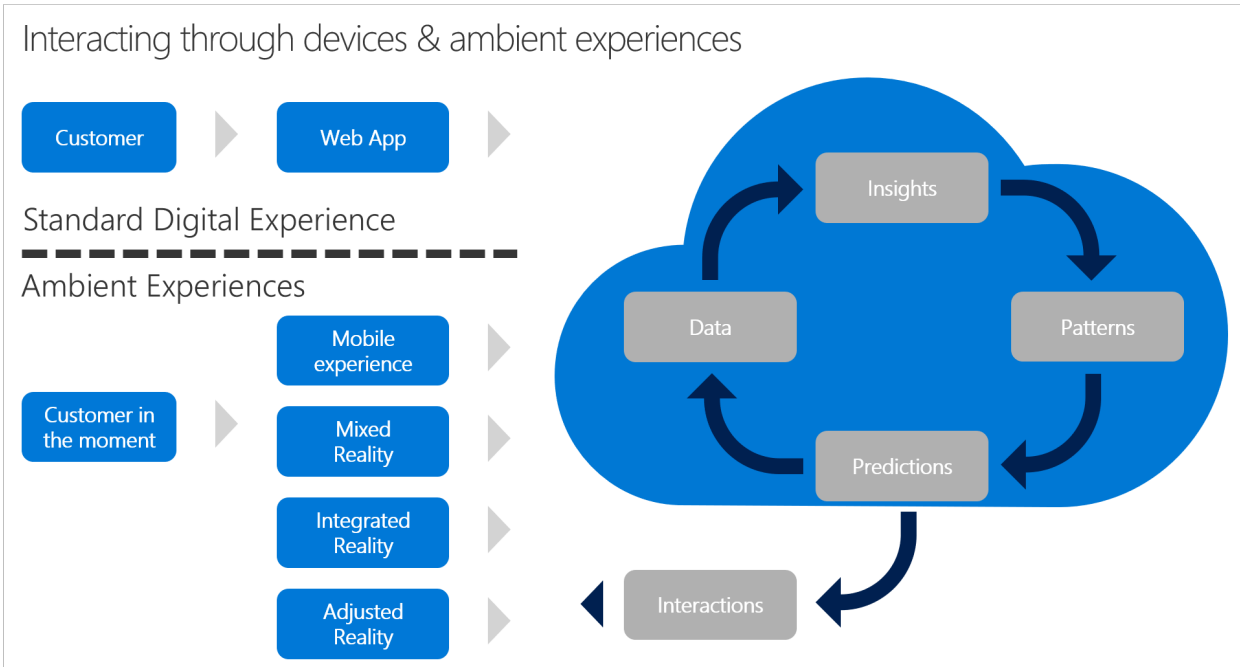
Ambient user experiences

An ambient user experience is a digital experience that relates to the immediate surroundings. Ambient user experiences occur when technology systems seamlessly interact with a user based on their needs and the context of their requests. A solution that features ambient user experiences tries to meet the customer in their moment of need. When possible, the solution meets the customer need without leaving the flow of activity that triggered it.

Life in the digital economy is full of distractions. We're all bombarded with social, email, web, visual, and verbal messaging, each of which is a risk of distraction. This risk increases with every second that passes between the customer's point of need and the moment they find a solution. Countless customers are lost in that brief time gap. To foster an increase in repeat adoption, you must reduce the number of distractions by reducing the time to solution.

Interact with devices

A standard web experience is the most common application development technique used to meet a customer's needs. This approach assumes that the customer is in front of their computer. If your customer consistently meets their point of need while in front of their laptop, build a web application. That application will provide an ambient user experience for that customer in that scenario. However, we know that this scenario is less and less likely.



Ambient user experiences typically require more than a web application these days. Through measurement and

learning with the customer, the behavior that triggers the customer's need can be observed, tracked, and used to build a more ambient and digital experience. The following list summarizes a few approaches to integration of ambient solutions into your hypotheses, with more details about each in the following paragraphs.

Types of interactive devices for ambient user experiences:

- Mobile experience: Mobile apps are ubiquitous in customer environments. In some situations, this might provide a sufficient level of interactivity to make a solution ambient.
- Mixed reality: Sometimes a customer's typical surroundings must be altered to make an interaction ambient. This factor creates something of a false reality in which the customer interacts with the solution and has a need met. In this case, the solution is ambient within the false reality.
- Integrated reality: Moving closer to true ambience, integrated reality solutions focus on the use of a device that exists within the customer's reality to integrate the solution into their natural behaviors. A Virtual Assistant is a great example of integrating reality into the surrounding environment. A less well-known option concerns Internet of Things (IoT) technologies, which integrate devices that already exist in the customer's surroundings.
- Adjusted reality: When any of these ambient solutions use predictive analysis in the cloud to define and provide an interaction with the customer through the natural surroundings, the solution has adjusted reality.

Understanding the customer need and measuring customer impact help you determine whether a device interaction or ambient user experience are necessary to validate your hypothesis. With each of those data points, the following sections will help you find the best digital experience solution.

Mobile experience

In the first stage of ambient user experience, the user moves away from the computer. Today's consumers and business professionals move fluidly between mobile and PC devices. Each of the platforms or devices used by your customer creates a new potential experience. Adding a mobile experience that extends the primary solution is the fastest way to improve integration into the customer's immediate surroundings. While a mobile device is far from ambient, it might edge closer to the customer's point of need.

When customers are mobile and change locations frequently, that may represent the most relevant form of ambient and digital experience for a particular solution. Over the past decade, innovation has frequently been triggered by the integration of existing solutions with a mobile experience.

Azure App Service is a great example of this approach. During early iterations, the web app feature of Azure App Service can be used to test the hypothesis. As the hypotheses become more complex, the mobile app feature of Azure App Service can extend the web app to run in a variety of mobile platforms.

Mixed reality

Mixed reality solutions represent the next step for ambient user experiences. This approach augments or replicates the customer's surroundings; it creates an extension of reality for the customer to operate within.

> **IMPORTANT**
> If a virtual reality (VR) device is required and it's not already part of a customer's immediate surroundings or natural behaviors, augmented or virtual reality is more of an alternative digital experience and less of an ambient experience.

Mixed reality experiences are increasingly common among remote workforces. Their use is growing even faster in industries that require collaboration or specialty skills that aren't readily available in the local market. Situations that require centralized implementation support of a complex product for a remote labor force are particularly fertile ground for augmented reality. In these scenarios, the central support team and remote employees might use augmented reality to work on, troubleshoot, and install the product.

For example, consider the case of spatial anchors. Spatial anchors allow you to create mixed reality experiences with objects that persist in their respective locations across devices over time. Through spatial anchors, a specific behavior can be captured, recorded, and persisted, providing an ambient experience the next time the user operates within that augmented environment. Azure Spatial Anchors is a service that moves this logic to the cloud, allowing digital experiences to be shared across interactive devices and even across solutions.

Integrated reality

Beyond mobile reality or even mixed reality lies integrated reality. Integrated reality aims to remove the digital experience entirely. All around us are devices with compute and connectivity capabilities. These devices can be used to collect data from the immediate surroundings without the customer having to ever touch a phone, laptop, or VR device.

This digital experience is best when some form of device is consistently within the same surroundings in which the customer need occurs. Common scenarios include factory floors, elevators, and even your car. These types of large devices already contain compute power. You can also use data from the device itself to detect customer behaviors and send those behaviors to the cloud. This automatic capture of customer behavior data dramatically reduces the need for a customer to input data. Additionally, the web, mobile, or VR experience can function as a feedback loop to share what's been learned from the integrated reality solution.

Examples of integrated reality in Azure could include:

- Azure Internet of Things (IoT) solutions: A collection of services in Azure that each aid in managing devices and the flow of data from those devices into the cloud and back out to end users.
- Azure Sphere: A combination of hardware and software that provides an intrinsically secure way to enable an existing device to securely transmit data between the device and Azure IoT solutions.
- Azure Kinect DK, AI sensors with advanced computer vision and speech models. These sensors can collect visual and audio data from the immediate surroundings and feed those inputs into your solution.

You can use all three of these digital experience tools to collect data from the natural surroundings and at the point of customer need. From there, your solution can respond to those data inputs to solve the need, sometimes before the customer is even aware that a trigger for that need has occurred.

Adjusted reality

The highest form of ambient user experience is adjusted reality, often referred to as *ambient intelligence*. Ambient intelligence refers to any electronic or digital environment that responds to the presence of people and adjusts automatically. Adjusted reality is an approach to using information from your solution to change the customer's reality without requiring them to interact directly with an application. In this approach, the application you initially built to prove your hypothesis might no longer be relevant at all. Instead, devices in the environment help modulate the inputs and outputs to meet customer needs.

Virtual assistants and smart speakers offer great examples of adjusted reality. Alone, a smart speaker is an example of simple integrated reality. But add a smart light and motion sensor to a smart speaker solution and it's easy to create a basic solution that turns on the lights when you enter a room.

Factory floors around the world provide additional examples of adjusted reality. During early stages of integrated reality, sensors on devices detected conditions like overheating, and then alerted a human being through an application. In adjusted reality, the customer might still be involved, but the feedback loop is tighter. On an adjusted reality factory floor, one device might detect overheating in a vital machine somewhere along the assembly line. Somewhere else on the floor, a second device then slows production slightly to allow the machine to cool and then resume full pace when the condition is resolved. In this situation, the customer is a second-hand participant. The customer uses your application to set the rules and understand how those rules have affected production, but they're not necessary to the feedback loop.

The Azure services described in Azure Internet of Things (IoT) solutions, Azure Sphere, and Azure Kinect DK can all be components of an adjusted reality solution. Your original application and business logic would then serve as the intermediary between the environmental input and the change that should be made in the physical environment.

A digital twin is another example of adjusted reality. This term refers to a digital representation of a physical device, presented through computer, mobile, or mixed-reality formats. Unlike less sophisticated 3D models, a digital twin reflects data collected from an actual device in the physical environment. This solution allows the user to interact with the digital representation in ways that could never be done in the real world. In this approach, physical devices adjust a mixed reality environment. However, the solution still gathers data from an integrated reality solution and uses that data to shape the reality of the customer's current surroundings.

In Azure, digital twins are created and accessed through a service called Azure Digital Twins.

Next steps

Now that you have a deeper understanding of device interactions and the ambient user experience or ambient intelligence tools that are right for your solution, you're ready to explore the final discipline of innovation, predict and influence.

Predict and influence

Predictive modeling and influencing customer behavior

5/20/2021 • 6 minutes to read • Edit Online

There are two classes of applications in the digital economy: *historical* and *predictive*. Many customer needs can be met solely by using historical data, including nearly real-time data. Most solutions focus primarily on aggregating data in the moment. They then process and share that data back to the customer in the form of a digital or ambient experience.

In contrast to historical modeling is predictive modeling. But, what is predictive modeling? Predictive modeling uses statistics and known results to process and create models that can be used to predict future outcomes, within reason. As predictive modeling becomes more cost-effective and readily available, customers demand forward-thinking experiences that lead to better decisions and actions. However, that demand doesn't always suggest a predictive solution. In most cases, a historical view can provide enough data to empower the customer to make a decision on their own.

Unfortunately, customers often take a myopic view that leads to decisions based on their immediate surroundings and sphere of influence. As options and decisions grow in number and impact, that myopic view may not serve the customer's needs. At the same time, as a hypothesis is proven at scale, the company providing the solution can see across thousands or millions of customer decisions. This big-picture approach makes it possible to see broad patterns and the impacts of those patterns. Predictive modeling capability is a wise investment when an understanding of those patterns is necessary to make decisions that best serve the customer.

Examples of predictive modeling and how it influences customer behavior

Various applications and ambient experiences use data to make predictions:

- **E-commerce:** Based on what other similar consumers have purchased, an e-commerce website suggests products that may be worth adding to your cart.
- **Adjusted reality:** IoT offers more advanced instances of predictive functionality. For example, suppose a device on an assembly line detects a rise in a machine's temperature. A cloud-based predictive model determines how to respond. Based on that prediction, another device slows down the assembly line until the machine can cool.
- **Consumer products:** Cell phones, smart homes, even your car, all use predictive capabilities, which they analyze to suggest user behavior based on factors like location or time of day. When a prediction and the initial hypothesis are aligned, the prediction leads to action. At a very mature stage, this alignment can make products like a self-driving car a reality.

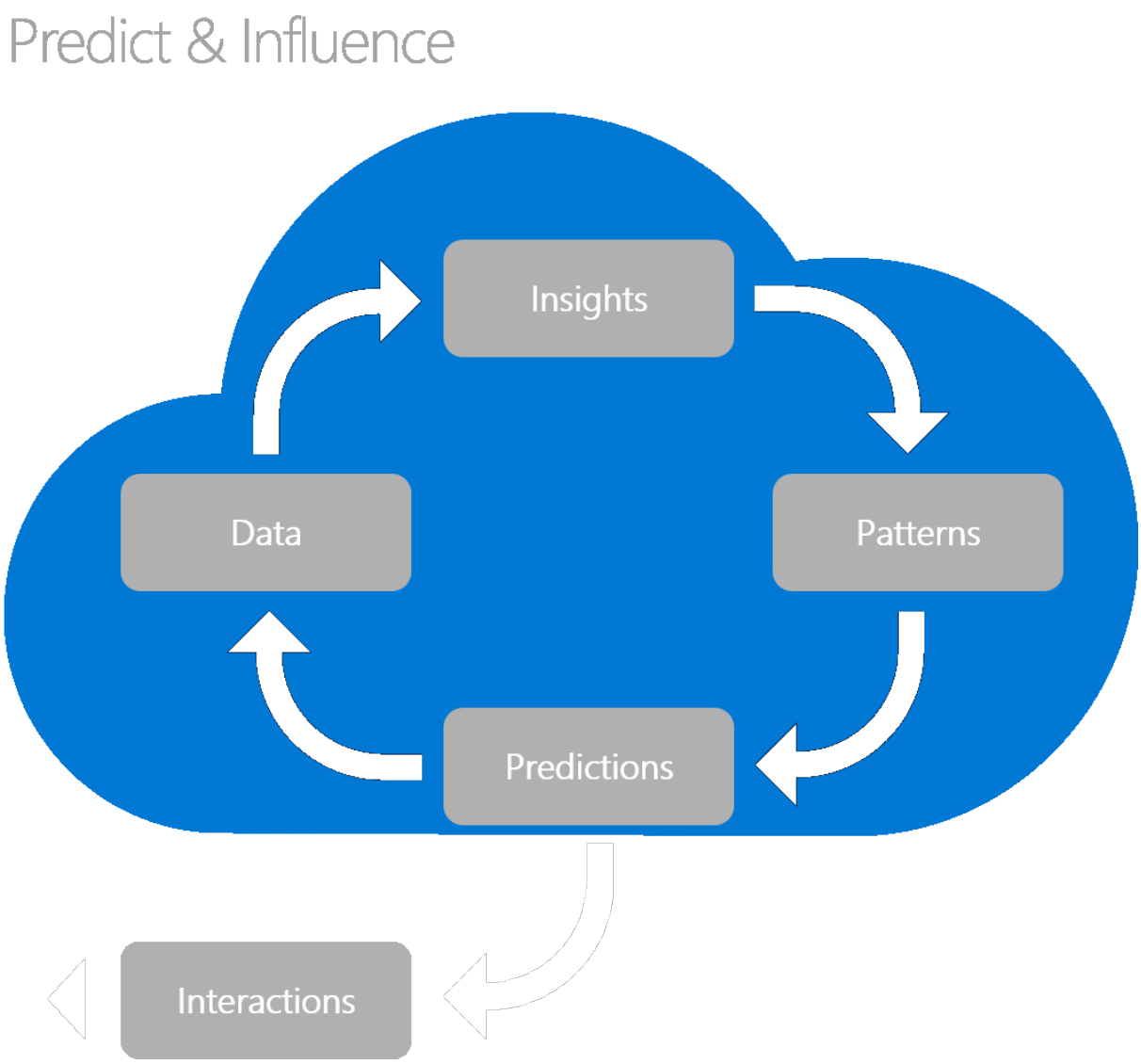
Develop predictive capabilities

Solutions that consistently provide accurate predictive capabilities commonly include five core characteristics. The five core predictive modeling characteristics are:

- Data
- Insights
- Patterns
- Predictions

- Interactions

Each aspect is required to develop predictive capabilities. Like all great innovations, the development of predictive capabilities requires a commitment to iteration. In each iteration, one or more of the following characteristics is matured to validate increasingly complex customer hypotheses.



Caution

If the customer hypothesis developed in Build with customer empathy includes predictive capabilities, the principles described there might well apply. However, predictive capabilities require significant investment of time and energy. When predictive capabilities are technical spikes, as opposed to a source of real customer value, we suggest that you delay predictions until the customer hypotheses have been validated at scale.

Data

Data is the most elemental of the characteristics mentioned earlier. Each of the disciplines for developing digital inventions generates data. That data, of course, contributes to the development of predictions. For more information on ways to get data into a predictive solution, see Democratize data with digital invention and Interact with devices.

Various data sources can be used to deliver predictive capabilities:

Insights

Subject matter experts use data about customer needs and behaviors to develop basic business insights from a study of raw data. Those insights can pinpoint occurrences of the desired customer behaviors (or, alternatively,

undesirable results). During iterations on the predictions, these insights can aid in identifying potential correlations that could ultimately generate positive outcomes. For guidance on enabling subject matter experts to develop insights, see Democratize data with digital invention.

Patterns

People have always tried to detect patterns in large volumes of data. Computers were designed for that purpose. Machine learning accelerates that quest by detecting precisely such patterns, a skill that comprises the machine learning model. Those patterns are then applied through machine learning algorithms to predict outcomes when a new set of data is entered into the algorithms.

Using insights as a starting point, machine learning develops and applies predictive models to capitalize on the patterns in data. Through multiple iterations of training, testing, and adoption, those models and algorithms can accurately predict future outcomes.

Azure Machine Learning is the cloud-native service in Azure for building and training models based on your data. This tool also includes a workflow for accelerating the development of machine learning algorithms. This workflow can be used to develop algorithms through a visual interface or Python.

For more robust machine learning models, ML Services in Azure HDInsight provides a machine learning platform built on Apache Hadoop clusters. This approach enables more granular control of the underlying clusters, storage, and compute nodes. Azure HDInsight also offers more advanced integration through tools like ScaleR and SparkR to create predictions based on integrated and ingested data, even working with data from a stream. The flight delay prediction solution demonstrates each of these advanced capabilities when used to predict flight delays based on weather conditions. The HDInsight solution also allows for enterprise controls, such as data security, network access, and performance monitoring to operationalize patterns.

Predictions

After a pattern is built and trained, you can apply it through APIs, which can make predictions during the delivery of a digital experience. Most of these APIs are built from a well-trained model based on a pattern in your data. As more customers deploy everyday workloads to the cloud, the prediction APIs used by cloud providers lead to ever-faster adoption.

Azure Cognitive Services is an example of a predictive API built by a cloud vendor. This service includes predictive APIs for content moderation, anomaly detection, and suggestions to personalize content. These APIs are ready to use and are based on well-known content patterns, which Microsoft has used to train models. Each of those APIs makes predictions based on the data you feed into the API.

Azure Machine Learning lets you deploy custom-built algorithms, which you can create and train based solely on your own data. For information about deploying predictions with Azure Machine Learning, see Deploy machine learning models to Azure.

For information about the processes for exposing predictions developed for ML Services on Azure HDInsight, see Set up HDInsight clusters.

Interactions

After a prediction is made available through an API, you can use it to influence customer behavior. That influence takes the form of interactions. An interaction with a machine learning algorithm happens within your other digital or ambient experiences. As data is collected through the application or experience, it's run through the machine learning algorithms. When the algorithm predicts an outcome, that prediction can be shared back with the customer through the existing experience.

Learn more about how to create an ambient experience through an adjusted reality solution.

Next steps

Review a prescriptive framework that includes the tools, programs, and content (best practices, configuration templates, and architecture guidance) to simplify adoption for the following innovation scenarios.

- Kubernetes in the Cloud Adoption Framework
- AI in the Cloud Adoption Framework

Governance in the Microsoft Cloud Adoption Framework for Azure

8/25/2021 • 3 minutes to read • Edit Online

The cloud creates new paradigms for the technologies that support the business. These new paradigms also change how those technologies are adopted, managed, and governed. When entire datacenters can be torn down virtually and rebuilt with one line of code executed by an unattended process, we have to rethink traditional approaches. This is especially true for governance.

Cloud governance is an iterative process. For organizations with existing policies that govern on-premises IT environments, cloud governance should complement those policies. The level of corporate policy integration between on-premises and the cloud varies depending on cloud governance maturity and a digital estate in the cloud. As the cloud estate changes over time, so do cloud governance processes and policies. The following exercises help you start building your initial governance foundation.

1. Methodology: Establish a basic understanding of the methodology that drives cloud governance in the Cloud Adoption Framework to begin thinking through the end state solution.
2. Benchmark: Assess your current state and future state to establish a vision for applying the framework.
3. Initial governance foundation: Begin your governance journey with a small, easily implemented set of governance tools. This initial governance foundation is called a minimum viable product (MVP).
4. Improve the initial governance foundation: Throughout implementation of the cloud adoption plan, iteratively add governance controls to address tangible risks as you progress toward the end state.

Objective of this content

The guidance in this section of the Cloud Adoption Framework serves two purposes:

- Provides examples of actionable governance guides that represent common experiences often encountered by customers. Each example encapsulates business risks, corporate policies for risk mitigation, and design guidance for implementing technical solutions. By necessity, the design guidance is specific to Azure. All other content in these guides could be applied in a cloud-agnostic or multicloud approach.
- Helps you create personalized governance solutions that meet various business needs. These needs include the governance of multiple public clouds through detailed guidance on the development of corporate policies, processes, and tooling.

This content is intended for use by the cloud governance team. It's also relevant to cloud architects who need to develop a strong foundation in cloud governance.

Intended audience

The content in the Cloud Adoption Framework affects the business, technology, and culture of enterprises. This section of the Cloud Adoption Framework interacts heavily with IT security, IT governance, finance, line-of-business leaders, networking, identity, and cloud adoption teams. Various dependencies on these personnel require a facilitative approach by the cloud architects using this guidance. Facilitation with these teams might be a one-time effort. In some cases, interactions with these other personnel will be ongoing.

The cloud architect serves as the thought leader and facilitator to bring these audiences together. The content in

this collection of guides is designed to help the cloud architect facilitate the right conversation, with the right audience, to drive necessary decisions. Business transformation that's empowered by the cloud depends on the cloud architect to help guide decisions throughout the business and IT.

Cloud architect specialization in this section: Each section of the Cloud Adoption Framework represents a different specialization or variant of the cloud architect role. This section of the Cloud Adoption Framework is designed for cloud architects with a passion for mitigating or reducing technical risks. Some cloud providers refer to these specialists as *cloud custodians*, but we prefer *cloud guardians* or, collectively, the *cloud governance team*. The actionable governance guides show how the composition and role of the cloud governance team might change over time.

Use this guide

Reading the Govern methodology content from start to finish will help you develop a robust cloud governance strategy in parallel with cloud implementation. The guidance walks you through the theory and implementation of this strategy.

For a crash course on the theory and quick access to Azure implementation, get started with the governance guides overview. Using this guidance, you can start small and iteratively improve your governance needs in parallel with cloud adoption efforts.

Govern methodology for the cloud

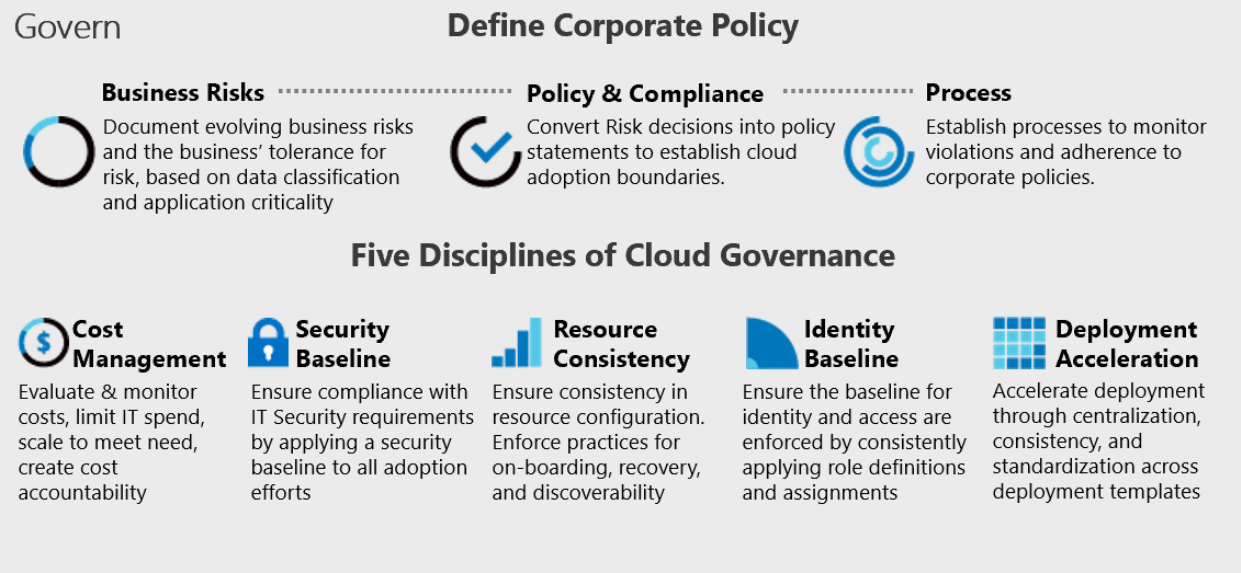
3/6/2021 • 3 minutes to read • Edit Online

Adopting the cloud is a journey, not a destination. Along the way, there are clear milestones and tangible business benefits. The final state of cloud adoption is unknown when a company begins the journey. Cloud governance creates guardrails that keep the company on a safe path throughout the journey.

The Cloud Adoption Framework provides governance guides that describe the experiences of fictional companies that are based on the experiences of real customers. Each guide follows the customer through the governance aspects of their cloud adoption.

Envision an end state

A journey without a target destination is just wandering. It's important to establish a rough vision of the end state before taking the first step. The following infographic provides a frame of reference for the end state. It's not your starting point, but it shows your potential destination.



The Cloud Adoption Framework governance model identifies key areas of importance during the journey. Each area relates to different types of risks the company must address as it adopts more cloud services. Within this framework, the governance guide identifies required actions for the cloud governance team. Along the way, each principle of the Cloud Adoption Framework governance model is described further. Broadly, these include:

Corporate policies: Corporate policies drive cloud governance. The governance guide focuses on specific aspects of corporate policy:

- **Business risks:** Identifying and understanding corporate risks.
- **Policy and compliance:** Converting risks into policy statements that support any compliance requirements.
- **Processes:** Ensuring adherence to the stated policies.

Five Disciplines of Cloud Governance: These disciplines support the corporate policies. Each discipline protects the company from potential pitfalls:

- Cost Management discipline
- Security Baseline discipline
- Resource Consistency discipline

- Identity Baseline discipline
- Deployment Acceleration discipline

Essentially, corporate policies serve as the early warning system to detect potential problems. The disciplines help the company manage risks and create guardrails.

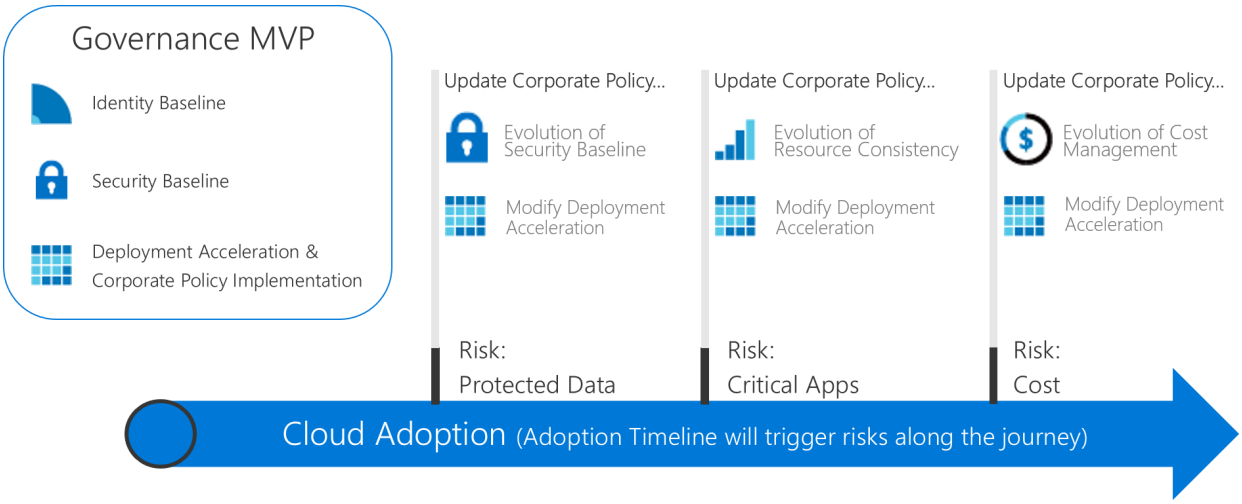
Grow to the end state

Because governance requirements will change throughout the cloud adoption journey, a different approach to governance is required. Companies can no longer wait for a small team to build guardrails and roadmaps on every highway **before taking the first step**. Business results are expected more quickly and smoothly. IT governance must also move quickly and keep pace with business demands to stay relevant during cloud adoption and avoid "shadow IT."

An *incremental governance* approach empowers these traits. Incremental governance relies on a small set of corporate policies, processes, and tools to establish a foundation for adoption and governance. That foundation is called a *minimum viable product (MVP)*. An MVP allows the governance team to quickly incorporate governance into implementations throughout the adoption lifecycle. An MVP can be established at any point during the cloud adoption process. It's a good practice to adopt an MVP as early as possible.

The ability to respond rapidly to changing risks empowers the cloud governance team to engage in new ways. The cloud governance team can join the cloud strategy team as scouts, moving ahead of the cloud adoption teams, plotting routes, and quickly establishing guardrails to manage risks associated with the adoption plans. These just-in-time governance layers are known as *governance iterations*. With this approach, governance strategy grows one step ahead of the cloud adoption teams.

The following diagram shows a simple governance MVP and three governance iterations. During the iterations, additional corporate policies are defined to remediate new risks. The Deployment Acceleration discipline then applies those changes across each deployment.



> **NOTE**
> Governance is not a replacement for key functions such as security, networking, identity, finance, DevOps, or operations. Along the way, there will be interactions with and dependencies on members from each function. Those members should be included on the cloud governance team to accelerate decisions and actions.

Next steps

Learn to use the Cloud Adoption Framework governance benchmark tool to assess your transformation journey and help you identify gaps in your organization across six key domains as defined in the framework.

Assess your transformation journey

Assess your transformation journey

3/6/2021 • 2 minutes to read • Edit Online

The Cloud Adoption Framework provides a governance benchmark tool to help you identify gaps in your organization across six key domains as defined in the framework.

Governance benchmark tool

Receive a personalized report that outlines the difference between your current state and business priorities, along with tailored resources to help you get started. Assess your current state and future state to establish a vision for applying the framework.

Use the governance benchmark tool

Next steps

Begin your governance journey with a small, easily implemented set of governance tools. This initial governance foundation is called a minimum viable product (MVP).

Establish an initial governance foundation

Establish an initial cloud governance foundation

3/6/2021 • 2 minutes to read • Edit Online

Establishing cloud governance is a broad iterative effort. It's challenging to strike an effective balance between speed and control, especially during execution of early methodologies within the cloud adoption. The governance guidance in the Cloud Adoption Framework helps provide that balance via an agile approach to adoption.

This article provides two options for establishing an initial foundation for governance. Either option ensures that governance constraints can be scaled and expanded as the adoption plan is implemented and requirements become more clearly defined. By default, the initial foundation assumes an isolate-and-control position. It also focuses more on resource organization than on resource governance. This lightweight starting point is called a *minimum viable product (MVP)* for governance. The objective of the MVP is reducing barriers to establishing an initial governance position, and then enabling rapid maturation of the solution to address a variety of tangible risks.

Already using the Cloud Adoption Framework

If you have been following along with the Cloud Adoption Framework, you may have already deployed a governance MVP. Governance is a core aspect of any operating model. It's present in every methodology of the cloud adoption lifecycle. As such, the Cloud Adoption Framework provides guidance that injects governance into activities related to the implementation of your cloud adoption plan. One example of this governance integration is using blueprints to deploy one or more landing zones present in the Ready methodology guidance. Another example is guidance for scaling out subscriptions. If you have followed either of those recommendations, then the following MVP sections are simply a review of your existing deployment decisions. After a quick review, jump ahead to mature the initial governance solution and apply best-practice controls.

Establish an initial governance foundation

The following are two different examples of initial governance foundations (also called governance MVPs) to apply a sound foundation for governance to new or existing deployments. Choose the MVP that best aligns with your business needs to get started:

- Standard governance guide: A guide for most organizations based on the recommended initial two-subscription model, designed for deployments in multiple regions but not spanning public and sovereign/government clouds.
- Governance guide for complex enterprises: A guide for enterprises that are managed by multiple independent IT business units or span public and sovereign/government clouds.

Next steps

Once a governance foundation is in place, apply suitable recommendations to improve the solution and protect against tangible risks.

Improve the initial governance foundation

Improve your initial cloud governance foundation

3/6/2021 • 2 minutes to read • Edit Online

This article assumes that you have established an initial cloud governance foundation. As your cloud adoption plan is implemented, tangible risks will emerge from the proposed approaches by which teams want to adopt the cloud. As these risks surface in release planning conversations, use the following grid to quickly identify a few best practices for getting ahead of the adoption plan to prevent risks from becoming real threats.

Maturity vectors

At any time, the following best practices can be applied to the initial governance foundation to address the risk or need mentioned in the following table.

> **IMPORTANT**
>
> Resource organization can affect how these best practices are applied. It's important to start with the recommendations that best align with the initial cloud governance foundation you implemented in the previous step.

RISK/NEED	STANDARD ENTERPRISE	COMPLEX ENTERPRISE
Sensitive data in the cloud	Discipline improvement	Discipline improvement
Mission-critical applications in the cloud	Discipline improvement	Discipline improvement
Cloud cost management	Discipline improvement	Discipline improvement
Multicloud	Discipline improvement	Discipline improvement
Complex/legacy identity management	N/A	Discipline improvement
Multiple layers of governance	N/A	Discipline improvement

Next steps

In addition to the application of best practices, the Govern methodology of the Cloud Adoption Framework can be customized to fit unique business constraints. After following the applicable recommendations, evaluate corporate policy to understand additional customization requirements.

Evaluate corporate policy

Cloud governance guides

3/6/2021 • 5 minutes to read • Edit Online

The actionable governance guides in this section illustrate the incremental approach of the Cloud Adoption Framework governance model, based on the Govern methodology previously described. You can establish an agile approach to cloud governance that will grow to meet the needs of any cloud governance scenario.

Review and adopt cloud governance best practices

To begin your cloud adoption journey, choose one of the following governance guides. Each guide outlines a set of best practices, based on a set of fictional customer experiences. For readers who are new to the incremental approach of the Cloud Adoption Framework governance model, review the high-level introduction to governance theory below before adopting either set of best practices.

- Standard governance guide: A guide for most organizations based on the recommended two-subscription model, designed for deployments in multiple regions but not spanning public and sovereign/government clouds.

Standard governance guide

- Governance guide for complex enterprises: A guide for enterprises that are managed by multiple independent IT business units or span public and sovereign/government clouds.

Governance guide for complex enterprises

An incremental approach to cloud governance

Choose a governance guide

The guides demonstrate how to implement a governance MVP. From there, each guide shows how the cloud governance team can work ahead of the cloud adoption teams as a partner to accelerate adoption efforts. The Cloud Adoption Framework governance model guides the application of governance from foundation through subsequent improvements and evolutions.

To begin a governance journey, choose one of the two options below. The options are based on synthesized customer experiences. The titles are based on the complexity of the enterprise for ease of navigation. Your decision may be more complex. The following tables outline the differences between the two options.

WARNING

A more robust governance starting point may be required. In such cases, consider the CAF enterprise-scale landing zone. This approach focuses on adoption teams who have a mid-term objective (within 24 months) to host more than 1,000 assets (infrastructure, apps, or data) in the cloud. The CAF enterprise-scale landing zone is the typical choice for complex governance scenarios in large cloud adoption efforts.

NOTE

It's unlikely that either guide aligns entirely with your situation. Choose whichever guide is closest and use it as a starting point. Throughout the guide, additional information is provided to help you customize decisions to meet specific criteria.

Business characteristics

CHARACTERISTIC	STANDARD ORGANIZATION	COMPLEX ENTERPRISE
Geography (country or geopolitical region)	Customers or staff reside largely in one geography	Customers or staff reside in multiple geographies or require sovereign clouds.
Business units affected	Business units that share a common IT infrastructure	Multiple business units that do not share a common IT infrastructure.
IT budget	Single IT budget	Budget allocated across business units and currencies.
IT investments	Capital expense-driven investments are planned yearly and usually cover only basic maintenance.	Capital expense-driven investments are planned yearly and often include maintenance and a refresh cycle of three to five years.

Current state before adopting cloud governance

STATE	STANDARD ENTERPRISE	COMPLEX ENTERPRISE
Datacenter or third-party hosting providers	Fewer than five datacenters	More than five datacenters
Networking	No WAN, or 1 – 2 WAN providers	Complex network or global WAN
Identity	Single forest, single domain.	Complex, multiple forests, multiple domains.

Desired future state after incremental improvement of cloud governance

STATE	STANDARD ORGANIZATION	COMPLEX ENTERPRISE
Cost Management: cloud accounting	Showback model. Billing is centralized through IT.	Chargeback model. Billing could be distributed through IT procurement.
Security Baseline: protected data	Company financial data and IP. Limited customer data. No third-party compliance requirements.	Multiple collections of customers' financial and personal data. Might need to consider third-party compliance.

CAF enterprise-scale landing zone

CAF enterprise-scale landing zone is an approach to making the most of the Azure cloud platform's capabilities while respecting an enterprise's security and governance requirements.

Compared to traditional on-premises environments, Azure allows workload development teams and their business sponsors to take advantage of the increased deployment agility that cloud platforms offer. As your cloud adoption efforts expand to include mission-critical data and workloads, this agility may conflict with corporate security and policy compliance requirements established by your IT teams. This is especially true for large enterprises that have existing sophisticated governance and regulatory requirements.

The CAF enterprise-scale landing zone architecture aims to address these concerns earlier in the adoption lifecycle by architectures, implementations, and guidance to help achieve a balance between cloud adoption team requirements and central IT team requirements during enterprise cloud adoption efforts. Central to this

approach is the concept of a shared service architecture and well-managed landing zones.

CAF enterprise-scale landing zone deploys your own "isolated cloud" within the Azure platform, integrating management processes, regulatory requirements, and security processes required by your governance policies. Within this virtual boundary, CAF enterprise-scale landing zone offers example models for deploying workloads while ensuring consistent compliance and provides basic guidance on implementing an organization's separation of roles and responsibilities in the cloud.

CAF enterprise-scale landing zone qualifications

Although smaller teams may benefit from the architecture and recommendations the CAF enterprise-scale landing zone provides. Our objective is to continue to streamline the CAF enterprise-scale landing zone implementations to make them more friendly for smaller teams. Currently, this approach is designed to guide central IT teams managing large cloud environments.

The CAF enterprise-scale landing zone approach focuses on adoption teams who have a mid-term objective (within 24 months) to **host more than 1,000 assets (applications, infrastructure, or data assets) in the cloud**.

For organizations that meet the following criteria, you may also want to start with the CAF enterprise-scale landing zone:

- Your enterprise is subject to regulatory compliance requirements that require centralized monitoring and audit capabilities.
- You need to maintain common policy and governance compliance and centralized IT control over core services.
- Your industry depends on a complex platform that requires complex controls and deep domain expertise to govern the platform. This is most common in large enterprises within finance, manufacturing, and oil and gas.
- Your existing IT governance policies require tighter parity with existing features, even during early stage adoption.

Next steps

Choose one of these guides:

Standard enterprise governance guide

Governance guide for complex enterprises

Standard enterprise governance guide

3/6/2021 • 8 minutes to read • Edit Online

Overview of best practices

This governance guide follows the experiences of a fictional company through various stages of governance maturity. It's based on real customer experiences. The best practices are based on the constraints and needs of the fictional company.

As a quick starting point, this overview defines a minimum viable product (MVP) for governance based on best practices. It also provides links to some governance improvements that add further best practices as new business or technical risks emerge.

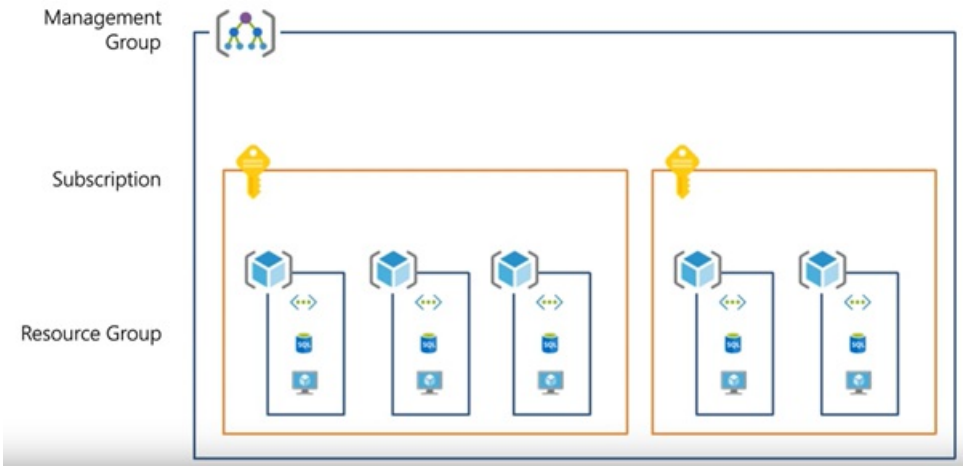
> **WARNING**
> This MVP is a baseline starting point, based on a set of assumptions. Even this minimal set of best practices is based on corporate policies that are driven by unique business risks and risk tolerances. To see whether these assumptions apply to you, read the longer narrative that follows this article.

Governance best practices

These best practices serve as a foundation for an organization to quickly and consistently add governance guardrails across your subscriptions.

Resource organization

The following diagram shows the governance MVP hierarchy for organizing resources.

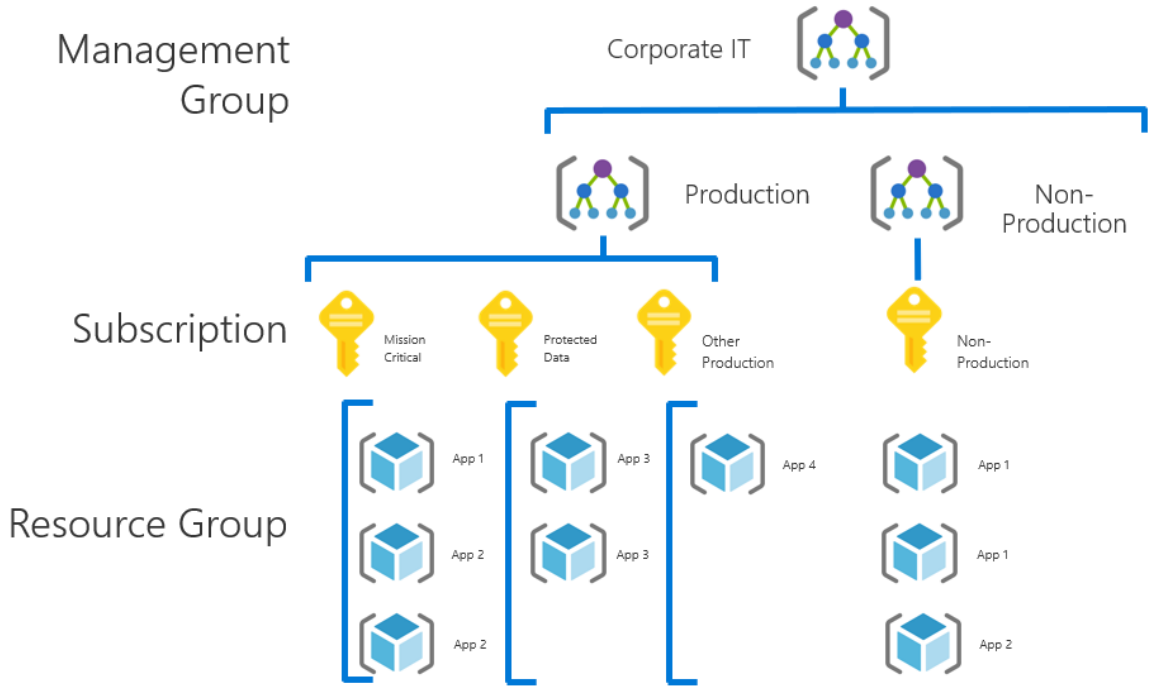


Every application should be deployed in the proper area of the management group, subscription, and resource group hierarchy. During deployment planning, the cloud governance team will create the necessary nodes in the hierarchy to empower the cloud adoption teams.

1. One management group for each type of environment (such as production, development, and test).
2. Two subscriptions, one for production workloads and another for nonproduction workloads.
3. Consistent nomenclature should be applied at each level of this grouping hierarchy.
4. Resource groups should be deployed in a manner that considers its contents lifecycle: everything that is developed together, is managed together, and retires together goes together. For more information about resource group best practices, see the resource consistency decision guide.
5. Region selection is incredibly important and must be considered so that networking, monitoring, auditing

can be in place for failover/failback as well as confirmation that needed SKUs are available in the preferred regions.

Here is an example of this pattern in use:



These patterns provide room for growth without complicating the hierarchy unnecessarily.

> **NOTE**
> In the event of changes to your business requirements, Azure management groups allow you to easily reorganize your management hierarchy and subscription group assignments. However, keep in mind that policy and role assignments applied to a management group are inherited by all subscriptions underneath that group in the hierarchy. If you plan to reassign subscriptions between management groups, make sure that you are aware of any policy and role assignment changes that may result. See the Azure management groups documentation for more information.

Governance of resources

A set of global policies and RBAC roles will provide a baseline level of governance enforcement. To meet the cloud governance team's policy requirements, implementing the governance MVP requires completing the following tasks:

1. Identify the Azure Policy definitions needed to enforce business requirements. This might include using built-in definitions and creating new custom definitions. To keep up with the pace of newly released built-in definitions, there's an atom feed of all the commits for built-in policies, which you can use for an RSS feed. Alternatively, you can check AzAdvertizer.
2. Create a blueprint definition using these built-in and custom policy and the role assignments required by the governance MVP.
3. Apply policies and configuration globally by assigning the blueprint definition to all subscriptions.

Identify policy definitions

Azure provides several built-in policies and role definitions that you can assign to any management group, subscription, or resource group. Many common governance requirements can be handled using built-in definitions. However, it's likely that you will also need to create custom policy definitions to handle your specific requirements.

Custom policy definitions are saved to either a management group or a subscription and are inherited through the management group hierarchy. If a policy definition's save location is a management group, that policy definition is available to assign to any of that group's child management groups or subscriptions.

Since the policies required to support the governance MVP are meant to apply to all current subscriptions, the following business requirements will be implemented using a combination of built-in definitions and custom definitions created in the root management group:

1. Restrict the list of available role assignments to a set of built-in Azure roles authorized by your cloud governance team. This requires a custom policy definition.
2. Require the following tags on all resources: *Department/Billing Unit*, *Geography*, *Data Classification*, *Criticality*, *SLA*, *Environment*, *Application Archetype*, *Application*, and *Application Owner*. This can be handled using the `Require specified tag` built-in definition.
3. Require that the `Application` tag for resources should match the name of the relevant resource group. This can be handled using the "Require tag and its value" built-in definition.

For information on defining custom policies see the Azure Policy documentation. For guidance and examples of custom policies, consult the Azure Policy samples site and the associated GitHub repository.

Assign Azure Policy and RBAC roles using Azure Blueprints

Azure policies can be assigned at the resource group, subscription, and management group level, and can be included in Azure Blueprints definitions. Although the policy requirements defined in this governance MVP apply to all current subscriptions, it's very likely that future deployments will require exceptions or alternative policies. As a result, assigning policy using management groups, with all child subscriptions inheriting these assignments, may not be flexible enough to support these scenarios.

Azure Blueprints allows consistent assignment of policy and roles, application of Resource Manager templates, and deployment of resource groups across multiple subscriptions. Like policy definitions, blueprint definitions are saved to management groups or subscriptions. The policy definitions are available through inheritance to any children in the management group hierarchy.

The cloud governance team has decided that enforcement of required Azure Policy and RBAC assignments across subscriptions will be implemented through Azure Blueprints and associated artifacts:

1. In the root management group, create a blueprint definition named `governance-baseline`.
2. Add the following blueprint artifacts to the blueprint definition:
   a. Policy assignments for the custom Azure Policy definitions defined at the management group root.
   b. Resource group definitions for any groups required in subscriptions created or governed by the Governance MVP.
   c. Standard role assignments required in subscriptions created or governed by the Governance MVP.
3. Publish the blueprint definition.
4. Assign the `governance-baseline` blueprint definition to all subscriptions.

See the Azure Blueprints documentation for more information on creating and using blueprint definitions.

Secure hybrid VNet

Specific subscriptions often require some level of access to on-premises resources. This is common in migration scenarios or dev scenarios where dependent resources reside in the on-premises datacenter.

Until trust in the cloud environment is fully established it's important to tightly control and monitor any allowed communication between the on-premises environment and cloud workloads, and that the on-premises network is secured against potential unauthorized access from cloud-based resources. To support these scenarios, the governance MVP adds the following best practices:

1. Establish a cloud secure hybrid VNet.

   a. The VPN reference architecture establishes a pattern and deployment model for creating a VPN Gateway in Azure.
   b. Validate that on-premises security and traffic management mechanisms treat connected cloud networks as untrusted. Resources and services hosted in the cloud should only have access to authorized on-premises services.
   c. Validate that the local edge device in the on-premises datacenter is compatible with Azure VPN Gateway requirements and is configured to access the public internet.
   d. Note that VPN tunnels should not be considered production ready circuits for anything but the most simple workloads. Anything beyond a few simple workloads requiring on-premises connectivity should use Azure ExpressRoute.
2. In the root management group, create a second blueprint definition named `secure-hybrid-vnet`.
   a. Add the Resource Manager template for the VPN Gateway as an artifact to the blueprint definition.
   b. Add the Resource Manager template for the virtual network as an artifact to the blueprint definition.
   c. Publish the blueprint definition.
3. Assign the `secure-hybrid-vnet` blueprint definition to any subscriptions requiring on-premises connectivity. This definition should be assigned in addition to the `governance-baseline` blueprint definition.

One of the biggest concerns raised by IT security and traditional governance teams is the risk that early stage cloud adoption will compromise existing assets. The above approach allows cloud adoption teams to build and migrate hybrid solutions, with reduced risk to on-premises assets. As trust in the cloud environment increases, later evolutions may remove this temporary solution.

> **NOTE**
>
> The above is a starting point to quickly create a baseline governance MVP. This is only the beginning of the governance journey. Further evolution will be needed as the company continues to adopt the cloud and takes on more risk in the following areas:
>
> - Mission-critical workloads
> - Protected data
> - Cost management
> - Multicloud scenarios
>
> Moreover, the specific details of this MVP are based on the example journey of a fictional company, described in the articles that follow. We highly recommend becoming familiar with the other articles in this series before implementing this best practice.

Iterative governance improvements

Once this MVP has been deployed, additional layers of governance can be incorporated into the environment quickly. Here are some ways to improve the MVP to meet specific business needs:

- Security baseline for protected data
- Resource configurations for mission-critical applications
- Controls for cost management
- Controls for multicloud evolution

What does this guidance provide?

In the MVP, practices and tools from the Deployment Acceleration discipline are established to quickly apply corporate policy. In particular, the MVP uses Azure Blueprints, Azure Policy, and Azure management groups to apply a few basic corporate policies, as defined in the narrative for this fictional company. Those corporate policies are applied using Resource Manager templates and Azure policies to establish a small baseline for

identity and security.



Incremental improvement of governance practices

Over time, this governance MVP will be used to improve governance practices. As adoption advances, business risk grows. Various disciplines within the Cloud Adoption Framework governance model will change to manage those risks. Later articles in this series discuss the incremental improvement of corporate policy affecting the fictional company. These improvements happen across three disciplines:

- The Cost Management discipline, as adoption scales.
- The Security Baseline discipline, as protected data is deployed.
- The Resource Consistency discipline, as IT operations begins supporting mission-critical workloads.



Next steps

Now that you're familiar with the governance MVP and have an idea of the governance improvements to follow, read the supporting narrative for additional context.

Read the supporting narrative

Standard enterprise governance guide: The narrative behind the governance strategy

3/6/2021 • 2 minutes to read • Edit Online

The following narrative describes the use case for governance during a standard enterprise's cloud adoption journey. Before implementing the journey, it's important to understand the assumptions and rationale that are reflected in this narrative. Then you can better align the governance strategy to your organization's journey.

Back story

The board of directors started the year with plans to energize the business in several ways. They're pushing leadership to improve customer experiences to gain market share. They're also pushing for new products and services that will position the company as a thought leader in the industry. They also initiated a parallel effort to reduce waste and cut unnecessary costs. Though intimidating, the actions of the board and leadership show that this effort is focusing as much capital as possible on future growth.

In the past, the company's CIO has been excluded from these strategic conversations. Because the future vision is intrinsically linked to technical growth, IT has a seat at the table to help guide these big plans and is now expected to deliver in new ways. The team isn't prepared for these changes and is likely to struggle with the learning curve.

Business characteristics

The company has the following business profile:

- All sales and operations reside in a single country, with a low percentage of global customers.
- The business operates as a single business unit, with budget aligned to functions, including sales, marketing, operations, and IT.
- The business views most of IT as a capital drain or a cost center.

Current state

Here is the current state of the company's IT and cloud operations:

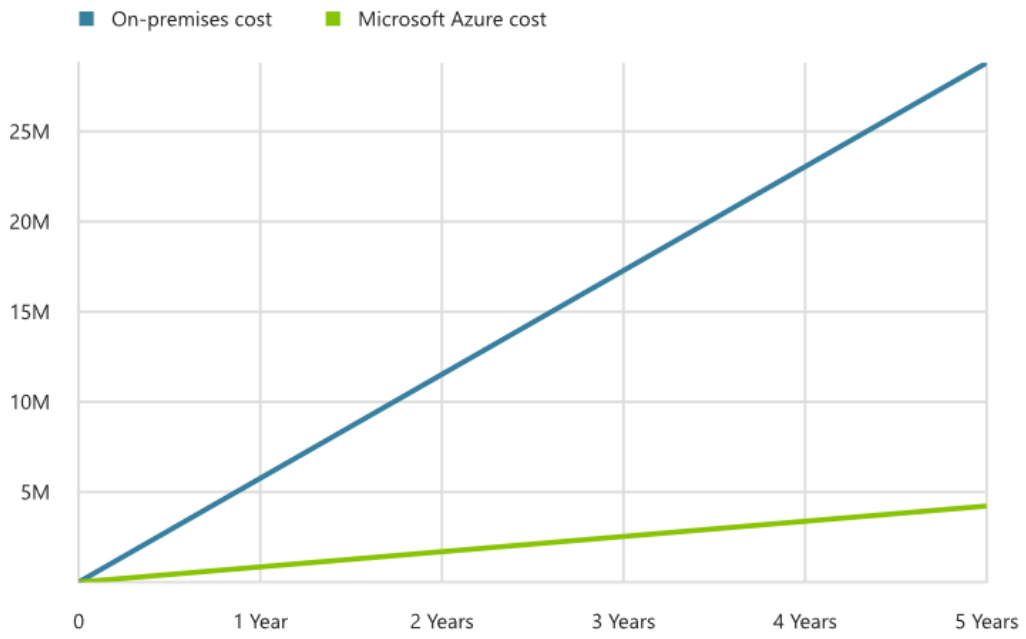
- IT operates two hosted infrastructure environments. One environment contains production assets. The second environment contains disaster recovery and some dev/test assets. These environments are hosted by two different providers. IT refers to these two datacenters as *Prod* and *DR* respectively.
- IT entered the cloud by migrating all end-user email accounts to Microsoft 365. This migration was completed six months ago. Few other IT assets have been deployed to the cloud.
- The application development teams are working in a dev/test capacity to learn about cloud-native capabilities.
- The business intelligence (BI) team is experimenting with big data in the cloud and curation of data on new platforms.
- The company has a loosely defined policy stating that personal customer data and financial data cannot be hosted in the cloud, which limits mission-critical applications in the current deployments.
- IT investments are controlled largely by capital expense. Those investments are planned yearly. In the past

several years, investments have included little more than basic maintenance requirements.

Future state

The following changes are anticipated over the next several years:

- The CIO is reviewing the policy on personal data and financial data to allow for the future state goals.
- The application development and BI teams want to release cloud-based solutions to production over the next 24 months based on the vision for customer engagement and new products.
- This year, the IT team will finish retiring the disaster recovery workloads of the DR datacenter by migrating 2,000 VMs to the cloud. This is expected to produce an estimated $25m USD cost savings over the next five years.



- The company plans to change how it makes IT investments by repositioning the committed capital expense as an operating expense within IT. This change will provide greater cost control and enable IT to accelerate other planned efforts.

Next steps

The company has developed a corporate policy to shape the governance implementation. The corporate policy drives many of the technical decisions.

Review the initial corporate policy

Standard enterprise governance guide: Initial corporate policy behind the governance strategy

3/6/2021 • 4 minutes to read • Edit Online

The following corporate policy defines an initial governance position, which is the starting point for this guide. This article defines early-stage risks, initial policy statements, and early processes to enforce policy statements.

> **NOTE**
> The corporate policy is not a technical document, but it drives many technical decisions. The governance MVP described in the overview ultimately derives from this policy. Before implementing a governance MVP, your organization should develop a corporate policy based on your own objectives and business risks.

Cloud governance team

In this narrative, the cloud governance team is comprised of two systems administrators who have recognized the need for governance. Over the next several months, they will inherit the job of cleaning up the governance of the company's cloud presence, earning them the title of *cloud custodians*. In subsequent iterations, this title will likely change.

Objective

The initial objective is to establish a foundation for governance agility. An effective Governance MVP allows the governance team to stay ahead of cloud adoption and implement guardrails as the adoption plan changes.

Business risks

The company is at an early stage of cloud adoption, experimenting and building proofs of concept. Risks are now relatively low, but future risks are likely to have a significant impact. There is little definition around the final state of the technical solutions to be deployed to the cloud. In addition, the cloud readiness of IT employees is low. A foundation for cloud adoption will help the team safely learn and grow.

Future-proofing: There is a risk of not empowering growth, but also a risk of not providing the right protections against future risks.

An agile yet robust governance approach is needed to support the board's vision for corporate and technical growth. Failure to implement such a strategy will slow technical growth, potentially risking current and future market share growth. The impact of such a business risk is unquestionably high. However, the role IT will play in those potential future states is unknown, making the risk associated with current IT efforts relatively high. That said, until more concrete plans are aligned, the business has a high tolerance for risk.

This business risk can be broken down tactically into several technical risks:

- Well-intended corporate policies could slow transformation efforts or break critical business processes, if not considered within a structured approval flow.
- The application of governance to deployed assets could be difficult and costly.
- Governance may not be properly applied across an application or workload, creating gaps in security.
- With so many teams working in the cloud, there is a risk of inconsistency.
- Costs may not properly align to business units, teams, or other budgetary management units.

- The use of multiple identities to manage various deployments could lead to security issues.
- Despite current policies, there is a risk that protected data could be mistakenly deployed to the cloud.

Tolerance indicators

The current tolerance for risk is high and the appetite for investing in cloud governance is low. As such, the tolerance indicators act as an early warning system to trigger more investment of time and energy. If and when the following indicators are observed, you should iteratively improve the governance strategy.

- **Cost management:** The scale of deployment exceeds predetermined limits on number of resources or monthly cost.
- **Security baseline:** Inclusion of protected data in defined cloud adoption plans.
- **Resource consistency:** Inclusion of any mission-critical applications in defined cloud adoption plans.

Policy statements

The following policy statements establish the requirements needed to remediate the defined risks. These policies define the functional requirements for the governance MVP. Each will be represented in the implementation of the governance MVP.

Cost Management:

- For tracking purposes, all assets must be assigned to an application owner within one of the core business functions.
- When cost concerns arise, additional governance requirements will be established with the finance team.

Security Baseline:

- Any asset deployed to the cloud must have an approved data classification.
- No assets identified with a protected level of data may be deployed to the cloud, until sufficient requirements for security and governance can be approved and implemented.
- Until minimum network security requirements can be validated and governed, cloud environments are seen as perimeter networks and should meet similar connection requirements to other datacenters or internal networks.

Resource Consistency:

- Because no mission-critical workloads are deployed at this stage, there are no SLA, performance, or BCDR requirements to be governed.
- When mission-critical workloads are deployed, additional governance requirements will be established with IT operations.

Identity Baseline:

- All assets deployed to the cloud should be controlled using identities and roles approved by current governance policies.
- All groups in the on-premises Active Directory infrastructure that have elevated privileges should be mapped to an approved RBAC role.

Deployment Acceleration:

- All assets must be grouped and tagged according to defined grouping and tagging strategies.
- All assets must use an approved deployment model.
- Once a governance foundation has been established for a cloud provider, any deployment tooling must be compatible with the tools defined by the governance team.

Processes

No budget has been allocated for ongoing monitoring and enforcement of these governance policies. Because of that, the cloud governance team has improvised ways to monitor adherence to policy statements.

- **Education:** The cloud governance team is investing time to educate the cloud adoption teams on the governance guides that support these policies.
- **Deployment reviews:** Before deploying any asset, the cloud governance team will review the governance guide with the cloud adoption teams.

Next steps

This corporate policy prepares the cloud governance team to implement the governance MVP, which will be the foundation for adoption. The next step is to implement this MVP.

Best practices explained

Standard enterprise governance guide: Best practices explained

5/18/2021 • 10 minutes to read • Edit Online

The governance guide starts with a set of initial corporate policies. These policies are used to establish a governance MVP that reflects best practices.

In this article, we discuss the high-level strategies that are required to create a governance MVP. The core of the governance MVP is the Deployment Acceleration discipline. The tools and patterns applied at this stage will enable the incremental improvements needed to expand governance in the future.

Governance MVP (initial governance foundation)

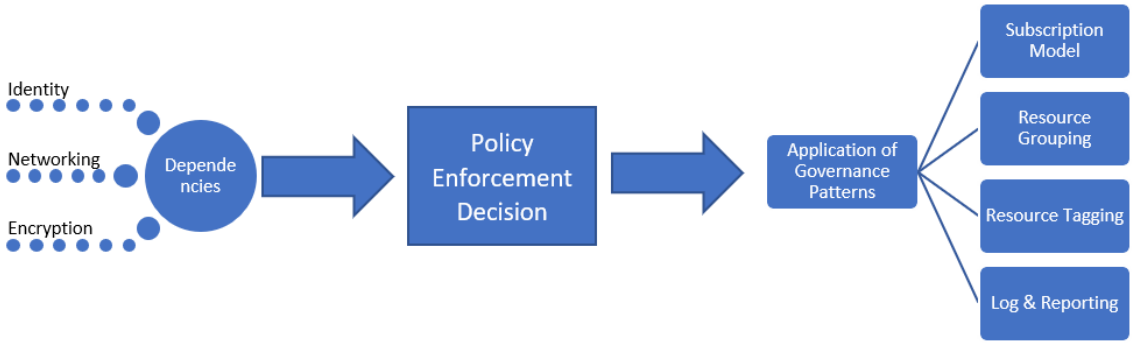
Rapid adoption of governance and corporate policy is achievable, thanks to a few simple principles and cloud-based governance tooling. These are the first three disciplines to approach in any governance process. Each discipline will be further described in this article.

To establish the starting point, this article discusses the high-level strategies behind the Security Baseline, Identity Baseline, and Deployment Acceleration disciplines that are required to create a governance MVP, which will serve as the foundation for all adoption.



Implementation process

The implementation of the governance MVP has dependencies on identity, security, and networking. Once the dependencies are resolved, the cloud governance team will decide a few aspects of governance. The decisions from the cloud governance team and from supporting teams will be implemented through a single package of enforcement assets.



This implementation can also be described using a simple checklist:

1. Solicit decisions regarding core dependencies: identity, networking, monitoring, and encryption.
2. Determine the pattern to be used during corporate policy enforcement.
3. Determine the appropriate governance patterns for the resource consistency, resource tagging, and logging and reporting disciplines.
4. Implement the governance tools aligned to the chosen policy enforcement pattern to apply the dependent decisions and governance decisions.

Dependent decisions

The following decisions come from teams outside of the cloud governance team. The implementation of each will come from those same teams. However, the cloud governance team is responsible for implementing a solution to validate that those implementations are consistently applied.

Identity Baseline

Identity Baseline is the fundamental starting point for all governance. Before attempting to apply governance, identity must be established. The established identity strategy will then be enforced by the governance solutions. In this governance guide, the Identity Management team implements the Directory Synchronization pattern:

- RBAC will be provided by Azure Active Directory (Azure AD), using the directory synchronization or "Same Sign-On" that was implemented during company's migration to Microsoft 365. For implementation guidance, see Reference Architecture for Azure AD Integration.
- The Azure AD tenant will also govern authentication and access for assets deployed to Azure.

In the governance MVP, the governance team will enforce application of the replicated tenant through subscription governance tooling, discussed later in this article. In future iterations, the governance team could also enforce rich tooling in Azure AD to extend this capability.

Security Baseline: Networking

Software Defined Network is an important initial aspect of the Security Baseline. Establishing the governance MVP depends on early decisions from the Security Management team to define how networks can be safely configured.

Given the lack of requirements, IT security is playing it safe and requires a Cloud DMZ pattern. That means governance of the Azure deployments themselves will be very light.

- Azure subscriptions may connect to an existing datacenter via VPN, but must follow all existing on-premises IT governance policies regarding connection of a perimeter network to protected resources. For implementation guidance regarding VPN connectivity, see On-premises network connected to Azure using a VPN gateway.
- Decisions regarding subnet, firewall, and routing are currently being deferred to each application/workload lead.
- Additional analysis is required before releasing of any protected data or mission-critical workloads.

In this pattern, cloud networks can only connect to on-premises resources over an existing VPN that is compatible with Azure. Traffic over that connection will be treated like any traffic coming from a perimeter network. Additional considerations may be required on the on-premises edge device to securely handle traffic from Azure.

The cloud governance team has proactively invited members of the networking and IT security teams to regular meetings, in order to stay ahead of networking demands and risks.

Security Baseline: Encryption

Encryption is another fundamental decision within the Security Baseline discipline. Because the company

currently does not yet store any protected data in the cloud, the Security Team has decided on a less aggressive pattern for encryption. At this point, a cloud-native pattern for encryption is suggested but not required of any development team.

- No governance requirements have been set regarding the use of encryption, because the current corporate policy does not permit mission-critical or protected data in the cloud.
- Additional analysis will be required before releasing any protected data or mission-critical workloads.

Policy enforcement

The first decision to make regarding Deployment Acceleration is the pattern for enforcement. In this narrative, the governance team decided to implement the Automated Enforcement pattern.

- Azure Security Center will be made available to the security and identity teams to monitor security risks. Both teams are also likely to use Security Center to identify new risks and improve corporate policy.
- RBAC is required in all subscriptions to govern authentication enforcement.
- Azure Policy will be published to each management group and applied to all subscriptions. However, the level of policies being enforced will be very limited in this initial Governance MVP.
- Although Azure management groups are being used, a relatively simple hierarchy is expected.
- Azure Blueprints will be used to deploy and update subscriptions by applying RBAC requirements, Resource Manager Templates, and Azure Policy across management groups.

Apply the dependent patterns

The following decisions represent the patterns to be enforced through the policy enforcement strategy above:

Identity Baseline. Azure Blueprints will set RBAC requirements at a subscription level to ensure that consistent identity is configured for all subscriptions.

Security Baseline: Networking. The cloud governance team maintains a Resource Manager template for establishing a VPN gateway between Azure and the on-premises VPN device. When an application team requires a VPN connection, the cloud governance team will apply the gateway Resource Manager template via Azure Blueprints.

Security Baseline: Encryption. At this point, no policy enforcement is required in this area. This will be revisited during later iterations.

Application of governance-defined patterns

The cloud governance team is responsible for the following decisions and implementations. Many require inputs from other teams, but the cloud governance team is likely to own both the decision and the implementation. The following sections outline the decisions made for this use case and details of each decision.

Subscription design

The decision on what subscription design to use determines how Azure subscriptions get structured and how Azure management groups will be used to efficiently manage access, policies, and compliance of these subscription. In this narrative, the governance team has established subscriptions for production and nonproduction workloads production-and-nonproduction subscription design pattern.

- Departments are not likely to be required given the current focus. Deployments are expected to be constrained within a single billing unit. At the stage of adoption, there may not even be an Enterprise Agreement to centralize billing. It's likely that this level of adoption is being managed by a single pay-as-you-go Azure subscription.
- Regardless of the use of the EA portal or the existence of an Enterprise Agreement, a subscription model should still be defined and agreed on to minimize administrative overheard beyond just billing.

- A common naming convention should be agreed on as part of the subscription design, based on the previous two points.

Resource consistency

Resource consistency decisions determine the tools, processes, and effort required to ensure Azure resources are deployed, configured, and managed consistently within a subscription. In this narrative, deployment consistency has been chosen as the primary resource consistency pattern.

- Resource groups are created for applications using the lifecycle approach. Everything that is created, maintained, and retired together should reside a single resource group. For more information, see the resource consistency decision guide.
- Azure Policy should be applied to all subscriptions from the associated management group.
- As part of the deployment process, Azure resource consistency templates for the resource group should be stored in source control.
- Each resource group is associated with a specific workload or application based on the lifecycle approach described above.
- Azure management groups enable updating governance designs as corporate policy matures.
- Extensive implementation of Azure Policy could exceed the team's time commitments and may not provide a great deal of value at this time. A simple default policy should be created and applied to each management group to enforce the small number of current cloud governance policy statements. This policy will define the implementation of specific governance requirements. Those implementations can then be applied across all deployed assets.

> **IMPORTANT**
> Any time a resource in a resource group no longer shares the same lifecycle, it should be moved to another resource group. Examples include common databases and networking components. While they may serve the application being developed, they may also serve other purposes and should therefore exist in other resource groups.

Resource tagging

Resource tagging decisions determine how metadata is applied to Azure resources within a subscription to support operations, management, and accounting purposes. In this narrative, the classification pattern has been chosen as the default model for resource tagging.

- Deployed assets should be tagged with:
  - Data classification
  - Criticality
  - SLA
  - Environment
- These four values will drive governance, operations, and security decisions.
- If this governance guide is being implemented for a business unit or team within a larger corporation, tagging should also include metadata for the billing unit.

Logging and reporting

Logging and reporting decisions determine how your store log data and how the monitoring and reporting tools that keep IT staff informed on operational health are structured. In this narrative, a cloud-native pattern** for logging and reporting is suggested.

Incremental improvement of governance processes

As governance changes, some policy statements can't or shouldn't be controlled by automated tooling. Other policies will result in effort by the IT security team and the on-premises identity management team over time. To

help manage new risks as they arise, the cloud governance team will oversee the following processes.

Adoption acceleration: The cloud governance team has been reviewing deployment scripts across multiple teams. They maintain a set of scripts that serve as deployment templates. Those templates are used by the cloud adoption and DevOps teams to define deployments more quickly. Each of those scripts contains the necessary requirements to enforce a set of governance policies with no additional effort from cloud adoption engineers. As the curators of these scripts, the cloud governance team can more quickly implement policy changes. As a result of script curation, the cloud governance team is seen as a source of adoption acceleration. This creates consistency among deployments, without strictly forcing adherence.

Engineer training: The cloud governance team offers bimonthly training sessions and has created two videos for engineers. These materials help engineers quickly learn the governance culture and how things are done during deployments. The team is adding training assets that show the difference between production and nonproduction deployments, so that engineers will understand how the new policies will affect adoption. This creates consistency among deployments, without strictly forcing adherence.

Deployment planning: Before deploying any asset containing protected data, the cloud governance team will review deployment scripts to validate governance alignment. Existing teams with previously approved deployments will be audited using programmatic tooling.

Monthly audit and reporting: Each month, the cloud governance team runs an audit of all cloud deployments to validate continued alignment to policy. When deviations are discovered, they're documented and shared with the cloud adoption teams. When enforcement doesn't risk a business interruption or data leak, the policies are automatically enforced. At the end of the audit, the cloud governance team compiles a report for the cloud strategy team and each cloud adoption team to communicate overall adherence to policy. The report is also stored for auditing and legal purposes.

Quarterly policy review: Each quarter, the cloud governance team and the cloud strategy team will review audit results and suggest changes to corporate policy. Many of those suggestions are the result of continuous improvements and the observation of usage patterns. Approved policy changes are integrated into governance tooling during subsequent audit cycles.

Alternative patterns

If any of the patterns selected in this governance guide don't align with the reader's requirements, alternatives to each pattern are available:

- Encryption patterns
- Identity patterns
- Logging and reporting patterns
- Policy enforcement patterns
- Resource consistency patterns
- Resource tagging patterns
- Software Defined Networking patterns
- Subscription design patterns

Next steps

Once this guide is implemented, each cloud adoption team can go forth with a sound governance foundation. At the same time, the cloud governance team will work to continuously update the corporate policies and governance disciplines.

The two teams will use the tolerance indicators to identify the next set of improvements needed to continue supporting cloud adoption. For the fictional company in this guide, the next step is improving the security baseline to support moving protected data to the cloud.

Improve the Security Baseline discipline

Standard enterprise governance guide: Improve the Security Baseline discipline

3/6/2021 • 9 minutes to read • Edit Online

This article advances the governance strategy narrative by adding security controls that support moving protected data to the cloud.

Advancing the narrative

IT and business leadership are happy with results from early experimentation by the IT, application development, and BI teams. To realize tangible business values from these experiments, those teams must be allowed to integrate protected data into solutions. This integration triggers changes to corporate policy. It also requires incremental improvement of the cloud governance implementations before protected data can land in the cloud.

Changes to the cloud governance team

Given the effect of the changing narrative and support provided so far, the cloud governance team is now viewed differently. The two system administrators who started the team are now viewed as experienced cloud architects. As this narrative develops, the perception of them will shift from that of cloud custodians to more of a cloud guardian role.

The difference is subtle, but it's an important distinction when you're creating a governance-focused IT culture. A cloud custodian cleans up the messes made by innovative cloud architects. The two roles have natural friction and opposing goals. On the other hand, a cloud guardian helps keep the cloud safe so other cloud architects can move more quickly, with fewer messes. And a cloud guardian is involved in creating templates that accelerate deployment and adoption. So they're innovation accelerators in addition to being defenders of the Five Disciplines of Cloud Governance.

Changes in the current state

At the start of this narrative, the application development teams were still working in a dev/test capacity, and the BI team was still in the experimental phase. IT operated two hosted infrastructure environments, referred to as `Prod` and `DR`.

Since then, some things have changed that will affect governance:

- The application development team has implemented a CI/CD pipeline to deploy a cloud-native application with an improved user experience. That application doesn't yet interact with protected data, so it isn't production ready.
- The business intelligence team within IT actively curates data in the cloud from logistics, inventory, and third-party sources. This data is used to drive new predictions, which could shape business processes. Those predictions and insights aren't actionable until customer and financial data can be integrated into the data platform.
- The IT team is progressing on the CIO and CFO plans to retire the DR datacenter. More than 1,000 of the 2,000 assets in the DR datacenter have been retired or migrated.
- The loosely defined policies for personal data and financial data have been modernized. The new corporate policies are contingent on the implementation of related security and governance policies. Teams are still stalled.

Incrementally improve the future state

Early experiments by the application development and BI teams show potential improvements in customer

experiences and data-driven decisions. Both teams want to expand adoption of the cloud over the next 18 months by deploying those solutions to production.

During the remaining six months, the cloud governance team will implement security and governance requirements to allow the cloud adoption teams to migrate the protected data in those datacenters.

The changes to current and future state expose new risks that require new policy statements.

Changes in tangible risks

Data breach: When you adopt any new data platform, there's an inherent increase in liabilities related to potential data breaches. Technicians adopting cloud technologies have increased responsibility to implement solutions that can reduce this risk. A robust security and governance strategy must be implemented to ensure those technicians fulfill those responsibilities.

This business risk can be expanded into a few technical risks:

- Mission-critical applications or protected data might be deployed unintentionally.
- Protected data might be exposed during storage because of poor encryption decisions.
- Unauthorized users might access protected data.
- External intrusion might result in access to protected data.
- External intrusion or denial-of-service attacks might cause a business interruption.
- Organization or employment changes might allow unauthorized access to protected data.
- New exploits could create new intrusion or access opportunities.
- Inconsistent deployment processes might result in security gaps, which could lead to data leaks or interruptions.
- Configuration drift or missed patches might result in unintended security gaps, which could lead to data leaks or interruptions.

Data loss: There's also an inherent risk of data loss in the new platform. The security and governance strategy should consider the following scenarios in which data loss can happen:

- A mission-critical resource is lost or deleted.
- A mission-critical resource is present, but the data is lost because of accidental deletion.
- A mission-critical resource is present, but the data is lost because of malicious administration.

Incremental improvement of policy statements

The following changes to policy will help remediate the new risks and guide implementation. The list looks long, but adopting these policies might be easier than you think.

- All deployed assets must be categorized by criticality and data classification. The cloud governance team and the application owner must review these classifications before deployment to the cloud.
- Applications that store or access protected data must be managed differently than applications that don't. At a minimum, they should be segmented to avoid unintended access of protected data.
- All protected data must be encrypted when at rest. This encryption is the default for all Azure Storage accounts. But you might need additional encryption strategies, including encryption of the data within the storage account, encryption of VMs, and database-level encryption when you use SQL in a VM (TDE and column encryption).
- Mission-critical data can be deleted accidentally. You need to develop a data backup strategy to handle this risk and restore the data from before the deletion point. A malicious admin can delete the mission-critical data and its backups as well. To handle this scenario, deletions of backup data should be soft deletions that can be reversed. Azure Backup can help with both of these scenarios.
- Elevated permissions in any segment that contains protected data should be an exception. Any such

exceptions will be recorded with the cloud governance team and audited regularly.

- Network subnets that contain protected data must be isolated from other subnets. Network traffic between protected data subnets will be audited regularly.
- No subnet that contains protected data should be directly accessible over the public internet or across datacenters. Access to those subnets must be routed through intermediate subnets. All access into those subnets must come through a firewall solution that can perform packet scanning and blocking functions.
- Governance tooling must audit and enforce network configuration requirements defined by the security management team.
- Governance tooling must limit VM deployment to only approved images.
- Governance process must validate that backup, recovery, and SLA adherence are properly implemented for mission-critical applications and protected data.
- Whenever possible, node configuration management should apply policy requirements to the configuration of any guest operating system.
- Governance tooling must enforce that automatic updates are enabled on all deployed assets. Violations must be reviewed with operational management teams and remediated in accordance with operations policies. Assets that aren't automatically updated must be included in processes owned by IT operations.
- Creation of new subscriptions or management groups for any mission-critical applications or protected data will require a review from the cloud governance team to ensure that the proper blueprint is assigned.
- A least-privilege access model will be applied to any management group or subscription that contains mission-critical applications or protected data.
- Trends and exploits that could affect cloud deployments should be reviewed regularly by the security team to provide updates to security management tooling used in the cloud.
- Deployment tooling must be approved by the cloud governance team to ensure ongoing governance of deployed assets.
- Deployment scripts must be maintained in a central repository accessible by the cloud governance team for periodic review and auditing.
- Governance processes must include audits at the point of deployment and at regular cycles to ensure consistency across all assets.
- Deployment of any applications that require customer authentication must use an approved identity provider that's compatible with the primary identity provider for internal users.
- Cloud governance processes must include quarterly reviews with identity management teams. These reviews can help identify malicious actors or usage patterns that should be prevented by cloud asset configuration.

Incremental improvement of governance practices

The governance MVP design will change to include new Azure policies and an implementation of Azure Cost Management + Billing. Together, these two design changes will fulfill the new corporate policy statements.

- The networking and IT security teams will define network requirements. The cloud governance team will support the conversation.
- The identity and IT security teams will define identity requirements and make any necessary changes to local Active Directory implementation. The cloud governance team will review changes.
- Create a repository in Azure DevOps to store and version all relevant Azure Resource Manager templates and scripted configurations.
- Azure Recovery Services vault implementation:
  - Define and deploy an Azure Recovery Services vault for backup and recovery processes.
  - Create a Resource Manager template for creation of a vault in each subscription.
- Azure Security Center implementation:
  - Configure Azure Security Center for any management group that contains protected data classifications.

    - Set automatic provisioning to on by default to ensure patching compliance.
    - Establish OS security configurations. The IT security team will define the configuration.
    - Support the IT security team in the initial use of Security Center. Transition the use of Security Center to the IT security team, but maintain access for the purpose of continually improving governance.
    - Create a Resource Manager template that reflects the changes required for Security Center configuration within a subscription.
- Update Azure policies for all subscriptions:
    - Audit and enforce the criticality of data and data classification across all management groups and subscriptions to identify any subscriptions with protected data classifications.
    - Audit and enforce the use of approved images only.
- Update Azure policies for all subscriptions that contain protected data classifications:
    - Audit and enforce the use of standard Azure roles only.
    - Audit and enforce encryption for all storage accounts and files at rest on individual nodes.
    - Audit and enforce the application of an NSG to all NICs and subnets. The networking and IT security teams will define the NSG.
    - Audit and enforce the use of approved network subnet and virtual network per network interface.
    - Audit and enforce the limitation of user-defined routing tables.
    - Apply the built-in policies for guest configuration as follows:
        - Audit that Windows web servers are using secure communication protocols.
        - Audit that password security settings are set correctly inside Linux and Windows machines.
    - Audit and enforce that Azure Recovery Services vaults exist in the subscription.
- Firewall configuration:
    - Identify a configuration of Azure Firewall that meets necessary security requirements. Alternatively, identify a compatible third-party appliance that's compatible with Azure. The Azure Security Benchmark provides additional information on network security strategy and firewall configurations to support your security strategy.
    - Create a Resource Manager template to deploy the firewall with required configurations.
- Azure Blueprints:
    - Create a new blueprint named `protected-data`.
    - Add the Azure Firewall templates, Azure Security Center templates, and Azure Recovery Services vault templates to the blueprint.
    - Add the new policies for protected data subscriptions.
    - Publish the blueprint to any management group that currently plans on hosting protected data.
    - Apply the new blueprint to each affected subscription and to existing blueprints.

Conclusion

Adding the above processes and changes to the governance MVP will help to remediate many of the risks associated with security governance. Together, they add the network, identity, and security monitoring tools needed to protect data.

Next steps

As cloud adoption continues and delivers additional business value, risks and cloud governance needs also change. For the fictional company in this guide, the next step is to support mission-critical workloads. At this point, resource consistency controls are needed.

Improve the Resource Consistency discipline

Standard enterprise governance guide: Improve the Resource Consistency discipline

3/6/2021 • 6 minutes to read • Edit Online

This article advances the narrative by adding resource consistency controls to support mission-critical applications.

Advancing the narrative

New customer experiences, new prediction tools, and migrated infrastructure continue to progress. The business is now ready to begin using those assets in a production capacity.

Changes in the current state

In the previous phase of this narrative, the application development and BI teams were nearly ready to integrate customer and financial data into production workloads. The IT team was in the process of retiring the DR datacenter.

Since then, some things have changed that will affect governance:

- IT has retired 100% of the DR datacenter, ahead of schedule. In the process, a set of assets in the production datacenter were identified as cloud migration candidates.
- The application development teams are now ready for production traffic.
- The BI team is ready to feed predictions and insights back into operation systems in the production datacenter.

Incrementally improve the future state

Before using Azure deployments in production business processes, cloud operations must mature. In conjunction, additional governance changes is required to ensure assets can be operated properly.

The changes to current and future state expose new risks that will require new policy statements.

Changes in tangible risks

Business interruption: There is an inherent risk of any new platform causing interruptions to mission-critical business processes. The IT operations team and the teams executing on various cloud adoptions are relatively inexperienced with cloud operations. This increases the risk of interruption and must be remediated and governed.

This business risk can be expanded into several technical risks:

1. External intrusion or denial of service attacks might cause a business interruption.
2. Mission-critical assets may not be properly discovered, and therefore might not be properly operated.
3. Undiscovered or mislabeled assets might not be supported by existing operational management processes.
4. The configuration of deployed assets may not meet performance expectations.
5. Logging might not be properly recorded and centralized to allow for remediation of performance issues.
6. Recovery policies may fail or take longer than expected.
7. Inconsistent deployment processes might result in security gaps that could lead to data leaks or interruptions.
8. Configuration drift or missed patches might result in unintended security gaps that could lead to data leaks or interruptions.

9. Configuration might not enforce the requirements of defined SLAs or committed recovery requirements.
10. Deployed operating systems or applications might fail to meet hardening requirements.
11. With so many teams working in the cloud, there is a risk of inconsistency.

Incremental improvement of the policy statements

The following changes to policy will help remediate the new risks and guide implementation. The list looks long, but adopting these policies may be easier than it appears.

1. All deployed assets must be categorized by criticality and data classification. Classifications are to be reviewed by the cloud governance team and the application owner before deployment to the cloud.
2. Subnets containing mission-critical applications must be protected by a firewall solution capable of detecting intrusions and responding to attacks.
3. Governance tooling must audit and enforce network configuration requirements defined by the security management team.
4. Governance tooling must validate that all assets related to mission-critical applications or protected data are included in monitoring for resource depletion and optimization.
5. Governance tooling must validate that the appropriate level of logging data is being collected for all mission-critical applications or protected data.
6. Governance process must validate that backup, recovery, and SLA adherence are properly implemented for mission-critical applications and protected data.
7. Governance tooling must limit virtual machine deployments to approved images only.
8. Governance tooling must enforce that automatic updates are prevented on all deployed assets that support mission-critical applications. Violations must be reviewed with operational management teams and remediated in accordance with operations policies. Assets that are not automatically updated must be included in processes owned by IT operations.
9. Governance tooling must validate tagging related to cost, criticality, SLA, application, and data classification. All values must align to predefined values managed by the governance team.
10. Governance processes must include audits at the point of deployment and at regular cycles to ensure consistency across all assets.
11. Trends and exploits that could affect cloud deployments should be reviewed regularly by the security team to provide updates to security management tooling used in the cloud.
12. Before release into production, all mission-critical applications and protected data must be added to the designated operational monitoring solution. Assets that cannot be discovered by the chosen IT operations tooling, cannot be released for production use. Any changes required to make the assets discoverable must be made to the relevant deployment processes to ensure assets will be discoverable in future deployments.
13. When discovered, operational management teams will size assets, to ensure that assets meet performance requirements.
14. Deployment tooling must be approved by the cloud governance team to ensure ongoing governance of deployed assets.
15. Deployment scripts must be maintained in a central repository accessible by the cloud governance team for periodic review and auditing.
16. Governance review processes must validate that deployed assets are properly configured in alignment with SLA and recovery requirements.

Incremental improvement of governance practices

This section of the article will change the governance MVP design to include new Azure policies and an implementation of Azure Cost Management + Billing. Together, these two design changes will fulfill the new corporate policy statements.

1. The cloud operations team will define operational monitoring tooling and automated remediation tooling. The cloud governance team will support those discovery processes. In this use case, the cloud operations team chose Azure Monitor as the primary tool for monitoring mission-critical applications.
2. Create a repository in Azure DevOps to store and version all relevant Resource Manager templates and scripted configurations.
3. Azure Recovery Services vault implementation:
   a. Define and deploy an Azure Recovery Services vault for backup and recovery processes.
   b. Create a Resource Manager template for creation of a vault in each subscription.
4. Update Azure Policy for all subscriptions:
   a. Audit and enforce criticality and data classification across all subscriptions to identify any subscriptions with mission-critical assets.
   b. Audit and enforce the use of approved images only.
5. Azure Monitor implementation:
   a. Once a mission-critical workload is identified, create an Azure Monitor Log Analytics workspace.
   b. During deployment testing, the cloud operations team deploys the necessary agents and tests discovery.
6. Update Azure Policy for all subscriptions that contain mission-critical applications.
   a. Audit and enforce the application of an NSG to all NICs and subnets. Networking and IT security define the NSG.
   b. Audit and enforce the use of approved network subnets and virtual networks for each network interface.
   c. Audit and enforce the limitation of user-defined routing tables.
   d. Audit and enforce deployment of Azure Monitor agents for all virtual machines.
   e. Audit and enforce that Azure Recovery Services vaults exist in the subscription.
7. Firewall configuration:
   a. Identify a configuration of Azure Firewall that meets security requirements. Alternatively, identify a third-party appliance that is compatible with Azure.
   b. Create a Resource Manager template to deploy the firewall with required configurations.
8. Azure blueprint:
   a. Create a new Azure blueprint named `protected-data`.
   b. Add the firewall and Azure Recovery Services vault templates to the blueprint.
   c. Add the new policies for protected data subscriptions.
   d. Publish the blueprint to any management group that will host mission-critical applications.
   e. Apply the new blueprint to each affected subscription as well as existing blueprints.

Conclusion

These additional processes and changes to the governance MVP help remediate many of the risks associated with resource governance. Together they add recovery, sizing, and monitoring controls that empower cloud-aware operations.

Next steps

As cloud adoption continues and delivers additional business value, risks and cloud governance needs will also change. For the fictional company in this guide, the next trigger is when the scale of deployment exceeds 100 assets to the cloud or monthly spending exceeds $1,000 per month. At this point, the cloud governance team adds cost management controls.

Improve the Cost Management discipline

Standard enterprise governance guide: Improve the Cost Management discipline

3/6/2021 • 4 minutes to read • Edit Online

This article advances the narrative by adding cost controls to the governance MVP.

Advancing the narrative

Adoption has grown beyond the cost tolerance indicator defined in the governance MVP. This is a good thing, as it corresponds with migrations from the DR datacenter. The increase in spending now justifies an investment of time from the cloud governance team.

Changes in the current state

In the previous phase of this narrative, IT had retired 100% of the DR datacenter. The application development and BI teams were ready for production traffic.

Since then, some things have changed that will affect governance:

- The migration team has begun migrating VMs out of the production datacenter.
- The application development teams is actively pushing production applications to the cloud through CI/CD pipelines. Those applications can reactively scale with user demands.
- The business intelligence team within IT has delivered several predictive analytics tools in the cloud. The volumes of data aggregated in the cloud continues to grow.
- All of this growth supports committed business outcomes. Costs have begun to balloon. Projected budgets are growing faster than expected. The CFO needs improved approaches to managing costs.

Incrementally improve the future state

Cost monitoring and reporting is to be added to the cloud solution. IT is still serving as a cost clearing house. This means that payment for cloud services continues to come from IT procurement. Reporting should tie direct operating expenses to the functions that are consuming the cloud costs. This model is referred to as a *showback* cloud accounting model.

The changes to current and future state expose new risks that will require new policy statements.

Changes in tangible risks

Budget control: There is an inherent risk that self-service capabilities will result in excessive and unexpected costs on the new platform. Governance processes for monitoring costs and mitigating ongoing cost risks must be in place to ensure continued alignment with the planned budget.

This business risk can be expanded into a few technical risks:

- Actual costs might exceed the plan.
- Business conditions change. When they do, there will be cases when a business function needs to consume more cloud services than expected, leading to spending anomalies. There is a risk that this extra spending will be considered overages, as opposed to a necessary adjustment to the plan.
- Systems could be overprovisioned, resulting in excess spending.

Incremental improvement of the policy statements

The following changes to policy will help remediate the new risks and guide implementation.

- All cloud costs should be monitored against plan on a weekly basis by the governance team. Reporting on deviations between cloud costs and plan is to be shared with IT leadership and finance monthly. All cloud costs and plan updates should be reviewed with IT leadership and finance monthly.
- All costs must be allocated to a business function for accountability purposes.
- Cloud assets should be continually monitored for optimization opportunities.
- Cloud governance tooling must limit asset sizing options to an approved list of configurations. The tooling must ensure that all assets are discoverable and tracked by the cost monitoring solution.
- During deployment planning, any required cloud resources associated with the hosting of production workloads should be documented. This documentation will help refine budgets and prepare additional automation to prevent the use of more expensive options. During this process consideration should be given to different discounting tools offered by the cloud provider, such as reserved instances or license cost reductions.
- All application owners are required to attend trained on practices for optimizing workloads to better control cloud costs.

Incremental improvement of best practices

This section of the article will change the governance MVP design to include new Azure policies and an implementation of Azure Cost Management + Billing. Together, these two design changes will fulfill the new corporate policy statements.

1. Implement Azure Cost Management + Billing.
    a. Establish the right scope of access to align with the subscription pattern and the Resource Consistency discipline. Assuming alignment with the governance MVP defined in prior articles, this requires **enrollment account scope** access for the cloud governance team executing on high-level reporting. Additional teams outside of governance may require **resource group scope** access.
    b. Establish a budget in Azure Cost Management + Billing.
    c. Review and act on initial recommendations. Have a recurring process to support reporting.
    d. Configure and execute Azure Cost Management + Billing reporting, both initial and recurring.
2. Update Azure Policy
    a. Audit the tagging, management group, subscription, and resource group values to identify any deviation.
    b. Establish SKU size options to limit deployments to SKUs listed in deployment planning documentation.

Conclusion

Adding these processes and changes to the governance MVP helps remediate many of the risks associated with cost governance. Together, they create the visibility, accountability, and optimization needed to control costs.

Next steps

As cloud adoption continues and delivers additional business value, risks and cloud governance needs will also change. For the fictional company in this guide, the next step is using this governance investment to manage multiple clouds.

Multicloud evolution

Standard enterprise governance guide: Multicloud improvement

3/6/2021 • 4 minutes to read • Edit Online

This article advances the narrative by adding controls for multicloud adoption.

Advancing the narrative

Microsoft recognizes that customers may adopt multiple clouds for specific purposes. The fictional customer in this guide is no exception. In parallel with their Azure adoption journey, business success has led to the acquisition of a small but complementary business. That business is running all of their IT operations on a different cloud provider.

This article describes how things change when integrating the new organization. For purposes of the narrative, we assume this company has completed each of the governance iterations outlined in this governance guide.

Changes in the current state

In the previous phase of this narrative, the company had begun actively pushing production applications to the cloud through CI/CD pipelines.

Since then, some things have changed that will affect governance:

- Identity is controlled by an on-premises instance of Active Directory. Hybrid identity is facilitated through replication to Azure Active Directory.
- IT operations or cloud operations are largely managed by Azure Monitor and related automated processes.
- Disaster recovery and business continuity is controlled by Azure Recovery Services vaults.
- Azure Security Center is used to monitor security violations and attacks.
- Azure Security Center and Azure Monitor are both used to monitor governance of the cloud.
- Azure Blueprints, Azure Policy, and Azure management groups are used to automate compliance with policy.

Incrementally improve the future state

The goal is to integrate the acquisition company into existing operations wherever possible.

Changes in tangible risks

Business acquisition cost: Acquisition of the new business is estimated to be profitable in approximately five years. Because of the slow rate of return, the board wants to control acquisition costs, as much as possible. There is a risk of cost control and technical integration conflicting with one another.

This business risk can be expanded into a few technical risks:

- Cloud migration might produce additional acquisition costs.
- The new environment might not be properly governed, which could result in policy violations.

Incremental improvement of the policy statements

The following changes to policy will help remediate the new risks and guide implementation:

- All assets in a secondary cloud must be monitored through existing operational management and security monitoring tools.

- All organization units must be integrated into the existing identity provider.
- The primary identity provider should govern authentication to assets in the secondary cloud.

Incremental improvement of governance practices

This section of the article will change the governance MVP design to include new Azure policies and an implementation of Azure Cost Management + Billing. Together, these design changes will fulfill the new corporate policy statements.

1. Connect the networks. This step is executed by the networking and IT security teams, and supported by the cloud governance team. Adding a connection from the MPLS/leased-line provider to the new cloud will integrate networks. Adding routing tables and firewall configurations will control access and traffic between the environments.
2. Consolidate identity providers. Depending on the workloads being hosted in the secondary cloud, there are a variety of options to identity provider consolidation. The following are a few examples:
    a. For applications that authenticate using OAuth 2, users from Active Directory in the secondary cloud can simply be replicated to the existing Azure AD tenant. This ensures all users can be authenticated in the tenant.
    b. At the other extreme, federation allows OUs to flow into Active Directory on-premises, then into the Azure AD instance.
3. Add assets to Azure Site Recovery.
    a. Azure Site Recovery was designed from the beginning as a hybrid or multicloud tool.
    b. VMs in the secondary cloud might be able to be protected by the same Azure Site Recovery processes used to protect on-premises assets.
4. Add assets to Azure Cost Management + Billing.
    a. Azure Cost Management + Billing was designed from the beginning as a multicloud tool.
    b. Virtual machines in the secondary cloud may be compatible with Azure Cost Management + Billing for some cloud providers. Additional costs may apply.
5. Add assets to Azure Monitor.
    a. Azure Monitor was designed as a hybrid cloud tool from inception.
    b. Virtual machines in the secondary cloud may be compatible with Azure Monitor agents, allowing them to be included in Azure Monitor for operational monitoring.
6. Adopt governance enforcement tools.
    a. Governance enforcement is cloud-specific.
    b. The corporate policies established in the governance guide are not cloud-specific. While the implementation may vary from cloud to cloud, the policies can be applied to the secondary provider.

Multicloud adoption should be contained to where it's required based on technical needs or specific business requirements. As multicloud adoption grows, so does complexity and security risks.

Conclusion

This series of articles described the incremental development of governance best practices, aligned with the experiences of this fictional company. By starting small, but with the right foundation, the company could move quickly and yet still apply the right amount of governance at the right time. The MVP by itself did not protect the customer. Instead, it created the foundation to manage risks and add protections. From there, layers of governance were applied to remediate tangible risks. The exact journey presented here won't align 100% with the experiences of any reader. Rather, it serves as a pattern for incremental governance. You should mold these best practices to fit your own unique constraints and governance requirements.

Governance guide for complex enterprises

3/6/2021 • 8 minutes to read • Edit Online

Overview of best practices

This governance guide follows the experiences of a fictional company through various stages of governance maturity. It's based on real customer experiences. The suggested best practices are based on the constraints and needs of the fictional company.

As a quick starting point, this overview defines a minimum viable product (MVP) for governance based on best practices. It also provides links to some governance improvements that add further best practices as new business or technical risks emerge.

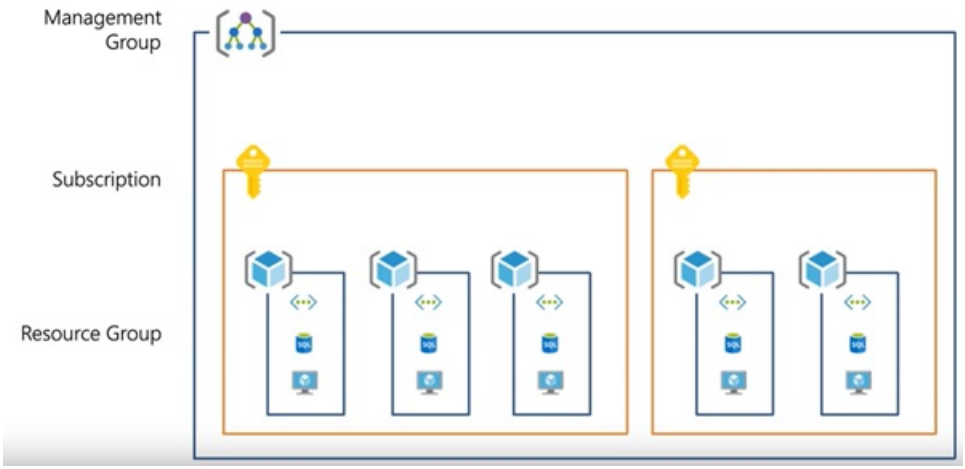
> **WARNING**
> This MVP is a baseline starting point, based on a set of assumptions. Even this minimal set of best practices is based on corporate policies that are driven by unique business risks and risk tolerances. To see whether these assumptions apply to you, read the longer narrative that follows this article.

Governance best practices

These best practices serve as a foundation for an organization to quickly and consistently add governance guardrails across multiple Azure subscriptions.

Resource organization

The following diagram shows the governance MVP hierarchy for organizing resources.

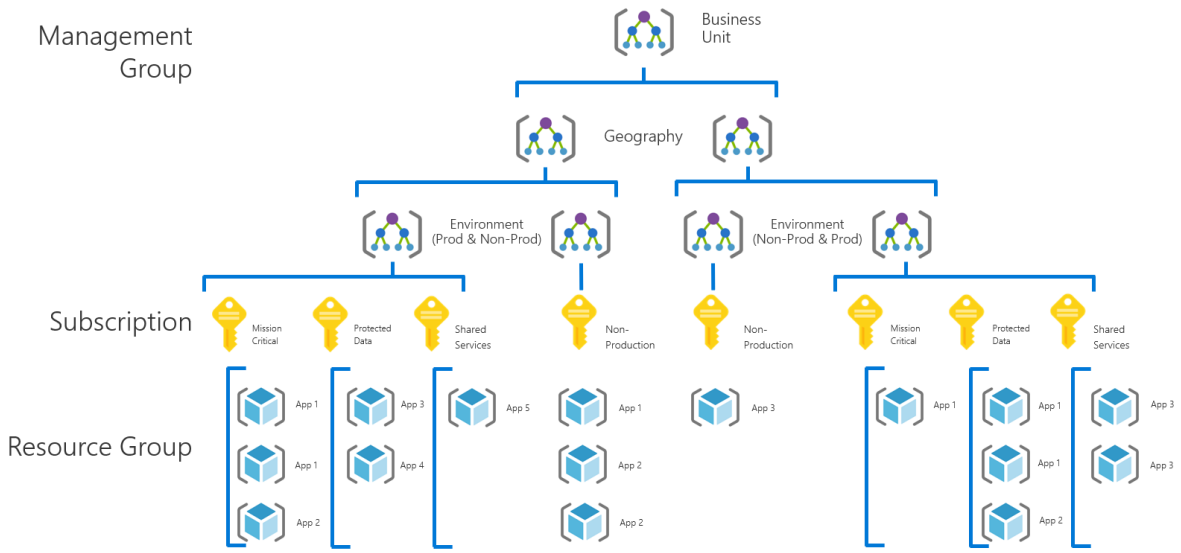


Every application should be deployed in the proper area of the management group, subscription, and resource group hierarchy. During deployment planning, the cloud governance team will create the necessary nodes in the hierarchy to empower the cloud adoption teams.

1. Define a management group for each business unit with a detailed hierarchy that reflects geography first, then environment type (for example, production or nonproduction environments).
2. Create a production subscription and a nonproduction subscription for each unique combination of discrete business unit or geography. Creating multiple subscriptions requires careful consideration. For more information, see the subscription decision guide.
3. Apply consistent nomenclature at each level of this grouping hierarchy.

4. Resource groups should be deployed in a manner that considers its contents lifecycle. Resources that are developed together, managed together, and retired together belong in the same resource group. For more information about best practices for using resource groups, see the resource consistency decision guide.

5. Region selection is incredibly important and must be considered so that networking, monitoring, auditing can be in place for failover/failback as well as confirmation that needed SKUs are available in the preferred regions.



These patterns provide room for growth without making the hierarchy needlessly complicated.

> **NOTE**
> In the event of changes to your business requirements, Azure management groups allow you to easily reorganize your management hierarchy and subscription group assignments. However, keep in mind that policy and role assignments applied to a management group are inherited by all subscriptions underneath that group in the hierarchy. If you plan to reassign subscriptions between management groups, make sure that you are aware of any policy and role assignment changes that may result. See the Azure management groups documentation for more information.

Governance of resources

A set of global policies and RBAC roles will provide a baseline level of governance enforcement. To meet the cloud governance team's policy requirements, implementing the governance MVP requires completing the following tasks:

1. Identify the Azure Policy definitions needed to enforce business requirements. This might include using built-in definitions and creating new custom definitions. To keep up with the pace of newly released built-in definitions, there's an atom feed of all the commits for built-in policies, which you can use for an RSS feed. Alternatively, you can check AzAdvertizer.
2. Create a blueprint definition using these built-in and custom policy and the role assignments required by the governance MVP.
3. Apply policies and configuration globally by assigning the blueprint definition to all subscriptions.

Identify policy definitions

Azure provides several built-in policies and role definitions that you can assign to any management group, subscription, or resource group. Many common governance requirements can be handled using built-in definitions. However, it's likely that you will also need to create custom policy definitions to handle your specific requirements.

Custom policy definitions are saved to either a management group or a subscription and are inherited through the management group hierarchy. If a policy definition's save location is a management group, that policy definition is available to assign to any of that group's child management groups or subscriptions.

Since the policies required to support the governance MVP are meant to apply to all current subscriptions, the following business requirements will be implemented using a combination of built-in definitions and custom definitions created in the root management group:

1. Restrict the list of available role assignments to a set of built-in Azure roles authorized by your cloud governance team. This requires a custom policy definition.
2. Require the following tags on all resources: *Department/Billing Unit*, *Geography*, *Data Classification*, *Criticality*, *SLA*, *Environment*, *Application Archetype*, *Application*, and *Application Owner*. This can be handled using the `Require specified tag` built-in definition.
3. Require that the `Application` tag for resources should match the name of the relevant resource group. This can be handled using the "Require tag and its value" built-in definition.

For information on defining custom policies see the Azure Policy documentation. For guidance and examples of custom policies, consult the Azure Policy samples site and the associated GitHub repository.

Assign Azure Policy and RBAC roles using Azure Blueprints

Azure policies can be assigned at the resource group, subscription, and management group level, and can be included in Azure Blueprints definitions. Although the policy requirements defined in this governance MVP apply to all current subscriptions, it's very likely that future deployments will require exceptions or alternative policies. As a result, assigning policy using management groups, with all child subscriptions inheriting these assignments, may not be flexible enough to support these scenarios.

Azure Blueprints allows consistent assignment of policy and roles, application of Resource Manager templates, and deployment of resource groups across multiple subscriptions. Like policy definitions, blueprint definitions are saved to management groups or subscriptions. The policy definitions are available through inheritance to any children in the management group hierarchy.

The cloud governance team has decided that enforcement of required Azure Policy and RBAC assignments across subscriptions will be implemented through Azure Blueprints and associated artifacts:

1. In the root management group, create a blueprint definition named `governance-baseline`.
2. Add the following blueprint artifacts to the blueprint definition:
   a. Policy assignments for the custom Azure Policy definitions defined at the management group root.
   b. Resource group definitions for any groups required in subscriptions created or governed by the Governance MVP.
   c. Standard role assignments required in subscriptions created or governed by the Governance MVP.
3. Publish the blueprint definition.
4. Assign the `governance-baseline` blueprint definition to all subscriptions.

See the Azure Blueprints documentation for more information on creating and using blueprint definitions.

Secure hybrid VNet

Specific subscriptions often require some level of access to on-premises resources. This is common in migration scenarios or dev scenarios where dependent resources reside in the on-premises datacenter.

Until trust in the cloud environment is fully established it's important to tightly control and monitor any allowed communication between the on-premises environment and cloud workloads, and that the on-premises network is secured against potential unauthorized access from cloud-based resources. To support these scenarios, the governance MVP adds the following best practices:

1. Establish a cloud secure hybrid VNet.

a. The VPN reference architecture establishes a pattern and deployment model for creating a VPN Gateway in Azure.
b. Validate that on-premises security and traffic management mechanisms treat connected cloud networks as untrusted. Resources and services hosted in the cloud should only have access to authorized on-premises services.
c. Validate that the local edge device in the on-premises datacenter is compatible with Azure VPN Gateway requirements and is configured to access the public internet.
d. Note that VPN tunnels should not be considered production ready circuits for anything but the most simple workloads. Anything beyond a few simple workloads requiring on-premises connectivity should use Azure ExpressRoute.

2. In the root management group, create a second blueprint definition named `secure-hybrid-vnet`.
   a. Add the Resource Manager template for the VPN Gateway as an artifact to the blueprint definition.
   b. Add the Resource Manager template for the virtual network as an artifact to the blueprint definition.
   c. Publish the blueprint definition.
3. Assign the `secure-hybrid-vnet` blueprint definition to any subscriptions requiring on-premises connectivity. This definition should be assigned in addition to the `governance-baseline` blueprint definition.

One of the biggest concerns raised by IT security and traditional governance teams is the risk that early stage cloud adoption will compromise existing assets. The above approach allows cloud adoption teams to build and migrate hybrid solutions, with reduced risk to on-premises assets. As trust in the cloud environment increases, later evolutions may remove this temporary solution.

> **NOTE**
> The above is a starting point to quickly create a baseline governance MVP. This is only the beginning of the governance journey. Further evolution will be needed as the company continues to adopt the cloud and takes on more risk in the following areas:
>
> - Mission-critical workloads
> - Protected data
> - Cost management
> - Multicloud scenarios
>
> Moreover, the specific details of this MVP are based on the example journey of a fictional company, described in the articles that follow. We highly recommend becoming familiar with the other articles in this series before implementing this best practice.

Incremental governance improvements

Once this MVP has been deployed, additional layers of governance can be quickly incorporated into the environment. Here are some ways to improve the MVP to meet specific business needs:

- Security baseline for protected data
- Resource configurations for mission-critical applications
- Controls for cost management
- Controls for incremental multicloud improvement

What does this guidance provide?

In the MVP, practices and tools from the Deployment Acceleration discipline are established to quickly apply corporate policy. In particular, the MVP uses Azure Blueprints, Azure Policy, and Azure management groups to apply a few basic corporate policies, as defined in the narrative for this fictional company. Those corporate policies are applied using Azure Resource Manager templates and Azure policies to establish a small baseline for

identity and security.



Incremental improvements to governance practices

Over time, this governance MVP will be used to incrementally improve governance practices. As adoption advances, business risk grows. Various disciplines within the Cloud Adoption Framework governance model will adapt to manage those risks. Later articles in this series discuss the changes in corporate policy affecting the fictional company. These changes happen across four disciplines:

- The Identity Baseline discipline, as migration dependencies change in the narrative.
- The Cost Management discipline, as adoption scales.
- The Security Baseline discipline, as protected data is deployed.
- The Resource Consistency discipline, as IT operations begins supporting mission-critical workloads.



Next steps

Now that you're familiar with the governance MVP and the forthcoming governance changes, read the supporting narrative for additional context.

Read the supporting narrative

Governance guide for complex enterprises: The supporting narrative

3/6/2021 • 4 minutes to read • Edit Online

The following narrative establishes a use case for governance during complex enterprise's cloud adoption journey. Before acting on the recommendations in the guide, it's important to understand the assumptions and reasoning that are reflected in this narrative. Then you can better align the governance strategy to your organization's cloud adoption journey.

Back story

Customers are demanding a better experience when interacting with this company. The current experience caused market erosion and led to the board to hire a chief digital officer (CDO). The CDO is working with marketing and sales to drive a digital transformation that will power improved experiences. Additionally, several business units recently hired data scientists to farm data and improve many of the manual experiences through learning and prediction. IT is supporting these efforts where it can. There are "shadow IT" activities occurring that fall outside of needed governance and security controls.

The IT organization is also facing its own challenges. Finance is planning continued reductions in the IT budget over the next five years, leading to some necessary spending cuts starting this year. Conversely, GDPR and other data sovereignty requirements are forcing IT to invest in assets in additional countries to localize data. Two of the existing datacenters are overdue for hardware refreshes, causing further problems with employee and customer satisfaction. Three more datacenters require hardware refreshes during the execution of the five-year plan. The CFO is pushing the CIO to consider the cloud as an alternative for those datacenters, to free up capital expenses.

The CIO has innovative ideas that could help the company, but she and her teams are limited to fighting fires and controlling costs. At a luncheon with the CDO and one of the business unit leaders, the cloud migration conversation generated interest from the CIO's peers. The three leaders aim to support each other using the cloud to achieve their business objectives, and they have begun the exploration and planning stages of cloud adoption.

Business characteristics

The company has the following business profile:

- Sales and operations span multiple geographic areas with global customers in multiple markets.
- The business grew through acquisition and operates across three business units based on the target customer base. Budgeting is a complex matrix across business units and functions.
- The business views most of IT as a capital drain or a cost center.

Current state

Here is the current state of the company's IT and cloud operations:

- IT operates more than 20 privately owned datacenters around the globe.
- Due to organic growth and multiple geographies, there are a few IT teams that have unique data sovereignty and compliance requirements that impact a single business unit operating within a specific

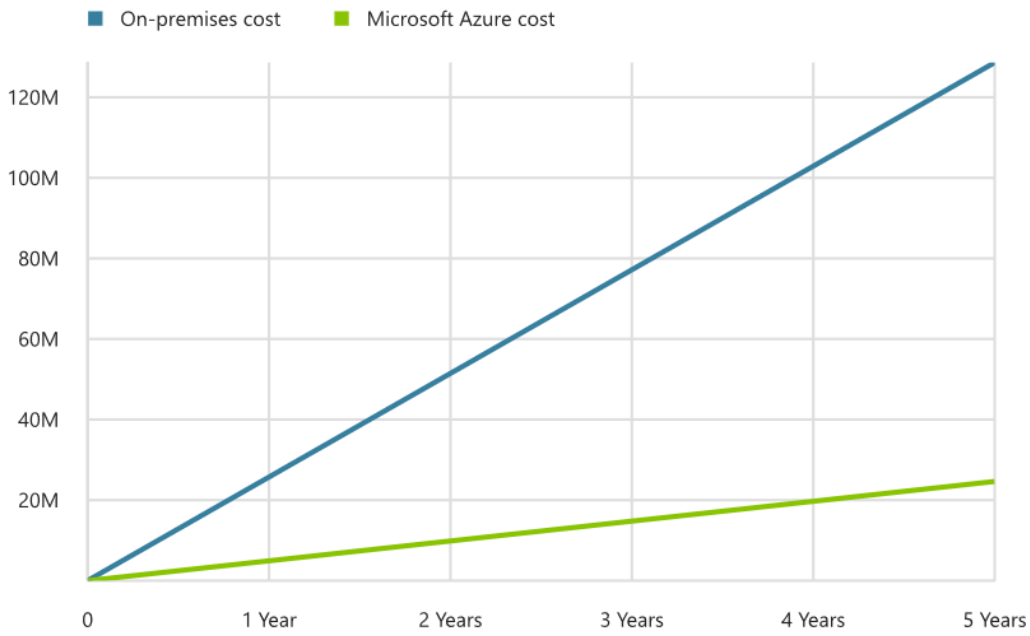
geography.

- Each datacenter is connected by a series of regional leased lines, creating a loosely coupled global WAN.
- IT entered the cloud by migrating all end-user email accounts to Microsoft 365. This migration was completed more than six months ago. Since then, only a few IT assets have been deployed to the cloud.
- The CDO's primary development team is working in a dev/test capacity to learn about cloud-native capabilities.
- One business unit is experimenting with big data in the cloud. The BI team inside of IT is participating in that effort.
- The existing IT governance policy states that personal customer data and financial data must be hosted on assets owned directly by the company. This policy blocks cloud adoption for any mission-critical applications or protected data.
- IT investments are controlled largely by capital expense. Those investments are planned yearly and often include plans for ongoing maintenance, as well as established refresh cycles of three to five years depending on the datacenter.
- Most investments in technology that don't align to the annual plan are addressed by shadow IT efforts. Those efforts are usually managed by business units and funded through the business unit's operating expenses.

Future state

The following changes are anticipated over the next several years:

- The CIO is leading an effort to modernize the policy on personal and financial data to support future goals. Two members of the IT governance team have visibility into this effort.
- The CIO wants to use the cloud migration as a forcing function to improve consistency and stability across business units and geographies. The future state must respect any external compliance requirements that would require deviation from standard approaches by specific IT teams.
- If the early experiments in application development and BI show leading indicators of success, they would each like to release small-scale production solutions to the cloud in the next 24 months.
- The CIO and CFO have assigned an architect and the vice president of infrastructure to create a cost analysis and feasibility study. These efforts will determine whether the company can and should move 5,000 assets to the cloud over the next 36 months. A successful migration would allow the CIO to eliminate two datacenters, reducing costs by over $100m USD during the five-year plan. If three to four datacenters can experience similar results, the budget will be back in the black, giving the CIO budget to support more innovative initiatives.



- Along with this cost savings, the company plans to change the management of some IT investments by repositioning the committed capital expense as an operating expense within IT. This change will provide greater cost control that IT can use to accelerate other planned efforts.

Next steps

The company has developed a corporate policy to shape the governance implementation. The corporate policy drives many of the technical decisions.

Review the initial corporate policy

Governance guide for complex enterprises: Initial corporate policy behind the governance strategy

3/6/2021 • 5 minutes to read • Edit Online

The following corporate policy defines the initial governance position that's the starting point for this guide. This article defines early-stage risks, initial policy statements, and early processes to enforce policy statements.

> **NOTE**
> The corporate policy is not a technical document, but it drives many technical decisions. The governance MVP described in the overview ultimately derives from this policy. Before implementing a governance MVP, your organization should develop a corporate policy based on your own objectives and business risks.

Cloud governance team

The CIO recently held a meeting with the IT governance team to understand the history of the personal data and mission-critical policies and review the effect of changing those policies. The CIO also discussed the overall potential of the cloud for IT and the company.

After the meeting, two members of the IT governance team requested permission to research and support the cloud planning efforts. Recognizing the need for governance and an opportunity to limit shadow IT, the director of IT governance supported this idea. With that, the cloud governance team was born. Over the next several months, they will inherit the cleanup of many mistakes made during exploration in the cloud from a governance perspective. This will earn them the moniker of *cloud custodians*. In later iterations, this guide will show how their roles change over time.

Objective

The initial objective is to establish a foundation for governance agility. An effective Governance MVP allows the governance team to stay ahead of cloud adoption and implement guardrails as the adoption plan changes.

Business risks

The company is at an early stage of cloud adoption, experimenting and building proofs of concept. Risks are now relatively low, but future risks are likely to have a significant impact. There is little definition around the final state of the technical solutions to be deployed to the cloud. In addition, the cloud readiness of IT employees is low. A foundation for cloud adoption will help the team safely learn and grow.

Future-proofing: There is a risk of not empowering growth, but also a risk of not providing the right protections against future risks.

An agile yet robust governance approach is needed to support the board's vision for corporate and technical growth. Failure to implement such a strategy will slow technical growth, potentially risking current and future market share growth. The impact of such a business risk is unquestionably high. However, the role IT will play in those potential future states is unknown, making the risk associated with current IT efforts relatively high. That said, until more concrete plans are aligned, the business has a high tolerance for risk.

This business risk can be broken down tactically into several technical risks:

- Well-intended corporate policies could slow transformation efforts or break critical business processes, if not

considered within a structured approval flow.
- The application of governance to deployed assets could be difficult and costly.
- Governance may not be properly applied across an application or workload, creating gaps in security.
- With so many teams working in the cloud, there is a risk of inconsistency.
- Costs may not properly align to business units, teams, or other budgetary management units.
- The use of multiple identities to manage various deployments could lead to security issues.
- Despite current policies, there is a risk that protected data could be mistakenly deployed to the cloud.

Tolerance indicators

The current risk tolerance is high and the appetite for investing in cloud governance is low. As such, the tolerance indicators act as an early warning system to trigger the investment of time and energy. If the following indicators are observed, it would be wise to advance the governance strategy.

- **Cost Management discipline:** Scale of deployment exceeds 1,000 assets to the cloud, or monthly spending exceeds $10,000 USD per month.
- **Identity Baseline discipline:** Inclusion of applications with legacy or third-party multi-factor authentication requirements.
- **Security Baseline discipline:** Inclusion of protected data in defined cloud adoption plans.
- **Resource Consistency discipline:** Inclusion of any mission-critical applications in defined cloud adoption plans.

Policy statements

The following policy statements establish the requirements needed to remediate the defined risks. These policies define the functional requirements for the governance MVP. Each will be represented in the implementation of the governance MVP.

Cost Management:

- For tracking purposes, all assets must be assigned to an application owner within one of the core business functions.
- When cost concerns arise, additional governance requirements will be established with the finance team.

Security Baseline:

- Any asset deployed to the cloud must have an approved data classification.
- No assets identified with a protected level of data may be deployed to the cloud, until sufficient requirements for security and governance can be approved and implemented.
- Until minimum network security requirements can be validated and governed, cloud environments are seen as perimeter networks and should meet similar connection requirements to other datacenters or internal networks.

Resource Consistency:

- Because no mission-critical workloads are deployed at this stage, there are no SLA, performance, or BCDR requirements to be governed.
- When mission-critical workloads are deployed, additional governance requirements will be established with IT operations.

Identity Baseline:

- All assets deployed to the cloud should be controlled using identities and roles approved by current governance policies.

- All groups in the on-premises Active Directory infrastructure that have elevated privileges should be mapped to an approved RBAC role.

Deployment Acceleration:

- All assets must be grouped and tagged according to defined grouping and tagging strategies.
- All assets must use an approved deployment model.
- Once a governance foundation has been established for a cloud provider, any deployment tooling must be compatible with the tools defined by the governance team.

Processes

No budget has been allocated for ongoing monitoring and enforcement of these governance policies. Because of that, the cloud governance team has improvised ways to monitor adherence to policy statements.

- **Education:** The cloud governance team is investing time to educate the cloud adoption teams on the governance guides that support these policies.
- **Deployment reviews:** Before deploying any asset, the cloud governance team will review the governance guide with the cloud adoption teams.

Next steps

This corporate policy prepares the cloud governance team to implement the governance MVP as the foundation for adoption. The next step is to implement this MVP.

Best practices explained

Governance guide for complex enterprises: Best practices explained

5/18/2021 • 11 minutes to read • Edit Online

The governance guide begins with a set of initial corporate policies. These policies are used to establish a minimum viable product (MVP) for governance that reflects best practices.

In this article, we discuss the high-level strategies that are required to create a governance MVP. The core of the governance MVP is the Deployment Acceleration discipline. The tools and patterns applied at this stage will enable the incremental improvements needed to expand governance in the future.

Governance MVP (initial governance foundation)

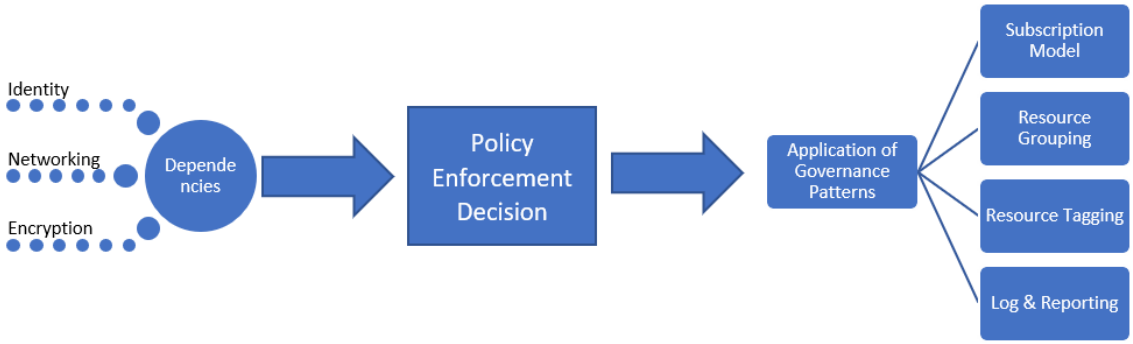
Rapid adoption of governance and corporate policy is achievable, thanks to a few simple principles and cloud-based governance tooling. These are the first of the three governance disciplines to approach in any governance process. Each discipline will be explained further on in this article.

To establish the starting point, this article discusses the high-level strategies behind the Security Baseline, Identity Baseline, and Deployment Acceleration disciplines that are required to create a governance MVP. The MVP serves as the foundation for all cloud adoption.



Implementation process

The implementation of the governance MVP has dependencies on identity, security, and networking. Once the dependencies are resolved, the cloud governance team will decide a few aspects of governance. The decisions from the cloud governance team and from supporting teams will be implemented through a single package of enforcement assets.



This implementation can also be described using a simple checklist:

1. Solicit decisions regarding core dependencies: identity, network, and encryption.
2. Determine the pattern to be used during corporate policy enforcement.
3. Determine the appropriate governance patterns for resource consistency, resource tagging, and logging and reporting.
4. Implement the governance tools aligned to the chosen policy enforcement pattern to apply the dependent decisions and governance decisions.

Dependent decisions

The following decisions come from teams outside of the cloud governance team. The implementation of each will come from those same teams. However, the cloud governance team is responsible for implementing a solution to validate that those implementations are consistently applied.

Identity Baseline

Identity Baseline is the fundamental starting point for all governance. Before attempting to apply governance, identity must be established. The established identity strategy will then be enforced by the governance solutions. In this governance guide, the Identity Management team implements the Directory Synchronization pattern:

- RBAC will be provided by Azure Active Directory (Azure AD), using the directory synchronization or "Same Sign-On" that was implemented during company's migration to Microsoft 365. For implementation guidance, see Reference Architecture for Azure AD Integration.
- The Azure AD tenant will also govern authentication and access for assets deployed to Azure.

In the governance MVP, the governance team will enforce application of the replicated tenant through subscription governance tooling, discussed later in this article. In future iterations, the governance team could also enforce rich tooling in Azure AD to extend this capability.

Security Baseline: Networking

Software Defined Network is an important initial aspect of the Security Baseline. Establishing the governance MVP depends on early decisions from the Security Management team to define how networks can be safely configured.

Given the lack of requirements, IT security is playing it safe and requires a Cloud DMZ pattern. That means governance of the Azure deployments themselves will be very light.

- Azure subscriptions may connect to an existing datacenter via VPN, but must follow all existing on-premises IT governance policies regarding connection of a perimeter network to protected resources. For implementation guidance regarding VPN connectivity, see On-premises network connected to Azure using a VPN gateway.
- Decisions regarding subnet, firewall, and routing are currently being deferred to each application/workload lead.
- Additional analysis is required before releasing of any protected data or mission-critical workloads.

In this pattern, cloud networks can only connect to on-premises resources over an existing VPN that is compatible with Azure. Traffic over that connection will be treated like any traffic coming from a perimeter network. Additional considerations may be required on the on-premises edge device to securely handle traffic from Azure.

The cloud governance team has proactively invited members of the networking and IT security teams to regular meetings, in order to stay ahead of networking demands and risks.

Security Baseline: Encryption

Encryption is another fundamental decision within the Security Baseline discipline. Because the company

currently does not yet store any protected data in the cloud, the Security Team has decided on a less aggressive pattern for encryption. At this point, a cloud-native pattern for encryption is suggested but not required of any development team.

- No governance requirements have been set regarding the use of encryption, because the current corporate policy does not permit mission-critical or protected data in the cloud.
- Additional analysis will be required before releasing any protected data or mission-critical workloads.

Policy enforcement

The first decision to make regarding Deployment Acceleration is the pattern for enforcement. In this narrative, the governance team decided to implement the Automated Enforcement pattern.

- Azure Security Center will be made available to the security and identity teams to monitor security risks. Both teams are also likely to use Security Center to identify new risks and improve corporate policy.
- RBAC is required in all subscriptions to govern authentication enforcement.
- Azure Policy will be published to each management group and applied to all subscriptions. However, the level of policies being enforced will be very limited in this initial Governance MVP.
- Although Azure management groups are being used, a relatively simple hierarchy is expected.
- Azure Blueprints will be used to deploy and update subscriptions by applying RBAC requirements, Resource Manager Templates, and Azure Policy across management groups.

Apply the dependent patterns

The following decisions represent the patterns to be enforced through the policy enforcement strategy above:

Identity Baseline. Azure Blueprints will set RBAC requirements at a subscription level to ensure that consistent identity is configured for all subscriptions.

Security Baseline: Networking. The cloud governance team maintains a Resource Manager template for establishing a VPN gateway between Azure and the on-premises VPN device. When an application team requires a VPN connection, the cloud governance team will apply the gateway Resource Manager template via Azure Blueprints.

Security Baseline: Encryption. At this point, no policy enforcement is required in this area. This will be revisited during later iterations.

Application of governance-defined patterns

The cloud governance team will be responsible for the following decisions and implementations. Many will require inputs from other teams, but the cloud governance team is likely to own both the decision and implementation. The following sections outline the decisions made for this use case and details of each decision.

Subscription design

The decision on what subscription design to use determines how Azure subscriptions get structured and how Azure management groups will be used to efficiently manage access, policies, and compliance of these subscription. In this narrative, the governance team has chosen a mixed subscription strategy.

- As new requests for Azure resources arise, a *department* should be established for each major business unit in each operating geography. Within each of the departments, *subscriptions* should be created for each application archetype.
- An application archetype is a means of grouping applications with similar needs. Common examples include:
  - Applications with protected data, governed applications (such as HIPAA or FedRAMP).

    - Low-risk applications.
    - Applications with on-premises dependencies.
    - SAP or other mainframe applications in Azure.
    - Applications that extend on-premises SAP or mainframe applications.

    Each organization has unique needs based on data classifications and the types of applications that support the business. Dependency mapping of the digital estate can help define the application archetypes in an organization.

- A common naming convention should be adopted as part of the subscription design, based on the above.

Resource consistency

Resource consistency decisions determine the tools, processes, and effort required to ensure Azure resources are deployed, configured, and managed consistently within a subscription. In this narrative, deployment consistency has been chosen as the primary resource consistency pattern.

- Resource groups are created for applications using the lifecycle approach. Everything that is created, maintained, and retired together should reside a single resource group. For more information, see the resource consistency decision guide.
- Azure Policy should be applied to all subscriptions from the associated management group.
- As part of the deployment process, Azure resource consistency templates for the resource group should be stored in source control.
- Each resource group is associated with a specific workload or application based on the lifecycle approach described above.
- Azure management groups enable updating governance designs as corporate policy matures.
- Extensive implementation of Azure Policy could exceed the team's time commitments and may not provide a great deal of value at this time. A simple default policy should be created and applied to each management group to enforce the small number of current cloud governance policy statements. This policy will define the implementation of specific governance requirements. Those implementations can then be applied across all deployed assets.

> **IMPORTANT**
> Any time a resource in a resource group no longer shares the same lifecycle, it should be moved to another resource group. Examples include common databases and networking components. While they may serve the application being developed, they may also serve other purposes and should therefore exist in other resource groups.

Resource tagging

Resource tagging decisions determine how metadata is applied to Azure resources within a subscription to support operations, management, and accounting purposes. In this narrative, the accounting pattern has been chosen as the default model for resource tagging.

- Deployed assets should be tagged with values for:
    - Department or billing unit
    - Geography
    - Data classification
    - Criticality
    - SLA
    - Environment
    - Application archetype
    - Application
    - Application owner

- These values, along with the Azure management group and subscription associated with a deployed asset, will drive governance, operations, and security decisions.

Logging and reporting

Logging and reporting decisions determine how your store log data and how the monitoring and reporting tools that keep IT staff informed on operational health are structured. In this narrative a hybrid monitoring pattern for logging and reporting is suggested, but not required of any development team at this point.

- No governance requirements are currently set regarding the specific data points to be collected for logging or reporting purposes. This is specific to this fictional narrative and should be considered an antipattern. Logging standards should be determined and enforced as soon as possible.
- Additional analysis is required before the release of any protected data or mission-critical workloads.
- Before supporting protected data or mission-critical workloads, the existing on-premises operational monitoring solution must be granted access to the workspace used for logging. Applications are required to meet security and logging requirements associated with the use of that tenant, if the application is to be supported with a defined SLA.

Incremental of governance processes

Some of the policy statements cannot or should not be controlled by automated tooling. Other policies will require periodic effort from IT security and on-premises identity baseline teams. The cloud governance team will need to oversee the following processes to implement the last eight policy statements:

Corporate policy changes: The cloud governance team will make changes to the governance MVP design to adopt the new policies. The value of the governance MVP is that it will allow for the automatic enforcement of the new policies.

Adoption acceleration: The cloud governance team has been reviewing deployment scripts across multiple teams. They've maintained a set of scripts that serve as deployment templates. Those templates can be used by the cloud adoption teams and DevOps teams to more quickly define deployments. Each script contains the requirements for enforcing governance policies, and additional effort from cloud adoption engineers is not needed. As the curators of these scripts, they can implement policy changes more quickly. Additionally, they're viewed as accelerators of adoption. This ensures consistent deployments without strictly enforcing adherence.

Engineer training: The cloud governance team offers bimonthly training sessions and has created two videos for engineers. Both resources help engineers get up to speed quickly on the governance culture and how deployments are performed. The team is adding training assets to demonstrate the difference between production and nonproduction deployments, which helps engineers understand how the new policies affect adoption. This ensures consistent deployments without strictly enforcing adherence.

Deployment planning: Before deploying any asset containing protected data, the cloud governance team will be responsible for reviewing deployment scripts to validate governance alignment. Existing teams with previously approved deployments will be audited using programmatic tooling.

Monthly audit and reporting: Each month, the cloud governance team runs an audit of all cloud deployments to validate continued alignment to policy. When deviations are discovered, they're documented and shared with the cloud adoption teams. When enforcement doesn't risk a business interruption or data leak, the policies are automatically enforced. At the end of the audit, the cloud governance team compiles a report for the cloud strategy team and each cloud adoption team to communicate overall adherence to policy. The report is also stored for auditing and legal purposes.

Quarterly policy review: Each quarter, the cloud governance team and the cloud strategy team to review audit results and suggest changes to corporate policy. Many of those suggestions are the result of continuous improvements and the observation of usage patterns. Approved policy changes are integrated into governance tooling during subsequent audit cycles.

Alternative patterns

If any of the patterns chosen in this governance guide don't align with the reader's requirements, alternatives to each pattern are available:

- Encryption patterns
- Identity patterns
- Logging and reporting patterns
- Policy enforcement patterns
- Resource consistency patterns
- Resource tagging patterns
- Software Defined Networking patterns
- Subscription design patterns

Next steps

Once this guidance is implemented, each cloud adoption team can proceed with a solid governance foundation. At the same time, the cloud governance team will work to continually update the corporate policies and governance disciplines.

Both teams will use the tolerance indicators to identify the next set of improvements needed to continue supporting cloud adoption. The next step for this company is incremental improvement of their governance baseline to support applications with legacy or third-party multi-factor authentication requirements.

Improve the Identity Baseline discipline

Governance guide for complex enterprises: Improve the Identity Baseline discipline

3/6/2021 • 4 minutes to read • Edit Online

This article advances the narrative by adding identity baseline controls to the governance MVP.

Advancing the narrative

The business justification for the cloud migration of the two datacenters was approved by the CFO. During the technical feasibility study, several roadblocks were discovered:

- Protected data and mission-critical applications represent 25% of the workloads in the two datacenters. Neither can be eliminated until the current governance policies regarding sensitive personal data and mission-critical applications have been modernized.
- 7% of the assets in those datacenters are not cloud-compatible. They will be moved to an alternate datacenter before termination of the datacenter contract.
- 15% of the assets in the datacenter (750 virtual machines) have a dependency on legacy authentication or third-party multi-factor authentication.
- The VPN connection that connects existing datacenters and Azure does not offer sufficient data transmission speeds or latency to migrate the volume of assets within the two-year timeline to retire the datacenter.

The first two roadblocks are being managed in parallel. This article will address the resolution of the third and fourth roadblocks.

Expand the cloud governance team

The cloud governance team is expanding. Given the need for additional support regarding identity management, a systems administrator from the identity baseline team now participates in a weekly meeting to keep the existing team members aware of changes.

Changes in the current state

The IT team has approval to move forward with the plans of the CIO and CFO to retire two datacenters. The team is concerned that 750 (15%) of the assets in those datacenters will have to be moved somewhere other than the cloud.

Incrementally improve the future state

The new future state plans require a more robust identity baseline solution to migrate the 750 virtual machines with legacy authentication requirements. Beyond these two datacenters, this challenge is expected to affect similar percentages of assets in other datacenters.

The future state now also requires a connection from the cloud provider to the company's MPLS/leased-line solution.

The changes to current and future state expose new risks that will require new policy statements.

Changes in tangible risks

Business interruption during migration. Migration to the cloud creates a controlled, time-bound risk that can be managed. Moving aging hardware to another part of the world is much higher risk. A mitigation strategy is needed to avoid interruptions to business operations.

Existing identity dependencies. Dependencies on existing authentication and identity services may delay or prevent the migration of some workloads to the cloud. Failure to return the two datacenters on time will incur millions of dollars in datacenter lease fees.

This business risk can be expanded into a few technical risks:

- Legacy authentication might not be available in the cloud, limiting deployment of some applications.
- The current third-party multi-factor authentication solution might not be available in the cloud, limiting deployment of some applications.
- Retooling or moving could create outages or add costs.
- The speed and stability of the VPN might impede migration.
- Traffic entering the cloud could cause security issues in other parts of the global network.

Incremental improvement of the policy statements

The following changes to policy will help remediate the new risks and guide implementation.

- The chosen cloud provider must offer a means of authenticating via legacy methods.
- The chosen cloud provider must offer a means of authentication with the current third-party multi-factor authentication solution.
- A high-speed private connection should be established between the cloud provider and the company's telco provider, connecting the cloud provider to the global network of datacenters.
- Until sufficient security requirements are established, no inbound public traffic may access company assets hosted in the cloud. All ports are blocked from any source outside of the global WAN.

Incremental improvement of best practices

The governance MVP design changes to include new Azure policies and an implementation of Active Directory on a virtual machine. Together, these two design changes fulfill the new corporate policy statements.

Here are the new best practices:

- **Secure hybrid virtual network blueprint:** The on-premises side of the hybrid network should be configured to allow communication between the following solution and the on-premises Active Directory servers. This best practice requires a perimeter network to enable Active Directory Domain Services across network boundaries.
- **Azure Resource Manager templates:**
  1. Define an NSG to block external traffic and allow internal traffic.
  2. Deploy two Active Directory virtual machines in a load-balanced pair based on a golden image. On first boot, that image runs a PowerShell script to join the domain and register with domain services. For more information, see Extend Active Directory Domain Services (AD DS) to Azure.
- Azure Policy: Apply the NSG to all resources.
- Azure Blueprints:
  1. Create a blueprint named `active-directory-virtual-machines`.
  2. Add each of the Active Directory templates and policies to the blueprint.
  3. Publish the blueprint to any applicable management group.
  4. Apply the blueprint to any subscription requiring legacy or third-party multi-factor authentication.
  5. The instance of Active Directory running in Azure can now be used as an extension of the on-premises Active Directory solution, allowing it to integrate with the existing multi-factor authentication tool and provide claims-based authentication, both through existing Active Directory functionality.

Conclusion

Adding these changes to the governance MVP helps remediate many of the risks in this article, allowing each cloud adoption team to quickly move past this roadblock.

Next steps

As cloud adoption continues and delivers additional business value, risks and cloud governance needs will also change. The following are a few changes that may occur. For this fictional company, the next trigger is the inclusion of protected data in the cloud adoption plan. This change requires additional security controls.

Improve the Security Baseline discipline

Governance guide for complex enterprises: Improve the Security Baseline discipline

3/6/2021 • 13 minutes to read • Edit Online

This article advances the narrative by adding security controls that support moving protected data to the cloud.

Advancing the narrative

The CIO has spent months collaborating with colleagues and the company's legal staff. A management consultant with expertise in cybersecurity was engaged to help the existing IT security and IT governance teams draft a new policy regarding protected data. The group was able to foster board support to replace the existing policy, allowing sensitive personal and financial data to be hosted by approved cloud providers. This required adopting a set of security requirements and a governance process to verify and document adherence to those policies.

For the past 12 months, the cloud adoption teams have cleared most of the 5,000 assets from the two datacenters to be retired. The 350 incompatible assets were moved to an alternate datacenter. Only the 1,250 virtual machines that contain protected data remain.

Changes in the cloud governance team

The cloud governance team continues to change along with the narrative. The two founding members of the team are now among the most respected cloud architects in the company. The collection of configuration scripts has grown as new teams tackle innovative new deployments. The cloud governance team has also grown. Most recently, members of the IT operations team have joined cloud governance team activities to prepare for cloud operations. The cloud architects who helped foster this community are seen both as cloud guardians and cloud accelerators.

While the difference is subtle, it's an important distinction when building a governance-focused IT culture. A cloud custodian cleans up the messes made by innovative cloud architects, and the two roles have natural friction and opposing objectives. A cloud guardian helps keep the cloud safe, so other cloud architects can move more quickly with fewer messes. A cloud accelerator performs both functions but is also involved in the creation of templates to accelerate deployment and adoption, becoming an innovation accelerator as well as a defender of the Five Disciplines of Cloud Governance.

Changes in the current state

In the previous phase of this narrative, the company had begun the process of retiring two datacenters. This ongoing effort includes migrating some applications with legacy authentication requirements, which required incremental improvements to the Identity Baseline discipline, described in the previous article.

Since then, some things have changed that will affect governance:

- Thousands of IT and business assets have been deployed to the cloud.
- The application development team has implemented a continuous integration and continuous deployment (CI/CD) pipeline to deploy a cloud-native application with an improved user experience. That application doesn't interact with protected data yet, so it isn't production ready.
- The business intelligence team within IT actively curates data in the cloud from logistics, inventory, and third-party data. This data is being used to drive new predictions, which could shape business processes. Those predictions and insights are not actionable until customer and financial data can be integrated into the data platform.
- The IT team is making progress on the CIO and CFO plans to retire two datacenters. Almost 3,500 of the

assets in the two datacenters have been retired or migrated.

- The policies regarding sensitive personal and financial data have been modernized. The new corporate policies are contingent on the implementation of related security and governance policies. Teams are still stalled.

Incrementally improve the future state

- Early experiments from the application development and BI teams have shown potential improvements in customer experiences and data-driven decisions. Both teams would like to expand adoption of the cloud over the next 18 months by deploying those solutions to production.
- IT has developed a business justification to migrate five more datacenters to Azure, which will further decrease IT costs and provide greater business agility. While smaller in scale, the retirement of those datacenters is expected to double the total cost savings.
- Capital expense and operating expense budgets have approved to implement the required security and governance policies, tools, and processes. The expected cost savings from the datacenter retirement are more than enough to pay for this new initiative. IT and business leadership are confident this investment will accelerate the realization of returns in other areas. The grassroots cloud governance team became a recognized team with dedicated leadership and staffing.
- Collectively, the cloud adoption teams, the cloud governance team, the IT security team, and the IT governance team will implement security and governance requirements to allow cloud adoption teams to migrate protected data into the cloud.

Changes in tangible risks

Data breach: There is an inherent increase in liabilities related to data breaches when adopting any new data platform. Technicians adopting cloud technologies have increased responsibilities to implement solutions that can decrease this risk. A robust security and governance strategy must be implemented to ensure those technicians fulfill those responsibilities.

This business risk can be expanded into several technical risks:

1. Mission-critical applications or protected data might be deployed unintentionally.
2. Protected data might be exposed during storage due to poor encryption decisions.
3. Unauthorized users might access protected data.
4. External intrusion could result in access to protected data.
5. External intrusion or denial of service attacks could cause a business interruption.
6. Organization or employment changes could allow for unauthorized access to protected data.
7. New exploits might create opportunities for intrusion or unauthorized access.
8. Inconsistent deployment processes might result in security gaps that could lead to data leaks or interruptions.
9. Configuration drift or missed patches might result in unintended security gaps that could lead to data leaks or interruptions.
10. Disparate edge devices might increase network operations costs.
11. Disparate device configurations might lead to oversights in configuration and compromises in security.
12. The cybersecurity team insists there is a risk of vendor lock-in from generating encryption keys on a single cloud provider's platform. While this claim is unsubstantiated, it was accepted by the team for the time being.

Incremental improvement of the policy statements

The following changes to policy will help remediate the new risks and guide implementation. The list looks long, but the adoption of these policies may be easier than it would appear.

1. All deployed assets must be categorized by criticality and data classification. Classifications are to be

reviewed by the cloud governance team and the application before deployment to the cloud.

2. Applications that store or access protected data are to be managed differently than those that don't. At a minimum, they should be segmented to avoid unintended access of protected data.
3. All protected data must be encrypted when at rest.
4. Elevated permissions in any segment containing protected data should be an exception. Any such exceptions will be recorded with the cloud governance team and audited regularly.
5. Network subnets containing protected data must be isolated from any other subnets. Network traffic between protected data subnets will be audited regularly.
6. No subnet containing protected data can be directly accessed over the public internet or across datacenters. Access to these subnets must be routed through intermediate subnets. All access into these subnets must come through a firewall solution that can perform packet scanning and blocking functions.
7. Governance tooling must audit and enforce network configuration requirements defined by the security management team.
8. Governance tooling must limit VM deployment to approved images only.
9. Whenever possible, node configuration management should apply policy requirements to the configuration of any guest operating system. Node configuration management should respect the existing investment in group policy objects (GPO) for resource configuration.
10. Governance tooling will audit that automatic updates are enabled on all deployed assets. When possible, automatic updates will be enforced. When not enforced by tooling, node-level violations must be reviewed with operational management teams and remediated in accordance with operations policies. Assets that are not automatically updated must be included in processes owned by IT operations.
11. Creation of new subscriptions or management groups for any mission-critical applications or protected data requires a review from the cloud governance team to ensure proper blueprint assignment.
12. A least-privilege access model will be applied to any subscription that contains mission-critical applications or protected data.
13. The cloud vendor must be capable of integrating encryption keys managed by the existing on-premises solution.
14. The cloud vendor must be capable of supporting the existing edge device solution and any required configurations to protect any publicly exposed network boundary.
15. The cloud vendor must be capable of supporting a shared connection to the global WAN, with data transmission routed through the existing edge device solution.
16. Trends and exploits that could affect cloud deployments should be reviewed regularly by the security team to provide updates to Security Baseline tools used in the cloud.
17. Deployment tooling must be approved by the cloud governance team to ensure ongoing governance of deployed assets. 18. Deployment scripts must be maintained in a central repository accessible by the cloud governance team for periodic review and auditing.
18. Governance processes must include audits at the point of deployment and at regular cycles to ensure consistency across all assets.
19. Deployment of any applications that require customer authentication must use an approved identity provider that is compatible with the primary identity provider for internal users. 1. Cloud governance processes must include quarterly reviews with identity baseline teams to identify malicious actors or usage patterns that should be prevented by cloud asset configuration.

Incremental improvement of best practices

This section modifies the governance MVP design to include new Azure policies and an implementation of Azure Cost Management + Billing. Together, these two design changes will fulfill the new corporate policy statements.

The new best practices fall into two categories: corporate IT (hub) and cloud adoption (spoke).

Establishing a corporate IT hub and spoke subscription to centralize the security baseline: In this

best practice, the existing governance capacity is wrapped by a hub and spoke topology with shared services, with a few key additions from the cloud governance team.

1. Azure DevOps repository. Create a repository in Azure DevOps to store and version all relevant Azure Resource Manager templates and scripted configurations.
2. Hub and spoke template:
   a. The guidance in the hub and spoke topology with shared services reference architecture can be used to generate Resource Manager templates for the assets required in a corporate IT hub.
   b. Using those templates, this structure can be made repeatable, as part of a central governance strategy.
   c. In addition to the current reference architecture, a network security group template should be created to capture any port blocking or allow-listing requirements for the virtual network to host the firewall. This network security group differs from prior groups, because it will be the first network security group to allow public traffic into a virtual network.
3. Create Azure policies. Create a policy named `hub NSG enforcement` to enforce the configuration of the network security group assigned to any virtual network created in this subscription. Apply the built-in policies for guest configuration as follows:
   a. Audit that Windows web servers are using secure communication protocols.
   b. Audit that password security settings are set correctly inside Linux and Windows machines.
4. Create the corporate IT blueprint.
   a. Create an Azure blueprint named `corporate-it-subscription`.
   b. Add the hub and spoke templates and `hub NSG enforcement` policy.
5. Expanding on initial management group hierarchy.
   a. For each management group that has requested support for protected data, the `corporate-it-subscription-blueprint` blueprint provides an accelerated hub solution.
   b. Because management groups in this fictional example include a regional hierarchy in addition to a business unit hierarchy, this blueprint will be deployed in each region.
   c. For each region in the management group hierarchy, create a subscription named `corporate IT subscription`.
   d. Apply the `corporate-it-subscription-blueprint` blueprint to each regional instance.
   e. This will establish a hub for each business unit in each region. Note: further cost savings could be achieved by sharing hubs across business units in each region.
6. Integrate group policy objects (GPO) through Desired State Configuration (DSC):
   a. Convert GPO to DSC. The Microsoft baseline management project in GitHub can accelerate this effort. Be sure to store DSC in the repository in parallel with Resource Manager templates.
   b. Deploy Azure Automation State Configuration to any instances of the corporate IT subscription. Azure Automation can be used to apply DSC to VMs deployed in supported subscriptions within the management group.
   c. The current roadmap aims to enable custom guest configuration policies. When that feature is released, the use of Azure Automation in this best practice will no longer be required.

Applying additional governance to a cloud adoption subscription (spoke): Building on the `corporate IT subscription`, minor changes to the governance MVP applied to each subscription dedicated to the support of application archetypes can produce rapid improvement.

In prior iterative changes to the best practice, we defined network security groups to block public traffic and allow internal traffic. Additionally, the Azure blueprint temporarily created DMZ and Active Directory capabilities. In this iteration, we will tweak those assets a bit, creating a new version of the Azure blueprint.

1. Network peering template. This template will peer the virtual network in each subscription with the hub virtual network in the corporate IT subscription.
   a. The reference architecture from the prior section, hub and spoke topology with shared services,

generated a Resource Manager template for enabling virtual network peering.

   b. That template can be used as a guide to modify the DMZ template from the prior governance iteration.
   c. We are now adding virtual network peering to the DMZ virtual network that was previously connected to the local edge device over VPN.
   d. The VPN should also be removed from this template as well to ensure no traffic is routed directly to the on-premises datacenter, without passing through the corporate IT subscription and firewall solution. You could also set this VPN as a failover circuit in the event of an ExpressRoute circuit outage.
   e. Additional network configuration is required by Azure Automation to apply DSC to hosted VMs.
2. Modify the network security group. Block all public **and** direct on-premises traffic in the network security group. The only inbound traffic should be coming through the virtual network peer in the corporate IT subscription.
   a. In the prior iteration, a network security group was created blocking all public traffic and allowing all internal traffic. Now we want to shift this network security group a bit.
   b. The new network security group configuration should block all public traffic, along with all traffic from the local datacenter.
   c. Traffic entering this virtual network should only come from the virtual network on the other side of the virtual network peer.
3. Azure Security Center implementation:
   a. Configure Azure Security Center for any management group that contains protected data classifications.
   b. Set automatic provisioning to on by default to ensure patching compliance.
   c. Establish OS security configurations. IT security to define the configuration.
   d. Support IT security in the initial use of Azure Security Center. Transition use of Security Center to IT security, but maintain access for governance continuous improvement purposes.
   e. Create a Resource Manager template reflecting the changes required for Azure Security Center configuration within a subscription.
4. Update Azure Policy for all subscriptions.
   a. Audit and enforce criticality and data classification across all management groups and subscriptions to identify any subscriptions with protected data classifications.
   b. Audit and enforce use of approved OS images only.
   c. Audit and enforce guest configurations based on security requirements for each node.
5. Update Azure Policy for all subscriptions that contains protected data classifications.
   a. Audit and enforce use of standard roles only.
   b. Audit and enforce application of encryption for all storage accounts and files at rest on individual nodes.
   c. Audit and enforce the application of the new version of the DMZ network security group.
   d. Audit and enforce use of approved network subnet and virtual network per network interface.
   e. Audit and enforce the limitation of user-defined routing tables.
6. Azure blueprint:
   a. Create an Azure blueprint named `protected-data`.
   b. Add the virtual network peer, network security group, and Azure Security Center templates to the blueprint.
   c. Ensure the template for Active Directory from the previous iteration is **not** included in the blueprint. Any dependencies on Active Directory will be provided by the corporate IT subscription.
   d. Terminate any existing Active Directory VMs deployed in the previous iteration.
   e. Add the new policies for protected data subscriptions.
   f. Publish the blueprint to any management group that will host protected data.

g. Apply the new blueprint to each affected subscription along with existing blueprints.

Conclusion

Adding these processes and changes to the governance MVP helps remediate many of the risks associated with security governance. Together, they add the network, identity, and security monitoring tools needed to protect data.

Next steps

As cloud adoption continues and delivers additional business value, risks and cloud governance needs also change. For the fictional company in this guide, the next step is to support mission-critical workloads. This is the point when resource consistency controls are needed. A critical part of the security governance narrative will be to review best practices that Microsoft has built for security. The Azure Security Benchmark (ASB) provides prescriptive best practices and recommendations to help improve the security of workloads, data, and services on Azure. Read here.

Improve the Resource Consistency discipline

Governance guide for complex enterprises: Improve the Resource Consistency discipline

3/6/2021 • 6 minutes to read • Edit Online

This article advances the narrative by adding resource consistency controls to the governance MVP to support mission-critical applications.

Advancing the narrative

The cloud adoption teams have met all requirements to move protected data. With those applications come SLA commitments to the business and need for support from IT operations. Right behind the team migrating the two datacenters, multiple application development and BI teams are ready to begin launching new solutions into production. IT operations is new to cloud operations and needs to quickly integrate existing operational processes.

Changes in the current state

- IT is actively moving production workloads with protected data into Azure. Some low-priority workloads are serving production traffic. More can be cut over as soon as IT operations signs off on readiness to support the workloads.
- The application development teams are ready for production traffic.
- The BI team is ready to integrate predictions and insights into the systems that run operations for the three business units.

Incrementally improve the future state

- IT operations is new to cloud operations and needs to quickly integrate existing operational processes.
- The changes to current and future state expose new risks that will require new policy statements.

Changes in tangible risks

Business interruption: There is an inherent risk of any new platform causing interruptions to mission-critical business processes. The IT operations team and the teams executing on various cloud adoptions are relatively inexperienced with cloud operations. This increases the risk of interruption and must be remediated and governed.

This business risk can be expanded into several technical risks:

1. Misaligned operational processes might lead to outages that can't be detected or mitigated quickly.
2. External intrusion or denial of service attacks might cause a business interruption.
3. Mission-critical assets might not be properly discovered and therefore not properly operated.
4. Undiscovered or mislabeled assets might not be supported by existing operational management processes.
5. Configuration of deployed assets might not meet performance expectations.
6. Logging might not be properly recorded and centralized to allow for remediation of performance issues.
7. Recovery policies may fail or take longer than expected.
8. Inconsistent deployment processes might result in security gaps that could lead to data leaks or interruptions.
9. Configuration drift or missed patches might result in unintended security gaps that could lead to data leaks or interruptions.
10. Configuration might not enforce the requirements of defined SLAs or committed recovery requirements.

11. Deployed operating systems or applications might not meet OS and application hardening requirements.
12. There is a risk of inconsistency due to multiple teams working in the cloud.

Incremental improvement of the policy statements

The following changes to policy will help remediate the new risks and guide implementation. The list looks long, but the adoption of these policies may be easier than it would appear.

1. All deployed assets must be categorized by criticality and data classification. Classifications are to be reviewed by the cloud governance team and the application owner before deployment to the cloud.
2. Subnets containing mission-critical applications must be protected by a firewall solution capable of detecting intrusions and responding to attacks.
3. Governance tooling must audit and enforce network configuration requirements defined by the security baseline team.
4. Governance tooling must validate that all assets related to mission-critical applications or protected data are included in monitoring for resource depletion and optimization.
5. Governance tooling must validate that the appropriate level of logging data is being collected for all mission-critical applications or protected data.
6. Governance process must validate that backup, recovery, and SLA adherence are properly implemented for mission-critical applications and protected data.
7. Governance tooling must limit virtual machine deployment to approved images only.
8. Governance tooling must enforce that automatic updates are **prevented** on all deployed assets that support mission-critical applications. Violations must be reviewed with operational management teams and remediated in accordance with operations policies. Assets that are not automatically updated must be included in processes owned by IT operations to quickly and effectively update those servers.
9. Governance tooling must validate tagging related to cost, criticality, SLA, application, and data classification. All values must align to predefined values managed by the cloud governance team.
10. Governance processes must include audits at the point of deployment and at regular cycles to ensure consistency across all assets.
11. Trends and exploits that could affect cloud deployments should be reviewed regularly by the security team to provide updates to Security Baseline tools used in the cloud.
12. Before release into production, all mission-critical applications and protected data must be added to the designated operational monitoring solution. Assets that cannot be discovered by the chosen IT operations tooling cannot be released for production use. Any changes required to make the assets discoverable must be made to the relevant deployment processes to ensure assets will be discoverable in future deployments.
13. When discovered, asset sizing is to be validated by operational management teams to validate that the asset meets performance requirements.
14. Deployment tooling must be approved by the cloud governance team to ensure ongoing governance of deployed assets.
15. Deployment scripts must be maintained in central repository accessible by the cloud governance team for periodic review and auditing.
16. Governance review processes must validate that deployed assets are properly configured in alignment with SLA and recovery requirements.

Incremental improvement of best practices

This section of the article will improve the governance MVP design to include new Azure policies and an implementation of Azure Cost Management + Billing. Together, these two design changes will fulfill the new corporate policy statements.

Following the experience of this fictional example, it's assumed that the protected data changes have already

occurred. Building on that best practice, the following will add operational monitoring requirements, readying a subscription for mission-critical applications.

Corporate IT subscription: Add the following to the corporate IT subscription, which acts as a hub.

1. As an external dependency, the cloud operations team will need to define operational monitoring tooling, business continuity and disaster recovery (BCDR) tooling, and automated remediation tooling. The cloud governance team can then support necessary discovery processes.
   a. In this use case, the cloud operations team chose Azure Monitor as the primary tool for monitoring mission-critical applications.
   b. The team also chose Azure Site Recovery as the primary BCDR tooling.
2. Azure Site Recovery implementation.
   a. Define and deploy Azure Site Recovery vault for backup and recovery processes.
   b. Create an Azure resource management template for creation of a vault in each subscription.
3. Azure Monitor implementation.
   a. Once a mission-critical subscription is identified, a Log Analytics workspace can be created.

Individual cloud adoption subscription: The following will ensure that each subscription is discoverable by the monitoring solution and ready to be included in BCDR practices.

1. Azure Policy for mission-critical nodes:
   a. Audit and enforce use of standard roles only.
   b. Audit and enforce application of encryption for all storage accounts.
   c. Audit and enforce use of approved network subnet and virtual network per network interface.
   d. Audit and enforce the limitation of user-defined routing tables.
   e. Audit and enforce the deployment of Log Analytics agents for Windows and Linux virtual machines.
2. Azure Blueprints:
   a. Create a blueprint named `mission-critical-workloads-and-protected-data`. This blueprint will apply assets in addition to the protected data blueprint.
   b. Add the new Azure policies to the blueprint.
   c. Apply the blueprint to any subscription that is expected to host a mission-critical application.

Conclusion

Adding these processes and changes to the governance MVP helps remediate many of the risks associated with resource governance. Together, they add the recovery, sizing, and monitoring controls necessary to empower cloud-aware operations.

Next steps

As cloud adoption grows and delivers additional business value, the risks and cloud governance needs will also change. For the fictional company in this guide, the next trigger is when the scale of deployment exceeds 1,000 assets to the cloud or monthly spending exceeds $10,000 USD per month. At this point, the cloud governance team adds cost management controls.

Improve the Cost Management discipline

Governance guide for complex enterprises: Improve the Cost Management discipline

3/6/2021 • 4 minutes to read • Edit Online

This article advances the narrative by adding cost controls to the minimum viable product (MVP) governance.

Advancing the narrative

Adoption has grown beyond the tolerance indicator defined in the governance MVP. The increases in spending now justifies an investment of time from the cloud governance team to monitor and control spending patterns.

As a clear driver of innovation, IT is no longer seen primarily as a cost center. As the IT organization delivers more value, the CIO and CFO agree that the time is right to shift the role IT plays in the company. Among other changes, the CFO wants to test a direct pay approach to cloud accounting for the Canadian branch of one of the business units. One of the two retired datacenters was exclusively hosted assets for that business unit's Canadian operations. In this model, the business unit's Canadian subsidiary will be billed directly for the operating expenses related to the hosted assets. This model allows IT to focus less on managing someone else's spending and more on creating value. Before this transition can begin cost management tooling needs to be in place.

Changes in the current state

In the previous phase of this narrative, the IT team was actively moving production workloads with protected data into Azure.

Since then, some things have changed that will affect governance:

- 5,000 assets have been removed from the two datacenters flagged for retirement. Procurement and IT security are now deprovisioning the remaining physical assets.
- The application development teams have implemented CI/CD pipelines to deploy some cloud-native applications, significantly affecting customer experiences.
- The BI team has created aggregation, curation, insight, and prediction processes driving tangible benefits for business operations. Those predictions are now empowering creative new products and services.

Incrementally improve the future state

Cost monitoring and reporting should be added to the cloud solution. Reporting should tie direct operating expenses to the functions that are consuming the cloud costs. Additional reporting should allow IT to monitor spending and provide technical guidance on cost management. For the Canadian branch, the department will be billed directly.

Changes in risk

Budget control: There is an inherent risk that self-service capabilities will result in excessive and unexpected costs on the new platform. Governance processes for monitoring costs and mitigating ongoing cost risks must be in place to ensure continued alignment with the planned budget.

This business risk can be expanded into a few technical risks:

- There is a risk of actual costs exceeding the plan.
- Business conditions change. When they do, there will be cases when a business function needs to consume more cloud services than expected, leading to spending anomalies. There is a risk that these additional costs

will be considered overages as opposed to a required adjustment to the plan. If successful, the Canadian experiment should help remediate this risk.
- There is a risk of systems being overprovisioned, resulting in excess spending.

Changes to the policy statements

The following changes to policy will help remediate the new risks and guide implementation.

- All cloud costs should be monitored against plan on a weekly basis by the cloud governance team. Reporting on deviations between cloud costs and plan is to be shared with IT leadership and finance monthly. All cloud costs and plan updates should be reviewed with IT leadership and finance monthly.
- All costs must be allocated to a business function for accountability purposes.
- Cloud assets should be continually monitored for optimization opportunities.
- Cloud governance tooling must limit asset sizing options to an approved list of configurations. The tooling must ensure that all assets are discoverable and tracked by the cost monitoring solution.
- During deployment planning, any required cloud resources associated with the hosting of production workloads should be documented. This documentation will help refine budgets and prepare additional automation tools to prevent the use of more expensive options. During this process consideration should be given to different discounting tools offered by the cloud provider, such as Azure Reserved VM Instances or license cost reductions.
- All application owners are required to attend trained on practices for optimizing workloads to better control cloud costs.

Incremental improvement of best practices

This section of the article will improve the governance MVP design to include new Azure policies and an implementation of Azure Cost Management + Billing. Together, these two design changes will fulfill the new corporate policy statements.

1. Make changes in the Azure EA portal to bill the Department Administrator for the Canadian deployment.
2. Implement Azure Cost Management + Billing.
    a. Establish the right level of access scope to align with the subscription pattern and resource grouping pattern. Assuming alignment with the governance MVP defined in prior articles, this would require **enrollment account scope** access for the cloud governance team executing on high-level reporting. Additional teams outside of governance, like the Canadian procurement team, will require **resource group scope** access.
    b. Establish a budget in Azure Cost Management + Billing.
    c. Review and act on initial recommendations. Create a recurring process to support the reporting process.
    d. Configure and execute Azure Cost Management + Billing reporting, both initial and recurring.
3. Update Azure Policy.
    a. Audit tagging, management group, subscription, and resource group values to identify any deviation.
    b. Establish SKU size options to limit deployments to SKUs listed in deployment planning documentation.

Conclusion

Adding the above processes and changes to the governance MVP helps remediate many of the risks associated with cost governance. Together, they create the visibility, accountability, and optimization needed to control costs.

Next steps

As cloud adoption grows and delivers additional business value, risks and cloud governance needs will also change. For this fictional company, the next step is using this governance investment to manage multiple clouds.

Multicloud improvement

Governance guide for complex enterprises: Multicloud improvement

3/6/2021 • 3 minutes to read • Edit Online

Advancing the narrative

Microsoft recognizes that customers may adopt multiple clouds for specific purposes. The fictional company in this guide is no exception. In parallel with their Azure adoption journey, business success has led to the acquisition of a small but complementary business. That business is running all of their IT operations on a different cloud provider.

This article describes how things change when integrating the new organization. For purposes of the narrative, we assume this company has completed each of the governance iterations outlined in this governance guide.

Changes in the current state

In the previous phase of this narrative, the company had begun to implement cost controls and cost monitoring, as cloud spending becomes part of the company's regular operating expenses.

Since then, some things have changed that will affect governance:

- Identity is controlled by an on-premises instance of Active Directory. Hybrid identity is facilitated through replication to Azure Active Directory.
- IT operations or cloud operations are largely managed by Azure Monitor and related automation capabilities.
- Business continuity and disaster recovery (BCDR) is controlled by Azure Recovery Services vaults.
- Azure Security Center is used to monitor security violations and attacks.
- Azure Security Center and Azure Monitor are both used to monitor governance of the cloud.
- Azure Blueprints, Azure Policy, and management groups are used to automate compliance to policy.

Incrementally improve the future state

The goal is to integrate the acquisition company into existing operations wherever possible.

Changes in tangible risks

Business acquisition cost: Acquisition of the new business is estimated to be profitable in approximately five years. Because of the slow rate of return, the board wants to control acquisition costs, as much as possible. There is a risk of cost control and technical integration conflicting with one another.

This business risk can be expanded into a few technical risks:

- There is a risk of cloud migration producing additional acquisition costs.
- There is also a risk of the new environment not being properly governed or resulting in policy violations.

Incremental improvement of the policy statements

The following changes to policy will help remediate the new risks and guide implementation.

- All assets in a secondary cloud must be monitored through existing operational management and security monitoring tools.
- All organizational units must be integrated into the existing identity provider.
- The primary identity provider should govern authentication to assets in the secondary cloud.

Incremental improvement of best practices

This section of the article improves the governance MVP design to include new Azure policies and an implementation of Azure Cost Management + Billing. Together, these two design changes will fulfill the new corporate policy statements.

1. Connect the networks. Executed by networking and IT security, supported by governance.
    a. Adding a connection from the MPLS or leased-line provider to the new cloud will integrate networks. Adding routing tables and firewall configurations will control access and traffic between the environments.
2. Consolidate identity providers. Depending on the workloads being hosted in the secondary cloud, there are a variety of options to identity provider consolidation. The following are a few examples:
    a. For applications that authenticate using OAuth 2, users in the Active Directory in the secondary cloud could simply be replicated to the existing Azure AD tenant.
    b. On the other extreme, federation between the two on-premises identity providers, would allow users from the new Active Directory domains to be replicated to Azure.
3. Add assets to Azure Site Recovery.
    a. Azure Site Recovery was built as a hybrid and multicloud tool from the beginning.
    b. Virtual machines in the secondary cloud might be able to be protected by the same Azure Site Recovery processes used to protect on-premises assets.
4. Add assets to Azure Cost Management + Billing.
    a. Azure Cost Management + Billing was built as a multicloud tool from the beginning.
    b. Virtual machines in the secondary cloud might be compatible with Azure Cost Management + Billing for some cloud providers. Additional costs may apply.
5. Add assets to Azure Monitor.
    a. Azure Monitor was built as a hybrid cloud tool from the beginning.
    b. Virtual machines in the secondary cloud might be compatible with Azure Monitor agents, allowing them to be included in Azure Monitor for operational monitoring.
6. Governance enforcement tools.
    a. Governance enforcement is cloud-specific.
    b. The corporate policies established in the governance guide are not cloud-specific. While the implementation may vary from cloud to cloud, the policy statements can be applied to the secondary provider.

Multicloud adoption should be contained to where it's required based on technical needs or specific business requirements. As multicloud adoption grows, so does complexity and security risks.

Next steps

In many large enterprises, the Five Disciplines of Cloud Governance can be blockers to adoption. The next article has some additional thoughts on making governance a team sport to help ensure long-term success in the cloud.

Multiple layers of governance

Governance guide for complex enterprises: Multiple layers of governance

3/6/2021 • 3 minutes to read • Edit Online

When large enterprises require multiple layers of governance, there are greater levels of complexity that must be factored into the governance MVP and later governance improvements.

A few common examples of such complexities include:

- Distributed governance functions.
- Corporate IT supporting business unit IT organizations.
- Corporate IT supporting geographically distributed IT organizations.

This article explores some ways to navigate this type of complexity.

Large enterprise governance is a team sport

Large established enterprises often have teams or employees who focus on the disciplines mentioned throughout this guide. This guide demonstrates one approach to making governance a team sport.

In many large enterprises, the Five Disciplines of Cloud Governance can be blockers to adoption. Developing cloud expertise in identity, security, operations, deployments, and configuration across an enterprise takes time. Holistically implementing IT governance policy and IT security can slow innovation by months or even years. Balancing the business need to innovate and the governance need to protect existing resources is delicate.

The inherent capabilities of the cloud can remove blockers to innovation but increase risks. In this governance guide, we showed how the example company created guardrails to manage the risks. Rather than tackling each of the disciplines required to protect the environment, the cloud governance team leads a risk-based approach to govern what could be deployed, while the other teams build the necessary cloud maturities. Most importantly, as each team reaches cloud maturity, governance applies their solutions holistically. As each team matures and adds to the overall solution, the cloud governance team can open stage gates, allowing additional innovation and adoption to thrive.

This model illustrates the growth of a partnership between the cloud governance team and existing enterprise teams (security, IT governance, networking, identity, and others). The guide starts with the governance MVP and grows to a holistic end state through governance iterations.

Requirements to supporting such a team sport

The first requirement of a multilayer governance model is to understand of the governance hierarchy. Answering the following questions will help you to understand the general governance hierarchy:

- How is cloud accounting (billing for cloud services) allocated across business units?
- How are governance responsibilities allocated across corporate IT and each business unit?
- What types of environments do each of those units of IT manage?

Central governance of a distributed governance hierarchy

Tools like management groups allow corporate IT to create a hierarchy structure that matches the governance hierarchy. Tools like Azure Blueprints can apply assets to different layers of that hierarchy. Azure Blueprints can be versioned and various versions can be applied to management groups, subscriptions, or resource groups.

Each of these concepts is described in more detail in the governance MVP.

The important aspect of each of these tools is the ability to apply multiple blueprints to a hierarchy. This allows governance to be a layered process. The following is one example of this hierarchical application of governance:

- **Corporate IT:** Corporate IT creates a set of standards and policies that apply to all cloud adoption. This is materialized in a *baseline* blueprint. Corporate IT then owns the management group hierarchy, ensuring that a version of the baseline is applied to all subscriptions in the hierarchy.
- **Regional or business unit IT:** Various IT teams can apply an additional layer of governance by creating their own blueprint. Those blueprints would create additive policies and standards. Once developed, corporate IT could apply those blueprints to the applicable nodes within the management group hierarchy.
- **Cloud adoption teams:** Detailed decisions and implementation about applications or workloads can be made by each cloud adoption team, within the context of governance requirements. The team can also request additional Azure resource consistency templates to accelerate adoption efforts.

The details regarding governance implementation at each level will require coordination between each team. The governance MVP and governance improvements outlined in this guide can aid in aligning that coordination.

Evaluate corporate policy

3/6/2021 • 2 minutes to read • Edit Online

Any change to business processes or technology platforms introduces risk to the business. Cloud governance teams, whose members are sometimes known as cloud custodians, are tasked with mitigating these risks with minimal interruption to adoption or innovation efforts.

But cloud governance requires more than technical implementation. Subtle changes in the corporate narrative or corporate policies can affect adoption efforts significantly. Before implementation, it's important to look beyond IT while defining corporate policy.



Figure 1: Visual of corporate policy and the Five Disciplines of Cloud Governance.

Define corporate policy

Defining corporate policy focuses on identifying and mitigating business risks regardless of the cloud platform. Healthy cloud governance strategy begins with sound corporate policy. The following three-step process guides the iterative development of such policies.

	Business risk: Investigate current cloud adoption plans and data classification to identify risks to the business. Work with the business to balance risk tolerance and mitigation costs.
	Policy and compliance: Evaluate risk tolerance to inform minimally invasive policies that govern cloud adoption and manage risks. In some industries, third-party compliance affects initial policy creation.
	Processes: The pace of adoption and innovation activities will naturally create policy violations. Executing relevant processes will aid in monitoring and enforcing adherence to policies.

Next steps

Learn how to prepare your corporate policy for the cloud.

Prepare your corporate policy for the cloud

Prepare corporate IT policy for the cloud

3/6/2021 • 4 minutes to read • Edit Online

Cloud governance is the product of an ongoing adoption effort over time, as a true lasting transformation doesn't happen overnight. Attempting to deliver complete cloud governance before addressing key corporate policy changes using a fast aggressive method seldom produces the desired results. Instead we recommend an incremental approach.

What is different about our Cloud Adoption Framework is the purchasing cycle and how it can enable authentic transformation. Since there is not a big capital expenditure acquisition requirement, engineers can begin experimentation and adoption sooner. In most corporate cultures, elimination of the capital expense barrier to adoption can lead to tighter feedback loops, organic growth, and incremental execution.

The shift to cloud adoption requires a shift in governance. In many organizations, corporate policy transformation allows for improved governance and higher rates of adherence through incremental policy changes and automated enforcement of those changes, powered by newly defined capabilities that you configure with your cloud service provider.

This article outlines key activities that can help you shape your corporate policies to enable an expanded governance model.

Define corporate policy to mature cloud governance

In traditional governance and incremental governance, corporate policy creates the working definition of governance. Most IT governance actions seek to implement technology to monitor, enforce, operate, and automate those corporate policies. Cloud governance is built on similar concepts.

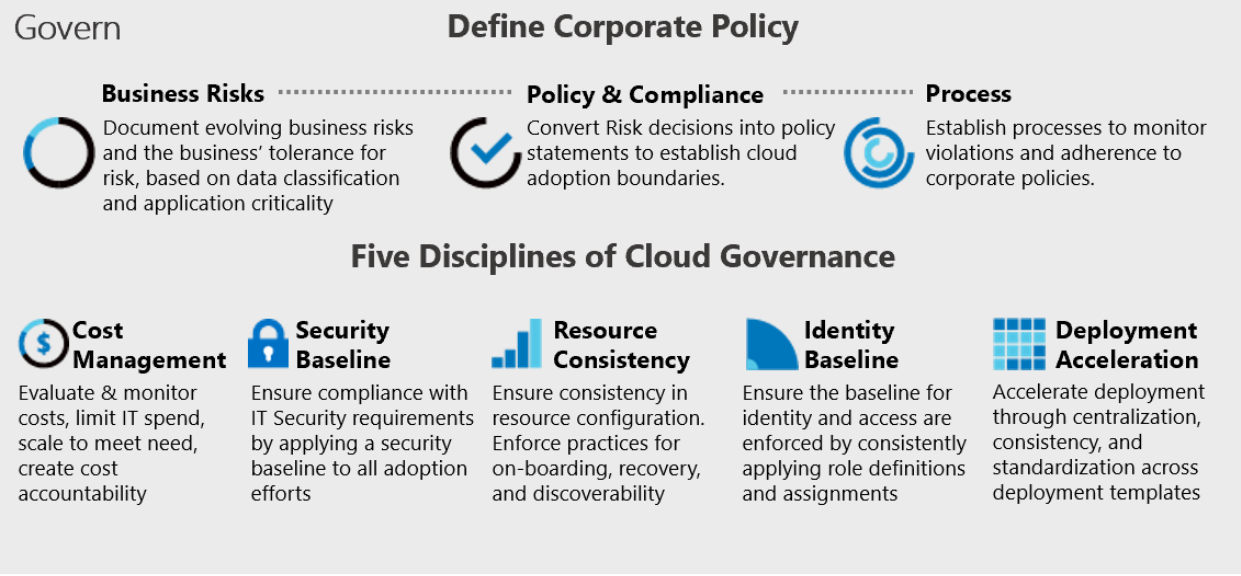


Figure 1: Corporate governance and governance disciplines.

The image above demonstrates the interactions between business risk, policy and compliance, and monitor and enforce to create a governance strategy. Followed by the Five Disciplines of Cloud Governance to realize your strategy.

Review existing policies

In the image above, the governance strategy (risk, policy and compliance, monitor and enforce) starts with

recognizing business risks. Understanding how business risk changes in the cloud is the first step to creating a lasting cloud governance strategy. Working with your business units to gain an accurate gauge of the business's tolerance for risk, helps you understand what level of risks need to be remediated. Your understanding of new risks and acceptable tolerance can fuel a review of existing policies, in order to determine the required level of governance that is appropriate for your organization.

> **TIP**
> If your organization is governed by third-party compliance, one of the biggest business risks to consider may be a risk of adherence to regulatory compliance. This risk often cannot be remediated, and instead may require a strict adherence. Be sure to understand your third-party compliance requirements before beginning a policy review.

An incremental approach to cloud governance

An incremental approach to cloud governance assumes that it's unacceptable to exceed the business's tolerance for risk. Instead, it assumes that the role of governance is to accelerate business change, help engineers understand architecture guidelines, and ensure that business risks are regularly communicated and remediated. Alternatively, the traditional role of governance can become a barrier to adoption by engineers or by the business as a whole.

With an incremental approach to cloud governance, there is sometimes a natural friction between teams building new business solutions and teams protecting the business from risks. In this model, those two teams can become peers working in increments or sprints. As peers, the cloud governance team and the cloud adoption teams begin to work together to expose, evaluate, and remediate business risks. This effort can create a natural means of reducing friction and building collaboration between teams.

Minimum viable product (MVP) for policy

The first step in an emerging partnership between your cloud governance and adoption teams is an agreement regarding the policy MVP. Your MVP for cloud governance should acknowledge that business risks are small in the beginning, but will likely grow as your organization adopts more cloud services over time.

For example, the business risk is small for a business deploying five VMs that don't contain any high business impact (HBI) data. Later in the cloud adoption process, when the number reaches 1,000 VMs and the business is starting to move HBI data, the business risk grows.

Policy MVP attempts to define a required foundation for policies needed to deploy the first *x* VMs or the first *x* number of applications, where *x* is a small yet meaningful quantity of the units being adopted. This policy set requires few constraints, but would contain the foundational aspects needed to quickly grow from one incremental cloud adoption effort to the next. Through incremental policy development, this governance strategy would grow over time. Through slow subtle shifts, the policy MVP would grow into feature parity with the outputs of the policy review exercise.

Incremental policy growth

Incremental policy growth is the key mechanism to growing policy and cloud governance over time. It's also the key requirement to adopting an incremental model to governance. For this model to work well, the governance team must be committed to an ongoing allocation of time at each sprint, in order to evaluate and implement changing governance disciplines.

Sprint time requirements: At the beginning of each iteration, each cloud adoption team creates a list of assets to be migrated or adopted in the current increment. The cloud governance team is expected to allow sufficient time to review the list, validate data classifications for assets, evaluate any new risks associated with each asset, update architecture guidelines, and educate the team on the changes. These commitments

commonly require 10-30 hours per sprint. It's also expected for this level of involvement to require at least one dedicated employee to manage governance in a large cloud adoption effort.

Release time requirements: At the beginning of each release, the cloud adoption teams and the cloud strategy team should prioritize a list of applications or workloads to be migrated in the current iteration, along with any business change activities. Those data points allow the cloud governance team to understand new business risks early. That allows time to align with the business and gauge the business's tolerance for risk.

Next steps

Effective cloud governance strategy begins with understanding business risk.

Understand business risk

Understand business risk during cloud migration

3/6/2021 • 4 minutes to read • Edit Online

An understanding of business risk is one of the most important elements of any cloud transformation. Risk drives policy, and it influences monitoring and enforcement requirements. Risk heavily influences how we manage the digital estate, on-premises or in the cloud.

Relativity of risk

Risk is relative. A small company with a few IT assets, in a closed building has little risk. Add users and an internet connection with access to those assets, the risk is intensified. When that small company grows to Fortune 500 status, the risks are exponentially greater. As revenue, business process, employee counts, and IT assets accumulate, risks increase and coalesce. IT assets that aid in generating revenue are at tangible risk of stopping that revenue stream in the event of an outage. Every moment of downtime equates to losses. Likewise, as data accumulates, the risk of harming customers grows.

In the traditional on-premises world, IT governance teams focus on assessing risks, creating processes to manage those risks, and deploying systems to ensure remediation measures are successfully implemented. These efforts work to balance risks required to operate in a connected, modern business environment.

Understand business risks in the cloud

During a transformation, the same relative risks exist.

- During early experimentation, a few assets are deployed with little to no relevant data. The risk is small.
- When the first workload is deployed, risk goes up a little. This risk is easily remediated by choosing an inherently low risk application with a small user base.
- As more workloads come online, risks change at each release. New applications go live and risks change.
- When a company brings the first 10 or 20 applications online, the risk profile is much different than when the 1000th application goes into production in the cloud.

The assets that accumulated in the traditional, on-premises estate likely accumulated over time. The maturity of the business and IT teams was likely growing in a similar fashion. That parallel growth can tend to create some unnecessary policy baggage.

During a cloud transformation, both the business and IT teams have an opportunity to reset those policies and build new with a matured mindset.

What is a business risk MVP?

A *minimum viable product (MVP)* is commonly used to define the smallest unit of something that can produce tangible value. In a business risk MVP, the cloud governance team starts with the assumption that some assets will be deployed to a cloud environment at some point in time. It's unknown what those assets are at the time, and the team may be unsure what types of data will be stored on those assets.

When planning for business risk, the cloud governance team could build for the worst case scenario and map every possible policy to the cloud. Identifying all potential business risks for all cloud usage scenarios can take considerable time and effort, potentially delaying the implementation of governance to your cloud workloads. This is not recommended, but is an option.

Conversely, an MVP approach can allow the team to define an initial starting point and set of assumptions that

would be true for most/all assets. This business risk MVP will support initial small scale or test cloud deployments, and then be used as a base for gradually identifying and remediating new risks as business needs arise or additional workloads are added to your cloud environment. This process allows you to apply governance throughout the cloud adoption process.

The following are a few basic examples of business risks that can be included as part of an MVP:

- All assets are at risk of being deleted (through error, mistake or maintenance).
- All assets are at risk of generating too much spending.
- All assets could be compromised by weak passwords or insecure settings.
- Any asset with open ports exposed to the internet are at risk of compromise.

The above examples are meant to establish MVP business risks as a theory. The actual list will be unique to every environment.

Once the business risk MVP is established, they can be converted to policies to remediate each risk.

Incremental risk mitigation

As your organization deploys more workloads to the cloud, development teams will make use of increasing amounts of cloud resources. At each iteration, new assets are created and staged. At each release, workloads are readied for production promotion. Each of these cycles has the potential to introduce previously unidentified business risks.

Assuming a business risk MVP is the starting point for your initial cloud adoption efforts, governance can mature in parallel with your increasing use of cloud resources. When the cloud governance team operates in parallel with cloud adoption teams, the growth of business risks can be addressed as they're identified, providing a stable ongoing model for developing governance maturity.

Each asset staged can easily be classified according to risk. Data classification documents can be built or created in parallel with staging cycles. Risk profile and exposure points can likewise be documented. Over time an extremely clear view of business risk will come into focus across the organization.

With each iteration, the cloud governance team can work with the cloud strategy team to quickly communicate new risks, mitigation strategies, tradeoffs, and potential costs. This empowers business participants and IT leaders to partner in mature, well-informed decisions. Those decisions then inform policy maturity. When required, the policy changes produce new work items for the maturity of core infrastructure systems. When changes to staged systems are required, the cloud adoption teams have ample time to make changes, while the business tests the staged systems and develops a user adoption plan.

This approach minimizes risks, while empowering the team to move quickly. It also ensures that risks are promptly addressed and resolved before deployment.

Next steps

Learn to evaluate risk tolerance during cloud adoption.

Evaluate risk tolerance

Evaluate risk tolerance

8/11/2021 • 8 minutes to read • Edit Online

Every business decision creates new risks. Making an investment in anything creates risk of losses. New products or services create risks of market failure. Changes to current products or services could reduce market share. Cloud transformation does not provide a magical solution to everyday business risk. To the contrary, connected solutions (cloud or on-premises) introduce new risks. Deploying assets to any network connected facility also expands the potential threat profile by exposing security weaknesses to a much broader, global community. Fortunately, cloud providers are aware of the changes, increases, and addition of risks. They invest heavily to reduce and manage those risks on the behalf of their customers.

This article is not focused on cloud risks. Instead it discusses the business risks associated with various forms of cloud transformation. Later in the article, the discussion shifts focus to discuss ways of understanding the business's tolerance for risk.

What business risks are associated with a cloud transformation?

True business risks are based on the details of specific transformations. Several common risks provide a conversation starter to understand business-specific risks.

> **IMPORTANT**
> Before reading the following, be aware that each of these risks can be managed. The goal of this article is to inform and prepare readers for more productive risk management discussions.

- **Data breach:** The top risk associated with any transformation is a data breach. Data leaks can cause significant damage to your company, leading to loss of customers, decrease in business, or even legal liability. Any changes to the way data is stored, processed, or used creates risk. Cloud transformations create a high degree of change regarding data management and analytics, so the risk should not be taken lightly. The Security Baseline discipline, data classification, and incremental rationalization can each help manage this risk.
- **Service disruption:** Business operations and customer experiences rely heavily on technical operations. Cloud transformations will create change in IT operations. In some organizations, that change is small and easily adjusted. In other organizations, these changes could require retooling, retraining, or new approaches to support cloud operations. The bigger the change, the bigger the potential impact on business operations and customer experience. Managing this risk will require the involvement of the business in transformation planning. Release planning and first workload selection in the incremental rationalization article discuss ways to choose workloads for transformation projects. The business's role in that activity is to communicate the business operations risk of changing prioritized workloads. Helping IT choose workloads that have a lower impact on operations will reduce the overall risk.
- **Budget control:** Cost models change in the cloud. This change can create risks associated with cost overruns or increases in the cost of goods sold (COGS), especially directly attributed operating expenses. When business works closely with IT, it's feasible to create transparency regarding costs and services consumed by various business units, programs, or projects. The Cost Management discipline provides examples of ways business and IT can partner on this topic.

The above are a few of the most common risks mentioned by customers. The cloud governance team and the cloud adoption teams can begin to develop a risk profile, as workloads are migrated and readied for production

release. Be prepared for conversations to define, refine, and manage risks based on the desired business outcomes and transformation effort.

Understand risk tolerance

Identifying risk is a fairly direct process. IT-related risks are generally standard across industries. Tolerance for these risks is specific to each organization. This is the point where business and IT conversations tend to get hung up. Each side of the conversation is essentially speaking a different language. The following comparisons and questions are designed to start conversations that help each party better understand and calculate risk tolerance.

Simple use case for comparison

To help understand risk tolerance, let's examine customer data. If a company in any industry posts customer data on an unsecured server, the technical risk of that data being compromised or stolen is roughly the same. Tolerance for that risk will vary wildly based on the nature and potential value of the data.

- Companies in healthcare and finance in the United States, are governed by rigid, third-party compliance requirements. It's assumed that personal data or healthcare-related data is extremely confidential. There are severe consequences for these types of companies, if they're involved in the risks scenario above. Their tolerance will be extremely low. Any customer data published inside or outside of the network must be governed by those third-party compliance policies.
- A gaming company whose customer data is limited to a user name, play times, and high scores is not as likely to suffer significant consequences beyond loss to reputation, if they engage in the risky behavior above. While any unsecured data is at risk, the impact of that risk is small. Therefore, the tolerance for risk in this case is high.
- A medium-sized enterprise that provides carpet-cleaning services to thousands of customers would fall in between these two tolerance extremes. Customer data may be more robust, containing details like addresses and phone numbers. Both are considered personal data and should be protected, but no specific governance requirement mandates that the data be secured. From an IT perspective, the answer is simple, secure the data. From a business perspective, it may not be as simple. The business would need more details before they could determine a level of tolerance for this risk.

The next section shares a few sample questions that could help the business determine a level of risk tolerance for the use case above or others.

Risk tolerance questions

This section lists conversation provoking questions in three categories: loss impact, probability of loss, and remediation costs. When business and IT partner to address each of these areas, the decision to expend effort on managing risks and the overall tolerance to a particular risk can easily be determined.

Loss impact: Questions to determine the impact of a risk. These questions can be difficult to answer. Quantifying the impact is best, but sometimes the conversation alone is enough to understand tolerance. Ranges are also acceptable, especially if they include assumptions that determined those ranges.

- Could this risk violate third-party compliance requirements?
- Could this risk violate internal corporate policies?
- Could this risk cause the loss of life, limb or property?
- Could this risk cost customers or market share? If so, can this cost be quantified?
- Could this risk create negative customer experiences? Are those experiences likely to affect sales or revenue?
- Could this risk create new legal liability? If so, is there a precedence for damage awards in these types of cases?

- Could this risk stop business operations? If so, how long would operations be down?
- Could this risk slow business operations? If so, how slow and how long?
- At this stage in the transformation is this a one-off risk or will it repeat?
- Does the risk increase or decrease in frequency as the transformation progresses?
- Does the risk increase or decrease in probability over time?
- Is the risk time sensitive in nature? Will the risk pass or get worse, if not addressed?

These basic questions will lead to many more. After exploring a healthy dialogue, it's suggested that the relevant risks be recorded and when possible quantified.

Risk remediation costs: Questions to determine the cost of removing or otherwise minimizing the risk. These questions can be fairly direct, especially when represented in a range.

- Is there a clear solution and what does it cost?
- Are there options for preventing or minimizing this risk? What is the range of costs for those solutions?
- What is needed from the business to select the best, clear solution?
- What is needed from the business to validate costs?
- What other benefits can come from the solution that would remove this risk?

These questions over simplify the technical solutions needed to manage or remove risks, but they communicate those solutions in ways the business can quickly integrate into a decision process.

Probability of loss: Questions to determine how likely it's that the risk will become a reality. This is the most difficult area to quantify. Instead it's suggested that the cloud governance team create categories for communicating probability, based on the supporting data. The following questions can help create categories that are meaningful to the team.

- Has any research been done regarding the likelihood of this risk being realized?
- Can the vendor provide references or statistics on the likelihood of an impact?
- Are there other companies in the relevant sector or vertical that have been hit by this risk?
- Look further, are there other companies in general that have been hit by this risk?
- Is this risk unique to something this company has done poorly?

After answering these questions along with questions as determined by the cloud governance team, groupings of probability will likely emerge. The following are a few grouping samples to help get started:

- **No indication:** Not enough research has been completed to determine probability.
- **Low risk:** Current research indicates realizing the risk is unlikely.
- **Future risk:** The current probability is low. Continued adoption would require a fresh analysis.
- **Medium risk:** It's likely that the risk will affect the business.
- **High risk:** Over time, it's increasingly likely that the business will realize this risk.
- **Declining risk:** The risk is medium to high. Actions in IT or the business are reducing the likelihood of an impact.

Determining tolerance:

The three question sets above should fuel enough data to determine initial tolerances. When risk and probability are low, and risk remediation costs are high, the business is unlikely to invest in remediation. When risk and probability are high, the business is likely to consider an investment, as long as the costs don't exceed the potential risks.

Next steps

This type of conversation can help the business and IT evaluate tolerance more effectively. These conversations

can be used during the creation of MVP policies and during incremental policy reviews.

Define corporate policy

Define corporate policy for cloud governance

3/6/2021 • 3 minutes to read • Edit Online

Once you've analyzed the known risks and related risk tolerances for your organization's cloud transformation journey, your next step is to establish policy that will explicitly address those risks and define the steps needed to remediate them where possible.

How can corporate IT policy become cloud-ready?

In traditional governance and incremental governance, corporate policy creates the working definition of governance. Most IT governance actions seek to implement technology to monitor, enforce, operate, and automate those corporate policies. Cloud governance is built on similar concepts.

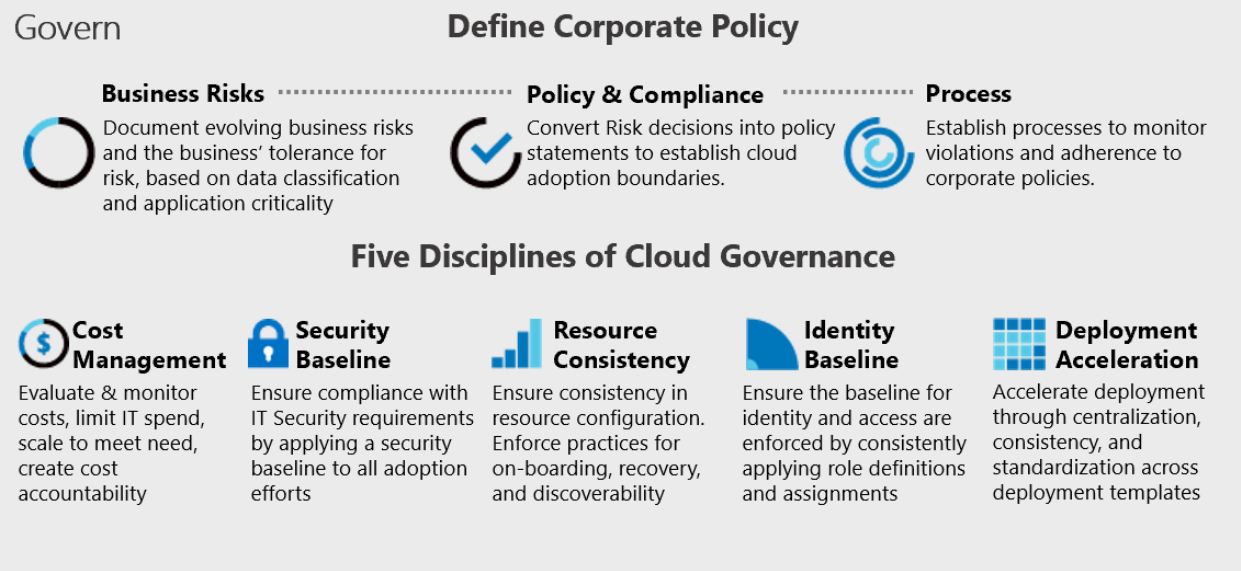


Figure 1: Corporate governance and governance disciplines.

The image above illustrates the relationship between business risk, policy and compliance, and monitoring and enforcement mechanisms that need to interact as part of your governance strategy. The Five Disciplines of Cloud Governance allow you to manage these interactions and realize your strategy.

Cloud governance is the product of an ongoing adoption effort over time, as a true lasting transformation doesn't happen overnight. Attempting to deliver complete cloud governance before addressing key corporate policy changes using a fast aggressive method seldom produces the desired results. Instead we recommend an incremental approach.

What is different about a Cloud Adoption Framework is the purchasing cycle and it can enable authentic transformation. Since there is not a large capital expenditure acquisition requirement, engineers can begin experimentation and adoption sooner. In most corporate cultures, elimination of the capital expense barrier to adoption can lead to tighter feedback loops, organic growth, and incremental execution.

The shift to cloud adoption requires a shift in governance. In many organizations, corporate policy transformation allows for improved governance and higher rates of adherence through incremental policy changes and automated enforcement of those changes, powered by newly defined capabilities that you configure with your cloud service provider.

Review existing policies

As governance is an ongoing process, policy should be regularly reviewed with IT staff and stakeholders to ensure resources hosted in the cloud continue to maintain compliance with overall corporate goals and requirements. Your understanding of new risks and acceptable tolerance can fuel a review of existing policies, in order to determine the required level of governance that is appropriate for your organization.

> **TIP**
>
> If your organization uses vendors or other trusted business partners, one of the biggest business risks to consider may be a lack of adherence to regulatory compliance by these external organizations. This risk often cannot be remediated, and instead may require a strict adherence to requirements by all parties. Make sure you've identified and understand any third-party compliance requirements before beginning a policy review.

Create cloud policy statements

Cloud-based IT policies establish the requirements, standards, and goals that your IT staff and automated systems will need to support. Policy decisions are a primary factor in your cloud architecture design and how you will implement your policy adherence processes.

Individual cloud policy statements are guidelines for addressing specific risks identified during your risk assessment process. While these policies can be integrated into your wider corporate policy documentation, cloud policy statements discussed throughout the Cloud Adoption Framework guidance tends to be a more concise summary of the risks and plans to deal with them. Each definition should include these pieces of information:

- **Business risk:** A summary of the risk this policy will address.
- **Policy statement:** A concise explanation of the policy requirements and goals.
- **Design or technical guidance:** Actionable recommendations, specifications, or other guidance to support and enforce this policy that IT teams and developers can use when designing and building their cloud deployments.

If you need help starting to define your policies, consult the governance disciplines introduced in the governance section overview. The articles for each of these disciplines includes examples of common business risks encountered when moving to the cloud and sample policies used to remediate those risks. For example, see the Cost Management discipline's sample policy definitions.

Incremental governance and integrating with existing policy

Planned additions to your cloud environment should always be vetted for compliance with existing policy, and policy updated to account for any issues not already covered. You should also perform regular cloud policy review to ensure your cloud policy is up-to-date and in-sync with any new corporate policy.

The need to integrate cloud policy with your legacy IT policies depends largely on the maturity of your cloud governance processes and the size of your cloud estate. See the article on incremental governance and the policy MVP for a broader discussion on dealing with policy integration during your cloud transformation.

Next steps

After defining your policies, draft an architecture design guide to provide IT staff and developers with actionable guidance.

Align your governance design guide with corporate policy

Align your cloud governance design guide with corporate policy

3/6/2021 • 2 minutes to read • Edit Online

After you've defined cloud policies based on your identified risks, you'll need to generate actionable guidance that aligns with these policies for your IT staff and developers to refer to. Drafting a cloud governance design guide allows you to specify specific structural, technological, and process choices based on the policy statements you generated for each of the five governance disciplines.

A cloud governance design guide should establish the architecture choices and design patterns for each of the core infrastructure components of cloud deployments that best meet your policy requirements. Alongside these you should provide a high-level explanation of the technology, tools, and processes that will support each of these design decisions.

Although your risk analysis and policy statements may, to some degree, be cloud platform agnostic, your design guide should provide platform-specific implementation details that your IT and dev teams can use when creating and deploying cloud-based workloads. Focus on the architecture, tools, and features of your chosen platform when making design decision and providing guidance.

While cloud design guides should take into account some of the technical details associated with each infrastructure component, they're not meant to be extensive technical documents or specifications. Make sure your guides address your policy statements and clearly state design decisions in a format easy for staff to understand and reference.

Use the actionable governance guides

If you're planning to use the Azure platform for your cloud adoption, the Cloud Adoption Framework provides actionable governance guides illustrating the incremental approach of the Cloud Adoption Framework governance model. These narrative guides cover a range of common adoption scenarios, including the business risks, tolerance requirements, and policy statements that went into creating a governance minimum viable product (MVP). These guides represent a synthesis of real-world customer experience of the cloud adoption process in Azure.

While every cloud adoption has unique goals, priorities, and challenges, these samples should provide a good template for converting your policy into guidance. Pick the closest scenario to your situation as a starting point, and mold it to fit your specific policy needs.

Next steps

With design guidance in place, establish policy adherence processes to ensure policy compliance.

Establish policy adherence processes

Establish policy adherence processes

3/6/2021 • 5 minutes to read • Edit Online

After establishing your cloud policy statements and drafting a design guide, you'll need to create a strategy for ensuring your cloud deployment stays in compliance with your policy requirements. This strategy will need to encompass your cloud governance team's ongoing review and communication processes, establish criteria for when policy violations require action, and defining the requirements for automated monitoring and compliance systems that will detect violations and trigger remediation actions.

See the corporate policy sections of the actionable governance guides for examples of how policy adherence process fit into a cloud governance plan.

Prioritize policy adherence processes

How much investment in developing processes is required to support your policy goals? Depending on the size and maturity of your cloud deployment, the effort required to establish processes that support compliance, and the costs associated with this effort, can vary widely.

For small deployments consisting of development and test resources, policy requirements may be simple and require few dedicated resources to address. On the other hand, a mature mission-critical cloud deployment with high-priority security and performance needs may require a team of staff, extensive internal processes, and custom monitoring tooling to support your policy goals.

As a first step in defining your policy adherence strategy, evaluate how the processes discussed below can support your policy requirements. Determine how much effort is worth investing in these processes, and then use this information to establish realistic budget and staffing plans to meet these needs.

Establish cloud governance team processes

Before defining triggers for policy compliance remediation, you need establish the overall processes that your team will use and how information will be shared and escalated between IT staff and the cloud governance team.

Assign cloud governance team members

Your cloud governance team will provide ongoing guidance on policy compliance and handle policy-related issues that emerge when deploying and operating your cloud assets. When building this team, invite staff members that have expertise in areas covered by your defined policy statements and identified risks.

For initial test deployments, this can be limited to a few system administrators responsible for establishing the basics of governance. As your governance processes mature, review the cloud guidance team's membership regularly to ensure that you can properly address new potential risks and policy requirements. Identify members of your IT and business staff with relevant experience or interest in specific areas of governance and include them in your teams on a permanent or temporary basis as needed.

Reviews and policy iteration

As additional resources and workloads are deployed, the cloud governance team will need to ensure that new workloads or assets comply with policy requirements. Evaluate new requirements from workload development teams to ensure their planned deployments will align with your design guides, and update your policies to support these requirements when appropriate.

Plan to evaluate new potential risks and update policy statements and design guides as needed. Work with IT

staff and workload teams to evaluate new Azure features and services on an ongoing basis. Also schedule regular review cycles each of the five governance disciplines to ensure policy is current and in compliance.

Education

Policy compliance requires IT staff and developers to understand the policy requirements that affect their areas of responsibility. Plan to devote resources to document decisions and requirements, and educate all relevant teams on the design guides that support your policy requirements.

As policy changes, regularly update documentation and training materials, and ensure education efforts communicate updated requirements and guidance to relevant IT staff.

At various stages of your cloud journey, you may find it best to consult with partners and professional training programs to enhance the education of your team, both technically, and procedurally. Additionally, many find that formal certifications are a valuable addition to your education portfolio and should be considered.

Establish escalation paths

If a resource goes out of compliance, who gets notified? If IT staff detect a policy compliance issue, who do they contact? Make sure the escalation process to the cloud governance team is clearly defined. Ensure these communication channels are kept updated to reflect staff and organization changes.

Violation triggers and actions

After defining your cloud governance team and its processes, you need to explicitly define what qualifies as compliance violations that will triggers actions, and what those actions should be.

Define triggers

For each of your policy statements, review requirements to determine what constitutes a policy violation. Generate your triggers using the information you've already established as part of the policy definition process.

- **Risk tolerance:** Create violation triggers based on the metrics and risk indicators you established as part of your risk tolerance analysis.
- **Defined policy requirements:** Policy statements may provide service-level agreement (SLA), business continuity and disaster recovery (BCDR), or performance requirements that should be used as the basis for compliance triggers.

Define actions

Each violation trigger should have a corresponding action. Triggered actions should always notify an appropriate IT staff or cloud governance team member when a violation occurs. This notification can lead to a manual review of the compliance issue or kickoff a predefined remediation process depending on the type and severity of the detected violation.

Some examples of violation triggers and actions:

GOVERNANCE DISCIPLINE	SAMPLE TRIGGER	SAMPLE ACTION
Cost Management	Monthly cloud spending is more than 20% higher than expected.	Notify the billing unit leader who will begin a review of resource usage.
Security Baseline	Detect suspicious user activity.	Notify the IT security team and disable the suspect user account.
Resource Consistency	CPU utilization for a workload is greater than 90%.	Notify the IT operations team and scale out additional resources to handle the load.

Automation of monitoring and compliance

After you've defined your compliance violation triggers and actions, you can start planning how best to use the logging and reporting tools and other features of the cloud platform to help automate your monitoring and policy compliance strategy.

For help choosing the best monitoring pattern for your deployment, see the logging and reporting decision guide.

Next steps

Learn more about regulatory compliance in the cloud.

Regulatory compliance

Implement cloud service controls

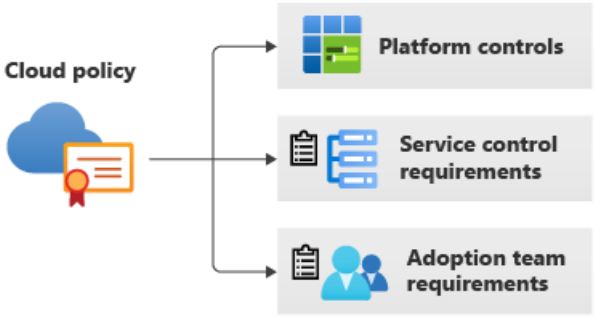
6/8/2021 • 14 minutes to read • Edit Online

Your organization might find it difficult to implement cloud governance at a pace that sustains a high level of innovation, and implements the right level of control for the environment.

The following process outlines a possible governance implementation, allowing for a controlled and measured implementation. This process is non-disruptive, wherever your organization is in their cloud journey.

Layers of governance

Determine the controls and measures to implement as part of your organization's governance process as outlined in Define Corporate Cloud Policy.



Implementing organizational cloud policy stretches across many layers of the implementation process. These layers include platform-wide controls, individual service controls, and application controls. The higher the level of control, the more coverage it gets. The more complex the implementation is, the easier it is to ensure the cloud policy covers all of the cloud adoption team needs.

CONTROL LEVEL	SUITABILITY	CONSIDERATIONS	EXAMPLES
Platform	Broad coverage	Hard to implement service-specific management or remediation. Ensure that the desired outcome is not negated by the application implementation.	Resource tagging or allowed regions
Cloud service	Tailored for individual services	Control must be implemented individually for each service. Ensure that the desired outcome is not negated by the application implementation.	Data encryption or monitoring
Adoption team	Tailored for each application implementation	Control must be implemented for each new application. Hard to standardize implementation. Multiple teams have to implement different controls for the desired outcome.	Application failover logic

Cloud services have the strongest and most complex level of control. They provide technical implementation with fine control of each service, while allowing for single implementations to be used by multiple adoption teams. The process of implementing controls at the cloud service layer requires a level of scale to ensure that all services needed by your organization are covered.

Levels of service governance

This level of implementation can be delivered in several layers. The governance team should assess the level of implementation they want based on the risk profile, compliance requirements, security posture, and other factors.

SERVICE GOVERNANCE	OUTLINE	ADOPTION TEAM RESPONSIBILITIES	GOVERNANCE TEAM RESPONSIBILITIES
None	Services aren't centrally governed.	The cloud adoption teams must build corporate policy into the end solution.	Identify and implement platform-wide controls.
Basic (allowlist and blocklist)	Services are allowed or blocked with no other controls.	The application teams build corporate policy into the end solution. Only services that have been approved by the governance team are used.	Review and approve the use of services.
Service review process	Services are individually reviewed and controls are placed around each service.	The application teams should build with confidence. Only the services that have been through a service review are available.	Complete service governance and implement controls for each service.

While classifying the different approaches lists out the extreme ends of the spectrum, many organizations will choose an approach between these models. For example, most organizations will allow some cloud services with a few controls, some will just be allowed without controls, and access to others will be denied.

Whatever the approach taken to service governance is, the key input is the corporate Cloud policy statements defined by the governance team.

Basic governance

The most basic implementation of service governance is a permitted services list, that lists the permitted or disallowed services in Azure.

In this form, the governance team publishes a list of the allowed and blocked services for the adoption teams.

Creating and reviewing allowed services

To create an allowlist and blocklist of cloud services, review each cloud service against several previously defined criteria, for example:

- Organization risk tolerance
- Corporate policy
- Capabilities of adoption, security, and operational teams

Implementing blocklists

Services that are explicitly denied use in your organization should use Azure Policy to block the use of the service. Azure Policy is a platform level of control and can be implemented by a management group to ensure

control over subordinate subscriptions.

The number of policies used to implement the block service is a balanced decision that must take into account the process for a policy exception.

There are often exceptions for the use of a service. Once the exception has been approved through the organizational process, Azure Policy exemptions can be used to allow the implementation of the service in a targeted area.

If a single policy is implemented with all the denied services listed, then granting an exception for a single service will grant an exception to all denied services. Upfront implementation is simpler, but releases more control with exceptions.

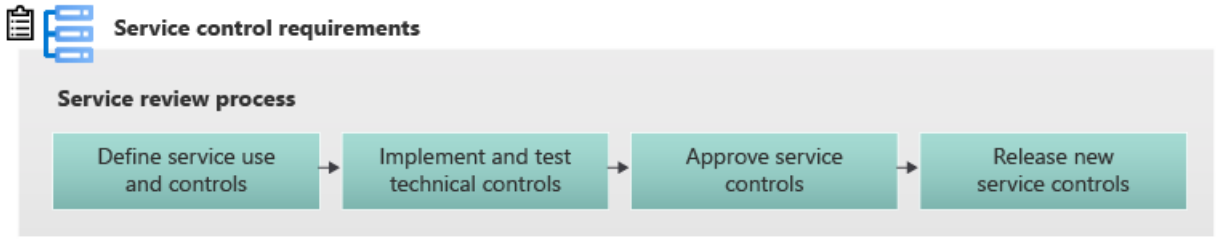
If policies are implemented to block each service individually, then exceptions only release control of that one policy. This implementation allows for more control in the exception process, but creates greater work in the implementation of the initial block policy set.

Your organization should balance the outcome of these two processes when deciding their own approach.

Service review process

Implementing cloud policy at the service level gives greater control, and allows for granular solutions, but requires more effort. For this type of implementation, a strong process is required to get the best result.

This example is a simple four-step process.



Triggers for service review process can include:

- Project need for new service
- Scheduled review of existing service
- Major cloud release of features in service
- Change in imperatives (which triggers all services in need of review)

The need for a service review puts the service in the backlog for the review process.

The schedule for the service review backlog should be published for adoption and operations teams. This schedule allows them to plan for their use of the service or any possible remediation work required because of changes.

The next part of the process contains steps that are iterative for each of the cloud services used. This part is continual for much of the cloud governance team. The core of this process is to take the imperatives that have been defined and interpret them into controls for each service.

Selecting a service for review should be based on several components:

- Internal demand for the service complexity of the implementation
- The review timing of the last review of the service
- High-level risk value of the service

This process should publish a backlog of items and a schedule of new reviews. Outside cloud adoption teams can then understand when services are going to be selected for review and how long they might wait for a new

service. The process allows cloud adoption teams to select services that are already under the controls of the environment.

Define service use and controls

Review the capabilities of an individual cloud service and then determine what service-specific controls and guidance should be applied.

The primary inputs are the technical documentation of the cloud service and the service control requirements.

Through this process, the team reviews the intended use of the service within the organization, and then documents any implementation-specific designs required. The designs might include different controls for different classifications of data, or it could be simple guidance for general use.

The core artifact delivered from this stage is the service guidance. This document outlines the envisioned usage of the service, guiding design principles, and the audited and enforced controls on the service. It acts as a reference for cloud adoption teams when they're choosing to implement a service as part of their design. The documents are necessary for the governance team to communicate their intent to adoption teams at scale.

Implement and test technical controls

The controls defined in the previous section should now be implemented in cloud controls. The technical implementation must provide the right level of enforcement and reporting.

The controls of the service are interpreted into their technical implementation on the service. The controls are identified in their implementation of the Azure environment. For example, requiring encryption on a storage account requires you to set a property on the storage account, which then becomes the technical requirement of that control.

There are many offerings in the market that provide this capability. Azure Policy is native to Azure and is recommended. Azure Policy works to report the current state versus the desired outcome, and to enforce the controls.

Approve service controls

The governance team must now approve the work done in the last two stages.

As this approval stage is with the governance team, the previous two stages can be completed by other teams. This process is outlined in Scaling the review process within an organization.

Release service controls

Releasing changed controls for services that have an existing service guidance is different than releasing new service guidance.

With modification of an existing service guidance, remember that there are adoption teams using the current service guidance in their work. Large changes might break their current projects.

When doing incremental reviews of service guidance, you can look at how a service is being used to determine what effect new controls might create.

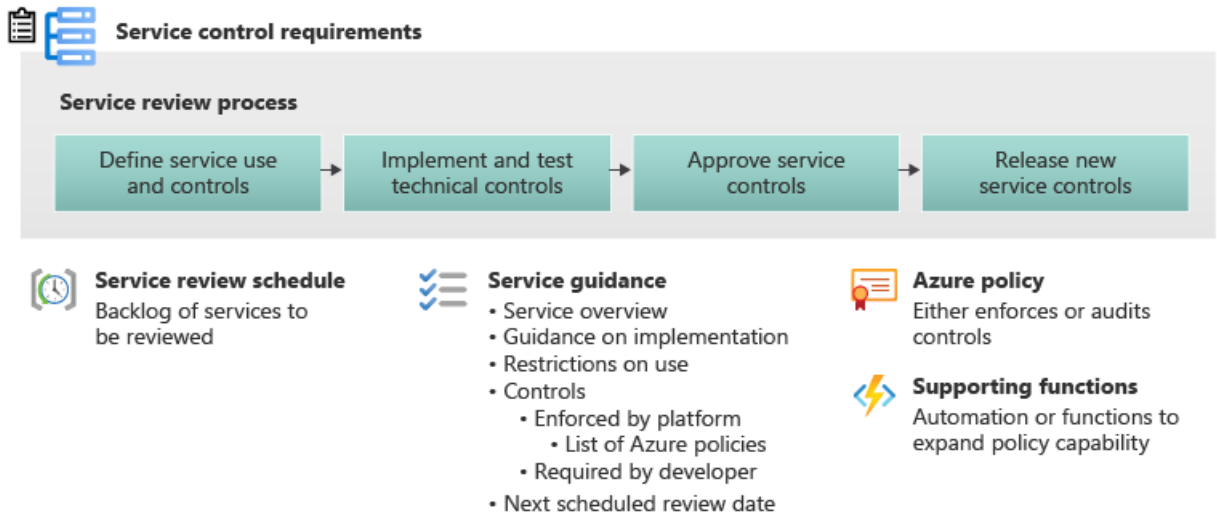
To start with, release new controls in an audit mode. Such a release automatically triggers the policy alerting engine to start notifications to the application owners about any remediation they must implement.

Service governance, release, and version control

The design imperatives must have a version and release cycle and be tied back to the service guidance. The service guidance must have a version and release cycle and be tied back to the policy controls. Finally, the policy controls must have versions and be released in manageable waves.

- All of these steps should happen with release cycles
- Services must be available on a set date
- Policy controls must be advertised on a release cycle

Artifacts generated



There are four key artifacts developed throughout this process.

- The **Service control requirements** are the outline of the key controls that are required across all services. This document is published above the reference documentation for cloud adoption teams. This positioning ensures that they've reviewed and understand the controls that are being built into the systems. This document must be available so that others can help the development of a service review. This process is outlined in Scaling the review process within an organization.
- The **Service review schedule** is a document that outlines when the services are being reviewed, how long the expected wait for a review is, and lets adoption teams know when a new service becomes available. They'll also know when existing services might change and need more implementation of controls.
- **Service guidance**, is a per service document that outlines the key components around a service:
  - When the service should be used
  - Any restrictions on what it should be used for
  - A list of controls that are enforced by the platform
  - A list of controls that the developer should implement

  Cloud adoption teams refer to these documents for services they can pick up and use, and then plan the implementation of their project around the controls and components. The service guidance should also list any Azure policies and supporting functions that have been implemented. This guidance ensures that a reference can be enforced between the policy and service controls.
- **Azure policy and supporting functions**, are the components or technical implementation of the controls that go into the platform. This code is version-controlled and deployed. These code items must act in enforcing the policy and include the code for reporting compliance of the policy.

Starting with a baseline

When starting this process, most organizations will already have an Azure footprint. To ensure the least disruptive implementation of service governance, it's best to baseline off the current implementation and then start service reviews. This method avoids stifling innovation or blocking new projects while the governance team is starting.

1. Create a service guidance for every service that's currently used in your organization with the current service controls.

   For example, if key vaults are used and there are no controls enforced on them, create a service guidance for "a key vault with no controls". If storage accounts are in use and a policy enforces encryption, create

the storage account guidance with the control that encryption will be enforced.

2. Trigger a service review for all services that are currently in use, adding them all to the service review schedule.
3. Start a rollout of the new or changed controls in line with the service review policy, giving product teams time to remediate their services.

In this stage, as the baseline, all services currently in production use are approved with their current policy controls.

From here, the process is ready to start working with the service guidance as if it were going through a normal review.

For example, if your organization has deployed virtual machines (VM) to production today, and the only policy governing that is tagging, then the MVP for the VM service guidance only documents the control around tagging.

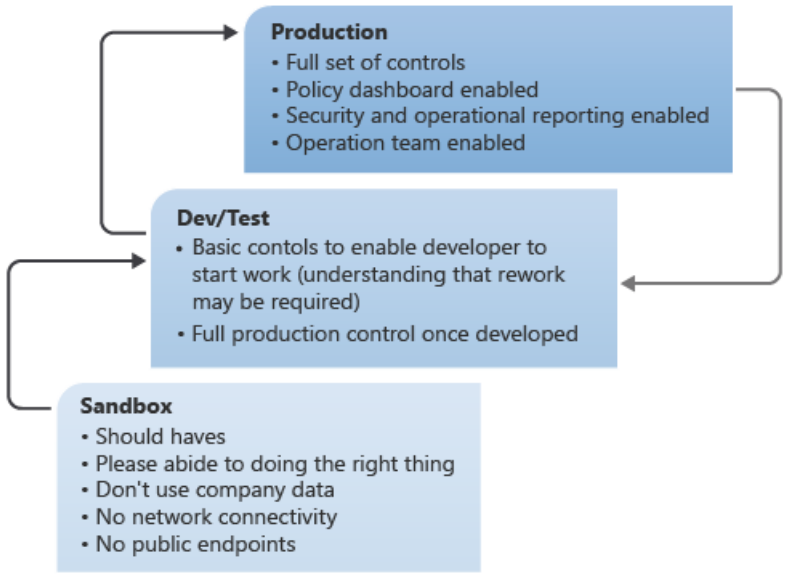
If more controls need to be enforced for virtual machines, a new version of the service guidance is written. The new controls are implemented and go through the release cycle for the new service guidance version.

Differences from the initial review

A main component of the service review process is to prevent services that don't have a current published service guidance from being deployed into the environment. It prevents the deployment of services that haven't had a risk assessment performed against the current cloud policy.

A staged approach for allowing service use in different environments gives adoption teams earlier access to controls while reducing the exposure of services without controls.

There are several stages that ensure adoption teams aren't blocked during the creation of new service guidance. The process accelerates their ability to start using services. The first stage, "Sandbox", is for the adoption team to nominate the new service that they need for a service review.



Here, organizations can decide to allow adoption teams to start using the service in a disconnected sandbox environment. The sandbox lets the adoption team start working with the service and educate themselves by prototyping solutions. The sandbox environment doesn't have any policy controls for the service. As no review has been done, generic compensating controls should be enforced on the environment. These compensating controls might include:

- Agreement from the adoption team to try to work within the cloud policy
- Agreement to not store any company data in the sandbox environment

- Limited network connectivity to the corporate environment
- Limited connectivity to outside services

Once a draft version of the service review has been done, the service can be provisionally approved for use in a development and test environment. At this point, the code for the controls might not be complete, but basic guidance about how the service is deployed has been written. The developers can start to work in the dev/test environment with the understanding that the service might require reworking later. For example, if new controls are added as part of the final implementation of the production controls, some rework might be needed.

For the release of service controls, and to authorize application teams to deploy the service into the production environment, the following tasks must be complete:

- Service guidance is completed
- Policy controls are code complete and tested
- Approval of service controls is complete
- Policy controls are released to production

Now the adoption team can deploy the final production controls with confidence.

After deployment, the production controls are then migrated back to the development and test environment. This migration might cause rework for the development environment but ensures that the full controls are in place.

Scaling the review process within an organization

The combination of the scale of the cloud services and the depth of the service review process often creates a large work effort. To avoid this backlog of work being the sole responsibility of the governance team, it's possible to scale through the adoption teams to complete the work. The governance team retains control and approval in this process. **Final approval of content stays with the governance team**.

Changes often happen with the cloud, or your organization, that make it difficult to keep controls current. Enabling people outside the cloud governance team to implement parts of the work helps.

The implementation of a clear set of service control requirements allows other teams to work on the first two stages of the process for a service, contributing a large amount of content.

The cloud has many services. Although the cloud governance team can prepare a backlog for all of these services to go through a service review, there's no way that they can keep pace with the demands of a large organization. Services in the cloud change at a continuing rate, placing more demand on the need for a service review of previously defined services. Projects might require new types of services quickly that they were unable to predict the demand for. Their immediate demand shouldn't become a problem for the cloud governance team. That demand must be addressed by the cloud adoption team that needs the service.

If the service control requirements are written clearly, teams outside the governance team can complete the first two steps of the service review process.

With the approval and release steps still controlled by the governance team, organizations can scale the service review process while keeping centralized control.

For example, a cloud adoption team might need a new service that hasn't been through a review. To get the service review completed more quickly, they can use the documentation in the imperatives to identify and produce the controls. This method moves them all the way through the technical implementation and policy code for the cloud governance team. The cloud adoption team then gets quicker access to the service and can be interpreted as a "first user pays" type scenario for the implementation of new services.

Next steps

Learn more about regulatory compliance in the cloud.

Regulatory compliance

Introduction to regulatory compliance

3/6/2021 • 3 minutes to read • Edit Online

This article provides an introduction to regulatory compliance, and therefore is not intended for implementing a compliance strategy. More detailed information about Azure compliance offerings is available at the Microsoft Trust Center. Moreover, all downloadable documentation is available to certain Azure customers from the Microsoft Service Trust Portal.

Regulatory compliance refers to the discipline and process of ensuring that a company follows the laws enforced by governing bodies in their geography or rules required by voluntarily adopted industry standards. For IT regulatory compliance, people and processes monitor corporate systems to detect and prevent violations of policies and procedures established by these governing laws, regulations, and standards. This in turn applies to a wide array of monitoring and enforcement processes. Depending on the industry and geography, these processes can become lengthy and complex.

Compliance is challenging for multinational organizations, especially in heavily regulated industries like healthcare and financial services. Standards and regulations abound, and in certain cases may change frequently, making it difficult for businesses to keep up with changing international electronic data handling laws.

As with security controls, organizations should understand the division of responsibilities regarding regulatory compliance in the cloud. Cloud providers strive to ensure that their platforms and services are compliant. Organizations also need to confirm that their applications, the infrastructure those applications depend on, and services supplied by third parties are also certified as compliant.

The following are descriptions of compliance regulations in various industries and geographies:

HIPAA

A healthcare application that processes protected health information (PHI) is subject to both the privacy rule and the security rule encompassed within the Health Insurance Portability and Accountability Act (HIPAA). At a minimum, HIPAA could likely require that a healthcare business must receive written assurances from the cloud provider that it will safeguard any PHI received or created.

PCI

The Payment Card Industry Data Security Standard (PCI DSS) is a proprietary information security standard for organizations that handle branded credit cards from the major card payment systems, including Visa, Mastercard, American Express, Discover, and JCB. The PCI standard is mandated by the card brands and administered by the Payment Card Industry Security Standards Council. The standard was created to increase controls around cardholder data to reduce credit-card fraud. Validation of compliance is performed annually, either by an external qualified security assessor (QSA) or by a firm-specific internal security assessor (ISA) who creates a report on compliance (ROC) for organizations handling large volumes of transactions, or by a self-assessment questionnaire (SAQ) for companies.

Personal data

Personal data is information that could be used to identify a consumer, employee, partner, or any other living or legal entity. Many emerging laws, particularly those dealing with privacy and personal data, require that businesses comply and report on compliance and any breaches that occur.

GDPR

One of the most important developments in this area is the General Data Protection Regulation (GDPR), designed to strengthen data protection for individuals within the European Union. GDPR requires that data about individuals (such as "a name, a home address, a photo, an email address, bank details, posts on social networking websites, medical information, or a computer's IP address") be maintained on servers within the EU and not transferred out of it. It also requires that companies notify individuals of any data breaches, and mandates that companies have a data protection officer (DPO). Other countries have, or are developing, similar types of regulations.

Compliant foundation in Azure

To help customers meet their own compliance obligations across regulated industries and markets worldwide, Azure maintains the largest compliance portfolio in the industry, in both breadth (total number of offerings) and depth (number of customer-facing services in assessment scope). Azure compliance offerings are grouped into four segments:

- Global
- US government
- Industry
- Regional

Azure compliance offerings are based on various types of assurances, including formal certifications, attestations, validations, authorizations, and assessments produced by independent third-party auditing firms, as well as contractual amendments, self-assessments, and customer guidance documents produced by Microsoft. Each offering description in this document provides an up-to-date scope statement indicating which Azure customer-facing services are in scope for the assessment, along with links to downloadable resources to assist customers with their own compliance obligations.

The Microsoft Trust Center provides more detailed information about Azure compliance offerings. Additionally, all downloadable documentation is available to certain Azure customers from the Microsoft Service Trust Portal in the following sections:

- **Audit reports:** Includes sections for FedRAMP, GRC assessment, ISO, PCI DSS, and SOC reports.
- **Data protection resources:** Includes compliance guides, FAQ and white papers, and pen test and security assessment sections.

Next steps

Learn more about cloud security readiness.

Cloud security readiness

CISO cloud readiness guide

3/6/2021 • 3 minutes to read • Edit Online

Microsoft guidance like the Cloud Adoption Framework is not positioned to determine or guide the unique security constraints of the thousands of enterprises supported by this documentation. When moving to the cloud, the role of the chief information security officer or chief information security office (CISO) isn't supplanted by cloud technologies. Quite the contrary, the CISO and the office of the CISO, become more engrained and integrated. This guide assumes the reader is familiar with CISO processes and is seeking to modernize those processes to enable cloud transformation.

Cloud adoption enables services that weren't often considered in traditional IT environments. Self-service or automated deployments are commonly executed by application development or other IT teams not traditionally aligned to production deployment. In some organizations, business constituents similarly have self-service capabilities. This can trigger new security requirements that weren't needed in the on-premises world. Centralized security is more challenging, security often becomes a shared responsibility across the business and IT culture. This article can help a CISO prepare for that approach and engage in incremental governance.

How can a CISO prepare for the cloud?

Like most policies, security and governance policies within an organization tend to grow organically. When security incidents happen, they shape policy to inform users and reduce the likelihood of repeat occurrences. While natural, this approach creates policy bloat and technical dependencies. Cloud transformation journeys create a unique opportunity to modernize and reset policies. While preparing for any transformation journey, the CISO can create immediate and measurable value by serving as the primary stakeholder in a policy review.

In such a review, the role of the CISO is to create a safe balance between the constraints of existing policy/compliance and the improved security posture of cloud providers. Measuring this progress can take many forms, often it's measured in the number of security policies that can be safely offloaded to the cloud provider.

Transferring security risks: As services are moved into infrastructure as a service (IaaS) hosting models, the business assumes less direct risk regarding hardware provisioning. The risk isn't removed, instead it's transferred to the cloud vendor. Should a cloud vendor's approach to hardware provisioning provide the same level of risk mitigation, in a secure repeatable process, the risk of hardware provisioning execution is removed from corporate IT's area of responsibility and transferred to the cloud provider. This reduces the overall security risk that corporate IT is responsible for managing, although the risk itself should still be tracked and reviewed periodically.

As solutions move further "up stack" to incorporate platform as a service (PaaS) or software as a service (SaaS) models, additional risks can be avoided or transferred. When risk is safely moved to a cloud provider, the cost of executing, monitoring, and enforcing security policies or other compliance policies can be safely reduced as well.

Growth mindset: Change can be scary to both the business and technical implementors. When the CISO leads a growth mindset shift in an organization, we've found that those natural fears are replaced with an increased interest in safety and policy compliance. Approaching a policy review, a transformation journey, or simple implementation reviews with a growth mindset, allows the team to move quickly but not at the cost of a fair and manageable risk profile.

Resources for the chief information security officer

Knowledge about the cloud is fundamental to approaching a policy review with a growth mindset. The following

resources can help the CISO better understand the security posture of Microsoft's Azure platform.

Security platform resources:

- Security development lifecycle, internal audits
- Mandatory security training, background checks
- Penetration testing, intrusion detection, DDoS, audits, and logging
- State-of-the-art datacenter, physical security, secure network

Privacy and controls:

- Manage your data all the time
- Control on data location
- Provide data access on your terms
- Responding to law enforcement
- Stringent privacy standards

Compliance:

- Microsoft Trust Center
- Common controls hub
- Cloud Services Due Diligence Checklist
- Regional and country compliance

Transparency:

- How Microsoft secures customer data in Azure services
- How Microsoft manages data location in Azure services
- Who in Microsoft can access your data on what terms
- Review certification for Azure services, transparency hub

Next steps

The first step to taking action in any governance strategy is a policy review. Policy and compliance could be a useful guide during your policy review.

Prepare for a policy review

Conduct a cloud policy review

3/13/2021 • 3 minutes to read • Edit Online

A cloud policy review is the first step toward governance maturity in the cloud. The objective of this process is to modernize existing corporate IT policies. When completed, the updated policies provide an equivalent level of risk management for cloud-based resources. This article explains the cloud policy review process and its importance.

Why perform a cloud policy review?

Most businesses manage IT through the execution of processes that align with governing policies. In small businesses, these policies may anecdotal and processes loosely defined. As businesses grow into large enterprises, policies and processes tend to be more clearly documented and consistently executed.

As companies mature corporate IT policies, dependencies on past technical decisions have a tendency to seep into governing policies. For instance, it's common to see disaster recovery processes include policy that mandates offsite tape backups. This inclusion assumes a dependency on one type of technology (tape backups), that may no longer be the most relevant solution.

Cloud transformations create a natural inflection point to reconsider the legacy policy decisions of the past. Technical capabilities and default processes change considerably in the cloud, as do the inherit risks. Using the prior example, the tape backup policy stemmed from the risk of a single point of failure by keeping data in one location and the business need to minimize the risk profile by mitigating this risk. In a cloud deployment, there are several options that deliver the same risk mitigation, with much lower recovery time objectives (RTO). For example:

- A cloud-native solution could enable geo-replication of the Azure SQL Database.
- A hybrid solution could use Azure Site Recovery to replicate an IaaS workload to Azure.

When executing a cloud transformation, policies often govern many of the tools, services, and processes available to the cloud adoption teams. If those policies are based on legacy technologies, they may hinder the team's efforts to drive change. In the worst case, important policies are entirely ignored by the migration team to enable workarounds. Neither is an acceptable outcome.

The cloud policy review process

Cloud policy reviews align existing IT governance and IT security policies with the Five Disciplines of Cloud Governance:

- Cost Management discipline
- Security Baseline discipline
- Identity Baseline discipline
- Resource Consistency discipline
- Deployment Acceleration discipline.

For each of these disciplines, the review process follows these steps:

1. Review existing on-premises policies related to the specific discipline, looking for two key data points: legacy dependencies and identified business risks.
2. Evaluate each business risk by asking a simple question: does the business risk still exist in a cloud model?
3. If the risk still exists, rewrite the policy by documenting the necessary business mitigation, not the technical

solution.

4. Review the updated policy with the cloud adoption teams to understand potential technical solutions to the required mitigation.

Example of a policy review for a legacy policy

To provide an example of the process, let's again use the tape backup policy in the prior section:

- A corporate policy mandates offsite tape backups for all production systems. In this policy, you can see two data points of interest:
  - Legacy dependency on a tape backup solution.
  - An assumed business risk associated with the storage of backups in the same physical location as the production equipment.
- Does the risk still exist? Yes. Even in the cloud, a dependence on a single facility does create some risk. There is a lower probability of this risk affecting the business than was present in the on-premises solution, but the risk still exists.
- Rewrite of the policy. In the case of a datacenter-wide disaster, there must exist a means of restoring production systems within 24 hours of the outage in a different datacenter and different geographic location.
  - It's also important to consider that the timeline specified in the above requirement may have been set by technical constraints that are no longer present in the cloud. Make sure to understand the technical constraints and capabilities of the cloud before simply applying a legacy RTO/RPO.
- Review with the cloud adoption teams. Depending on the solution being implemented, there are multiple means of adhering to this Resource Consistency policy.

Next steps

Learn more about including data classification in your cloud governance strategy.

Data classification

What is data classification?

3/6/2021 • 2 minutes to read • Edit Online

Data classification allows you to determine and assign value to your organization's data and provides a common starting point for governance. The data classification process categorizes data by sensitivity and business impact in order to identify risks. When data is classified, you can manage it in ways that protect sensitive or important data from theft or loss.

Understand data risks, then manage them

Before any risk can be managed, it must be understood. In the case of data breach liability, that understanding starts with data classification. Data classification is the process of associating a metadata characteristic to every asset in a digital estate, which identifies the type of data associated with that asset.

Any asset identified as a potential candidate for migration or deployment to the cloud should have documented metadata to record the data classification, business criticality, and billing responsibility. These three points of classification can go a long way to understanding and mitigating risks.

Classifications Microsoft uses

The following is a list of classifications Microsoft uses. Depending on your industry or existing security requirements, data classification standards might already exist within your organization. If no standard exists, you might want to use this sample classification to better understand your own digital estate and risk profile.

- **Non-business:** Data from your personal life that doesn't belong to Microsoft.
- **Public:** Business data that is freely available and approved for public consumption.
- **General:** Business data that isn't meant for a public audience.
- **Confidential:** Business data that can cause harm to Microsoft if overshared.
- **Highly confidential:** Business data that would cause extensive harm to Microsoft if overshared.

Tagging data classification in Azure

Resource tags are a good approach for metadata storage, and you can use these tags to apply data classification information to deployed resources. Although tagging cloud assets by classification isn't a replacement for a formal data classification process, it provides a valuable tool for managing resources and applying policy. Azure Information Protection is an excellent solution to help you classify data itself, regardless of where it resides (on-premises, in Azure, or somewhere else). Consider it as part of an overall classification strategy.

Take action

Take action by defining and tagging assets with a defined data classification.

- Choose one of the actionable governance guides for examples of applying tags across your portfolio.
- Review the recommended naming and tagging conventions to define a more comprehensive tagging standard.
- For additional information on resource tagging in Azure, see Use tags to organize your Azure resources and management hierarchy.

Next steps

Learn more about the Five Disciplines of Cloud Governance.

Five Disciplines of Cloud Governance

The Five Disciplines of Cloud Governance

3/6/2021 • 2 minutes to read • Edit Online

Any change to business processes or technology platforms introduces risk. Cloud governance teams, whose members are sometimes known as cloud custodians, are tasked with mitigating these risks and ensuring minimal interruption to adoption or innovation efforts.

The Cloud Adoption Framework governance model guides these decisions, irrespective of the chosen cloud platform, by focusing on development of corporate policy and the Five Disciplines of Cloud Governance. Actionable design guides demonstrate this model using Azure services. Learn about the disciplines of the Cloud Adoption Framework governance model below.



Figure 1: Visual of corporate policy and the Five Disciplines of Cloud Governance.

Disciplines of Cloud Governance

With any cloud platform, there are common governance disciplines that help inform policies and align toolchains. These disciplines guide decisions about the proper level of automation and enforcement of corporate policy across cloud platforms.

	Cost Management: Cost is a primary concern for cloud users. Develop policies for cost control for all cloud platforms.
	Security Baseline: Security is a complex subject, unique to each company. Once security requirements are established, cloud governance policies and enforcement apply those requirements across network, data, and asset configurations.
	Identity Baseline: Inconsistencies in the application of identity requirements can increase the risk of breach. The Identity Baseline discipline focuses ensuring that identity is consistently applied across cloud adoption efforts.
	Resource Consistency: Cloud operations depend on consistent resource configuration. Through governance tooling, resources can be configured consistently to manage risks related to onboarding, drift, discoverability, and recovery.

	Deployment Acceleration: Centralization, standardization, and consistency in approaches to deployment and configuration improve governance practices. When provided through cloud-based governance tooling, they create a cloud factor that can accelerate deployment activities.

Cost Management discipline overview

3/6/2021 • 2 minutes to read • Edit Online

The Cost Management discipline is one of the Five Disciplines of Cloud Governance within the Cloud Adoption Framework governance model. For many customers, governing their costs is a major concern when adopting cloud technologies. Balancing performance demands, adoption pacing, and cloud services costs can be challenging. This is especially relevant during major business transformations that implement cloud technologies. This section outlines the approach to developing a Cost Management discipline as part of a cloud governance strategy.

> **NOTE**
>
> Cost Management discipline does not replace the existing business teams, accounting practices, and procedures that are involved in your organization's financial management of IT-related costs. The primary purpose of this discipline is to identify potential cloud-related risks related to IT spending, and provide risk-mitigation guidance to the business and IT teams responsible for deploying and managing cloud resources.

The primary audience for this guidance is your organization's cloud architects and other members of your cloud governance team. The decisions, policies, and processes that emerge from this discipline should involve engagement and discussions with relevant members of your business and IT teams, especially those leaders responsible for owning, managing, and paying for cloud-based workloads.

Policy statements

Actionable policy statements and the resulting architecture requirements serve as the foundation of a Cost Management discipline. Use sample policy statements as a starting point for defining your Cost Management policies.

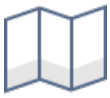
Caution

The sample policies come from common customer experiences. To better align these policies to specific cloud governance needs, execute the following steps to create policy statements that meet your unique business needs.

Develop governance policy statements

The following steps help you define governance policies to control costs in your environment.

	Cost Management discipline template: Download the template for documenting a Cost Management discipline.
	Business risks: Understand the motives and risks commonly associated with the Cost Management discipline.

	Indicators and metrics: Indicators to understand whether it's the right time to invest in the Cost Management discipline.
	Policy adherence processes: Suggested processes for supporting policy compliance in the Cost Management discipline.
	Maturity: Align cloud management maturity with phases of cloud adoption.
	Toolchain: Azure services that can be implemented to support the Cost Management discipline.

Next steps

Get started by evaluating business risks in a specific environment.

Understand business risks

Cost Management discipline template

3/6/2021 • 2 minutes to read • Edit Online

The first step to implementing change is communicating the desired change. The same is true when changing governance practices. The following template serves as a starting point for documenting and communicating policy statements that govern cost management issues in the cloud.

As your discussions progress, use this template's structure as a model for capturing the business risks, risk tolerances, compliance processes, and tooling needed to define your organization's Cost Management policy statements.

> **IMPORTANT**
> This template is a limited sample. Before updating this template to reflect your requirements, you should review the subsequent steps for defining an effective Cost Management discipline within your cloud governance strategy.

Download the Cost Management discipline template

Next steps

Solid governance practices start with an understanding of business risk. Review the article on business risks and begin to document the business risks that align with your current cloud adoption plan.

Understand business risks

Motivations and business risks in the Cost Management discipline

3/6/2021 • 2 minutes to read • Edit Online

This article discusses the reasons that customers typically adopt a Cost Management discipline within a cloud governance strategy. It also provides a few examples of business risks that drive policy statements.

Relevance

In terms of cost governance, cloud adoption creates a paradigm shift. Managing costs in a traditional on-premises world is based on refresh cycles, datacenter acquisitions, host renewals, and recurring maintenance issues. You can forecast, plan, and refine these costs to align with annual capital expenditure budgets.

For cloud solutions, many businesses tend to take a more reactive approach to cost management. In many cases, businesses will prepurchase, or commit to use, a set amount of cloud services. This model assumes that maximizing discounts, based on how much the business plans on spending with a specific cloud vendor, creates the perception of a proactive, planned cost cycle. That perception will only become reality if the business also implements a mature Cost Management discipline.

The cloud offers self-service capabilities that were previously unheard of in traditional on-premises datacenters. These new capabilities empower businesses to be more agile, less restrictive, and more open to adopt new technologies. The downside of self-service is that end users can unknowingly exceed allocated budgets. Conversely, the same users can experience a change in plans and unexpectedly not use the amount of cloud services forecasted. The potential of shift in either direction justifies investment in a Cost Management discipline within the governance team.

Business risk

The Cost Management discipline attempts to address core business risks related to expenses incurred when hosting cloud-based workloads. Work with your business to identify these risks and monitor each of them for relevance as you plan for and implement your cloud deployments.

Risks will differ between organization, but the following serve as common cost-related risks that you can use as a starting point for discussions within your cloud governance team:

- **Budget control:** Not controlling budget can lead to excessive spending with a cloud vendor.
- **Utilization loss:** Prepurchases or precommitments that go unused can result in lost investments.
- **Spending anomalies:** Unexpected spikes in either direction can be indicators of improper usage.
- **Overprovisioned assets:** When assets are deployed in a configuration that exceed the needs of an application or virtual machine (VM), they can create waste.

Next steps

Use the Cost Management policy template to document business risks that are likely to be introduced by the current cloud adoption plan.

After you've gained an understanding of realistic business risks, the next step is to document the business's tolerance for risk and the indicators and key metrics to monitor that tolerance.

Understand indicators, metrics, and risk tolerance

Risk tolerance metrics and indicators in the Cost Management discipline

3/6/2021 • 3 minutes to read • Edit Online

Learn to quantify business risk tolerance associated with the Cost Management discipline. Defining metrics and indicators helps to create a business case for investing in the maturity of this discipline.

Metrics

Cost management generally focuses on metrics related to costs. As part of your risk analysis, you'll want to gather data related to your current and planned spending on cloud-based workloads to determine how much risk you face, and how important investment in your Cost Management discipline is for your planned cloud deployments.

The following are examples of useful metrics that you should gather to help evaluate risk tolerance within the Cost Management discipline:

- **Annual spending:** The total annual cost for services provided by a cloud provider.
- **Monthly spending:** The total monthly cost for services provided by a cloud provider.
- **Forecasted versus actual ratio:** The ratio comparing forecasted and actual spending (monthly or annual).
- **Pace of adoption (month-over-month) ratio:** The percentage of the delta in cloud costs from month to month.
- **Accumulated cost:** Total accrued daily spending, starting from the beginning of the month.
- **Spending trends:** Spending trend against the budget.

Risk tolerance indicators

During early small-scale deployments, such as dev/test or experimental first workloads, cost management is likely to be of relatively low risk. As more assets are deployed, the risk grows and the business's tolerance for risk is likely to decline. Additionally, as more cloud adoption teams are given the ability to configure or deploy assets to the cloud, the risk grows and tolerance decreases. Conversely, developing a Cost Management discipline will take people from the cloud adoption phase to deploying more innovative technologies.

In the early stages of cloud adoption, you will work with your business to determine a risk tolerance baseline. Once you have a baseline, you will need to determine the criteria that would trigger an investment in the Cost Management discipline. These criteria will likely be different for every organization.

Once you have identified business risks, you will work with your business to identify benchmarks that you can use to identify triggers that could potentially increase those risks. The following are a few examples of how metrics, such as those mentioned above, can be compared against your risk baseline tolerance to indicate your business's need to further invest in cost management.

- **Commitment-driven (most common):** A company that is committed to spending *$x,000,000* this year on a cloud vendor. They need a Cost Management discipline to ensure that the business doesn't exceed its spending targets by more than 20%, and that they will use at least 90% of their commitment.
- **Percentage trigger:** A company with cloud spending that is stable for their production systems. If that changes by more than *x%*, then a Cost Management discipline is a wise investment.
- **Overprovisioned trigger:** A company who believes their deployed solutions are overprovisioned. Cost management is a priority investment until they demonstrate proper alignment of provisioning and asset

utilization.

- **Monthly spending trigger:** A company that spends over *$x,000* per month is considered a sizable cost. If spending exceeds that amount in a given month, they will need to invest in cost management.
- **Annual spending trigger:** A company with an IT R&D budget that allows for spending *$x,000* per year on cloud experimentation. They may run production workloads in the cloud, but they're still considered experimental solutions if the budget doesn't exceed that amount. If the budget is exceeded, they will need to treat the budget like a production investment and manage spending closely.
- **Operating expense-adverse (uncommon):** As a company, they're averse to operating expenses and will need cost management controls in place before deploying a dev/test workload.

Next steps

Use the Cost Management discipline template to document metrics and tolerance indicators that align to the current cloud adoption plan.

Review sample Cost Management policies as a starting point to develop your own policies to address specific business risks aligned with your cloud adoption plans.

Review sample policies

Cost Management sample policy statements

3/6/2021 • 3 minutes to read • Edit Online

Individual cloud policy statements are guidelines for addressing specific risks identified during your risk assessment process. These statements should provide a concise summary of risks and plans to deal with them. Each statement definition should include these pieces of information:

- **Business risk:** A summary of the risk this policy will address.
- **Policy statement:** A clear summary explanation of the policy requirements.
- **Design options:** Actionable recommendations, specifications, or other guidance that IT teams and developers can use when implementing the policy.

The following sample policy statements address common cost-related business risks. These statements are examples you can reference when drafting policy statements to address your organization's needs. These examples are not meant to be prescriptive, and there are potentially several policy options for dealing with each identified risk. Work closely with business and IT teams to identify the best policies for your unique set of risks.

Future-proofing

Business risk: Current criteria that don't warrant an investment in a Cost Management discipline from the governance team, but you anticipate such an investment in the future.

Policy statement: You should associate all assets deployed to the cloud with a billing unit and application or workload. This policy will ensure that your Cost Management discipline is effective.

Design options: For information on establishing a future-proof foundation, see the discussions related to creating a governance MVP in the actionable design guides included as part of the Cloud Adoption Framework guidance.

Budget overruns

Business risk: Self-service deployment creates a risk of overspending.

Policy statement: Any cloud deployment must be allocated to a billing unit with approved budget and a mechanism for budgetary limits.

Design options: In Azure, budget can be controlled with Azure Cost Management + Billing.

Underutilization

Business risk: The company has prepaid for cloud services or has made an annual commitment to spend a specific amount. There is a risk that the agreed-on amount won't be used, resulting in a lost investment.

Policy statement: Each billing unit with an allocated cloud budget will meet annually to set budgets, quarterly to adjust budgets, and monthly to allocate time for reviewing planned versus actual spending. Discuss any deviations greater than 20% with the billing unit leader monthly. For tracking purposes, assign all assets to a billing unit.

Design options:

- In Azure, planned versus actual spending can be managed via Azure Cost Management + Billing.
- There are several options for grouping resources by billing unit. In Azure, a resource consistency model should be chosen in conjunction with the governance team and applied to all assets.

Overprovisioned assets

Business risk: In traditional on-premises datacenters, it's common practice to deploy assets with extra capacity planning for growth in the distant future. The cloud can scale more quickly than traditional equipment. Assets in the cloud are also priced based on the technical capacity. There is a risk of the old on-premises practice artificially inflating cloud spending.

Policy statement: Any asset deployed to the cloud must be enrolled in a program that can monitor utilization and report any capacity in excess of 50% of utilization. Any asset deployed to the cloud must be grouped or tagged in a logical manner, so governance team members can engage the workload owner regarding any optimization of overprovisioned assets.

Design options:

- In Azure, Azure Advisor can provide optimization recommendations.
- There are several options for grouping resources by billing unit. In Azure, a resource consistency model should be chosen in conjunction with the governance team and applied to all assets.

Overoptimization

Business risk: Effective cost management creates new risks. Optimization of spending is inverse to system performance. When reducing costs, there is a risk of overtightening spending and producing poor user experiences.

Policy statement: Any asset that directly affects customer experiences must be identified through grouping or tagging. Before optimizing any asset that affects customer experience, the cloud governance team must adjust optimization based on at least 90 days of utilization trends. Document any seasonal or event-driven bursts considered when optimizing assets.

Design options:

- In Azure, Azure Monitor's insights features can help with analysis of system utilization.
- There are several options for grouping and tagging resources based on roles. In Azure, you should choose a resource consistency model in conjunction with the governance team and apply this to all assets.

Next steps

Use the samples mentioned in this article as a starting point to develop policies that address specific business risks that align with your cloud adoption plans.

To begin developing your own custom Cost Management policy statements, download the Cost Management policy template.

To accelerate adoption of this discipline, choose the actionable governance guide that most closely aligns with your environment. Then modify the design to incorporate your specific corporate policy decisions.

Building on risks and tolerance, establish a process for governing and communicating Cost Management policy adherence.

Establish policy compliance processes

Cost Management policy compliance processes

3/6/2021 • 3 minutes to read • Edit Online

This article discusses an approach to creating processes that support an effective Cost Management discipline. Effective governance of cloud costs starts with recurring manual processes designed to support policy compliance. This requires regular involvement of the cloud governance team and interested business stakeholders to review and update policy and ensure policy compliance. In addition, many ongoing monitoring and enforcement processes can be automated or supplemented with tooling to reduce the overhead of governance and allow for faster response to policy deviation.

Planning, review, and reporting processes

The best Cost Management tools in the cloud are only as good as the processes and policies that they support. The following is a set of example processes commonly involved in the Cost Management discipline. Use these examples as a starting point when planning the processes that will allow you to continue to update cost policy based on business change and feedback from the business teams, subject to cost governance guidance.

Initial risk assessment and planning: As part of your initial adoption of the Cost Management discipline, identify your core business risks and tolerances related to cloud costs. Use this information to discuss budget and cost-related risks with members of your business teams and develop a baseline set of policies for mitigating these risks to establish your initial governance strategy.

Deployment planning: Before deploying any asset, establish a forecasted budget based on expected cloud allocation. Ensure that ownership and accounting information for the deployment is documented.

Annual planning: On an annual basis, perform a roll-up analysis on all deployed and to-be-deployed assets. Align budgets by business units, departments, teams, and other appropriate divisions to empower self-service adoption. Ensure that the leader of each billing unit is aware of the budget and how to track spending.

This is the time to make a precommitment or prepurchase to maximize discounting. It's wise to align annual budgeting with the cloud vendor's fiscal year to further capitalize on year-end discount options.

Quarterly planning: On a quarterly basis, review budgets with each billing unit leader to align forecast and actual spending. If there are changes to the plan or unexpected spending patterns, align and reallocate the budget.

This quarterly planning process is also a good time to evaluate the current membership of your cloud governance team for knowledge gaps related to current or future business plans. Invite relevant staff and workload owners to participate in reviews and planning as either temporary advisors or permanent members of your team.

Education and training: On a bimonthly basis, offer training sessions to make sure business and IT staff are up-to-date on the latest Cost Management policy requirements. As part of this process review and update any documentation, guidance, or other training assets to ensure they're in sync with the latest corporate policy statements.

Monthly reporting: On a monthly basis, report actual spending against forecast. Notify billing leaders of any unexpected deviations.

These basic processes will help align spending and establish a foundation for the Cost Management discipline.

Processes for ongoing monitoring

A successful Cost Management strategy depends on visibility into the past, current, and planned future cloud-related spending. Without the ability to analyze the relevant metrics and data of your existing costs, you cannot identify changes in your risks or detect violations of your risk tolerances. The ongoing governance processes discussed above require quality data to ensure policy can be modified to better protect your infrastructure against changing business requirements and cloud usage.

Ensure that your IT teams have implemented automated systems for monitoring your cloud spending and usage for unplanned deviations from expected costs. Establish reporting and alerting systems to ensure prompt detection and mitigation of potential policy violations.

Compliance violation triggers and enforcement actions

When violations are detected, you should take enforcement actions to realign with policy. You can automate most violation triggers using the tools outlined in the Cost Management toolchain for Azure.

The following are examples of triggers:

- **Monthly budget deviations:** Discuss any deviations in monthly spending that exceed 20% forecast-versus-actual ratio with the billing unit leader. Record resolutions and changes in forecast.
- **Pace of adoption:** Any deviation at a subscription level exceeding 20% will trigger a review with billing unit leader. Record resolutions and changes in forecast.

Next steps

Use the Cost Management discipline template to document the processes and triggers that align to the current cloud adoption plan.

For guidance on executing Cost Management policies in alignment with adoption plans, see Cost Management discipline improvement.

Cost Management discipline improvement

Cost Management discipline improvement

3/6/2021 • 4 minutes to read • Edit Online

The Cost Management discipline attempts to address core business risks related to expenses incurred when hosting cloud-based workloads. Within the Five Disciplines of Cloud Governance, the Cost Management discipline is involved in controlling cost and usage of cloud resources with the goal of creating and maintaining a planned cost cycle.

This article outlines potential tasks your company perform to develop and mature your Cost Management discipline. These tasks can be broken down into planning, building, adopting, and operating phases of implementing a cloud solution. The tasks are then iterated on to allow the development of an incremental approach to cloud governance.

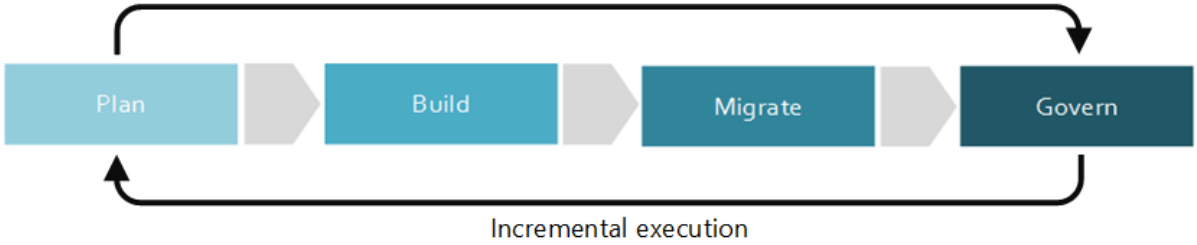


Figure 1: Phases of an incremental approach to cloud governance.

No single document can account for the requirements of all businesses. As such, this article outlines suggested minimum and potential example activities for each phase of the governance maturation process. The initial objective of these activities is to help you build a policy MVP and establish a framework for incremental policy improvement. Your cloud governance team will need to decide how much to invest in these activities to improve your Cost Management discipline capabilities.

Caution

Neither the minimum or potential activities outlined in this article are aligned to specific corporate policies or third-party compliance requirements. This guidance is designed to help facilitate the conversations that will lead to alignment of both requirements with a cloud governance model.

Planning and readiness

This phase of governance maturity bridges the divide between business outcomes and actionable strategies. During this process, the leadership team defines specific metrics, maps those metrics to the digital estate, and begins planning the overall migration effort.

Minimum suggested activities:

- Evaluate your Cost Management toolchain options.
- Develop a draft document for architecture guidelines and distribute to key stakeholders.
- Educate and involve the people and teams affected by the development of architecture guidelines.

Potential activities:

- Ensure budgetary decisions that support the business justification for your cloud strategy.
- Validate learning metrics that you use to report on the successful allocation of funding.
- Understand the desired cloud accounting model that affects how cloud costs should be accounted for.
- Become familiar with the digital estate plan and validate accurate costing expectations.

- Evaluate buying options to determine whether it's better to "pay as you go" or to make a precommitment by purchasing an Enterprise Agreement.
- Align business goals with planned budgets and adjust budgetary plans as necessary.
- Develop a goals and budget reporting mechanism to notify technical and business stakeholders at the end of each cost cycle.

Build and predeployment

Several technical and nontechnical prerequisites are required to successfully migrate an environment. This process focuses on the decisions, readiness, and core infrastructure that proceeds a migration.

Minimum suggested activities:

- Implement your Cost Management toolchain by rolling out in a predeployment phase.
- Update the architecture guidelines document and distribute to key stakeholders.
- Develop educational materials and documentation, awareness communications, incentives, and other programs to help drive user adoption.
- Determine whether your purchase requirements align with your budgets and goals.

Potential activities:

- Align your budgetary plans with the subscription strategy that defines your core ownership model.
- Use the Resource Consistency discipline strategy to enforce architecture and cost guidelines over time.
- Determine whether any cost anomalies affect your adoption and migration plans.

Adopt and migrate

Migration is an incremental process that focuses on the movement, testing, and adoption of applications or workloads in an existing digital estate.

Minimum suggested activities:

- Migrate your Cost Management toolchain from predeployment to production.
- Update the architecture guidelines document and distribute to key stakeholders.
- Develop educational materials and documentation, awareness communications, incentives, and other programs to help drive user adoption.

Potential activities:

- Implement your cloud accounting model.
- Ensure that your budgets reflect your actual spending during each release and adjust as necessary.
- Monitor changes in budgetary plans and validate with stakeholders if additional sign-offs are needed.
- Update changes to the architecture guidelines document to reflect actual costs.

Operate and post-implementation

After the transformation is complete, governance and operations must live on for the natural lifecycle of an application or workload. This phase of governance maturity focuses on the activities that commonly come after the solution is implemented and the transformation cycle begins to stabilize.

Minimum suggested activities:

- Customize your Cost Management toolchain based on your organization's changing needs.
- Consider automating any notifications and reports to reflect actual spending.
- Refine architecture guidelines to guide future adoption processes.

- Educate affected teams on a periodic basis to ensure ongoing adherence to the architecture guidelines.

Potential activities:

- Execute a quarterly cloud business review to communicate value delivered to the business and associated costs.
- Adjust plans quarterly to reflect changes to actual spending.
- Determine financial alignment to P&Ls for business unit subscriptions.
- Analyze stakeholder value and cost reporting methods on a monthly basis.
- Remediate underused assets and determine whether they're worth continuing.
- Detect misalignments and anomalies between the plan and actual spending.
- Assist the cloud adoption teams and the cloud strategy team with understanding and resolving these anomalies.

Next steps

Now that you understand the concept of cloud cost governance, review the Cost Management discipline best practices to find ways to reduce your overall spend.

Cost Management discipline best practices

Best practices for costing and sizing resources hosted in Azure

5/18/2021 • 22 minutes to read • Edit Online

While delivering the disciplines of governance, cost management is a recurring theme at the enterprise level. By optimizing and managing costs, you can ensure the long-term success of your Azure environment. It's critical that all teams (such as finance, management, and application development teams) understand associated costs and review them on a recurring basis.

Best practices by team and accountability

Cost management across the enterprise is a cloud governance and cloud operation function. All cost management decisions result in a change to the assets which support a workload. When those changes impact the architecture of a workload, additional considerations are required to minimize the impact on end users and business functions. The cloud adoption team who configured or developed that workload are likely to hold accountability for completing those types of changes.

Collective the centralized and decentralized teams should collaborate to ensure the following high level behaviors are implemented across your portfolio:

- **Tagging is critical to all governance.** Ensure all workloads and resources follow proper naming and tagging conventions and enforce tagging conventions using Azure Policy. This will help your centralized governance teams make wise cost management decisions.
- **Licensing alignment:** Proper allocation of Azure Hybrid Benefit and Azure Reserved VM Instances will significantly reduce your per unit cost for assets across your portfolio. These types of licensing decisions are generally established and maintained by central procurement functions. However, decentralized workload teams may want to be consulted on purchase and allocation of the licenses to minimize the cost of their individual workloads.
- **Identify right size opportunities.** Review your current resource utilization and performance requirements across the environment. Modify each resource to use the smallest instance or SKU that can support the performance requirements of each resource.
- **Shut down and deprovision unused resources:** Unused assets add cost in a cloud environment. Identify and terminate any resources that are adding to costs, but are not adding to business value.
- **Horizontal over vertical scale.** Using multiple small instances can allow for an easier scaling path that a single larger instance. This allows for scale automation, which creates cost optimization.

Operational cost management best practices

The following best practices are typically completed by a member of the cloud governance or cloud operations team, in accordance with patching and other scheduled maintenance processes. These best practices map to actionable guidance later in this article.

- **Tagging is critical to all governance:** Ensure all workloads and resources follow proper naming and tagging conventions and enforce tagging conventions using Azure Policy.
- **Licensing alignment:** The most immediate cost impact on a large portfolio of workloads will come from a well-planned license acquisition strategy. Purchase and allocation of Azure Hybrid Benefit, Azure Reserved VM Instances, Spot VMs and other buying strategies will rapidly reduce costs across your entire cloud portfolio.
- **Identify right size opportunities:** Review your current resource utilization and performance requirements

across the environment to identify resources which have remained underutilized for a period of time (generally more than 90 days).

- **Right-size provisioned SKUs:** Modify underutilized resource to use the smallest instance or SKU that can support the performance requirements of each resource.
- **Auto-shutdown for VMs:** When a VM isn't in constant use, consider automated shutdown. The VM won't be deleted or decommissioned, but it will stop consuming compute and memory costs until it's turned back on.
- **Auto-shutdown all nonproduction assets:** If a VM is part of a nonproduction environment, specifically development environments, establish an auto-shutdown policy to reduce unused costs. Whenever possible, use Azure DevTest Labs as a self-service option to help developers hold themselves accountable for cost.
- **Shut down and decommission unused resources:** Yes, we said it twice. If a resource hasn't been used in more than 90 days and doesn't have a clear uptime requirement, turn it off. More importantly, if a machine has been stopped or shut down for more than 90 days, then deprovision and delete that resource. Validate that any data retention policies are met through backup or other mechanisms.
- **Clean up orphaned disks:** Delete unused storage, especially VM storage that is no longer attached to any VMs.
- **Right-size redundancy:** If the resource doesn't require a high degree of redundancy, remove geo-redundant storage.
- **Adjust autoscale parameters:** Operational monitoring is likely to uncover usage patterns for various assets. When those usage patterns map to the parameters used to drive autoscale behaviors, it's common for the operations team to adjust autoscale parameters to meet seasonal demand or changes to budget allocations. Review workload cost management best practices for important precautions.

Workload cost management best practices

Before making architectural changes, consult the technical lead for the workload. Facilitating a review of the workload using Microsoft Azure Well-Architected Review and the Microsoft Azure Well-Architected Framework to guide decisions regarding the following types of architectural changes.

- **Azure App Service.** Verify production requirements for any Premium tier App Service plans. Without an understanding of the business requirements for a workload and the underlying assets configuration, it's difficult to determine whether a Premium tier plan is required.
- **Horizontal over vertical scale.** Using multiple small instances can allow for an easier scaling path that a single larger instance. This allows for scale automation, which creates cost optimization. Before a workload can scale horizontally, the technical team must verify that the application is idempotent. Achieving horizontal scale may first require changes to the code and configuration of various layers of the of the application.
- **Autoscale.** Enable autoscale on all app services to allow for a burstable number of smaller VMs. Enabling autoscale has the same idempotent requirement, which requires an understanding of the workload architecture. The workload and supporting assets must be approved for horizontal scaling and autoscaling by the adoption team, prior to any operational changes.
- **Implement serverless technologies:** VM workloads are often migrated "as is" to avoid downtime. Often VMs may host tasks that are intermittent, taking a short period to run, or alternatively many hours. For example, VMs that run scheduled tasks such as Windows task scheduler or PowerShell scripts. When these tasks aren't running, you're nevertheless absorbing VM and disk storage costs. After migration, consider rearchitecting layers of the workload to serverless technologies such as Azure Functions or Azure Batch jobs.

Actionable best practices

The remainder of this article provides tactical examples of operational best practices that a cloud governance or cloud operations team can follow to optimize cost across the enterprise.

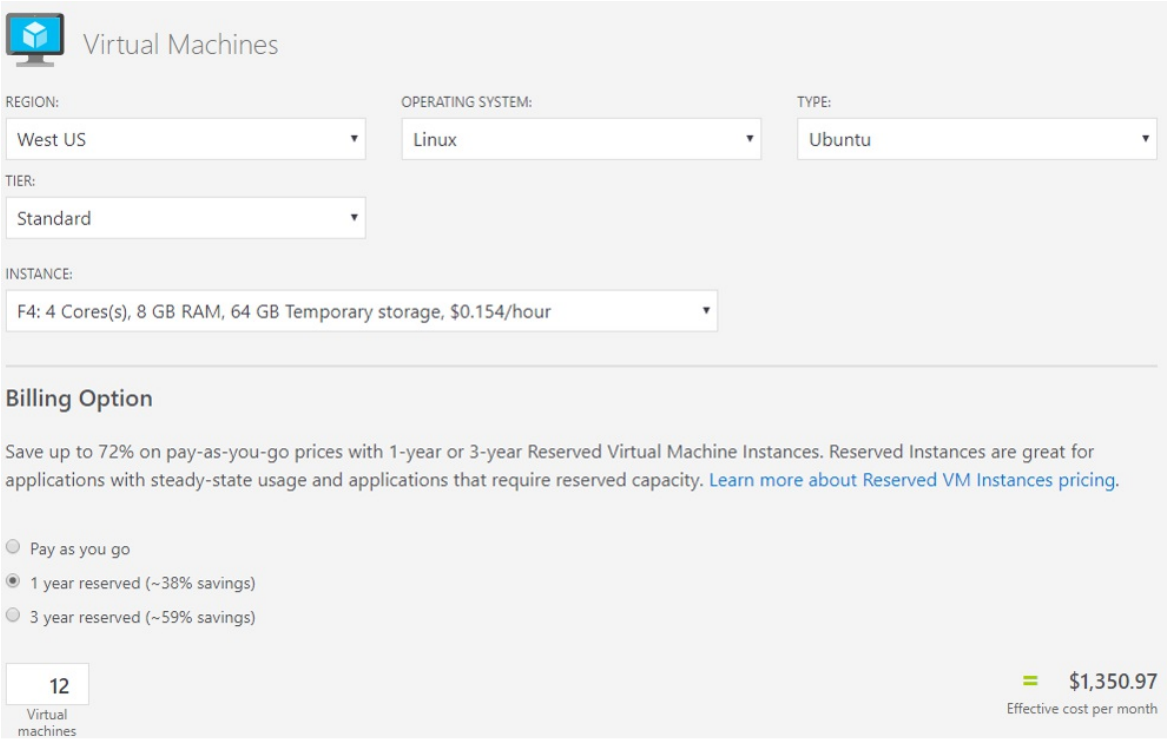
Before adoption

Before you move your workloads to the cloud, estimate the monthly cost of running them in Azure. Proactively managing cloud costs helps you adhere to your operating expense budget. The best practices in this section help you to estimate costs, and perform right-sizing for VMs and storage before a workload is deployed to the cloud.

Best practice: Estimate monthly workload costs

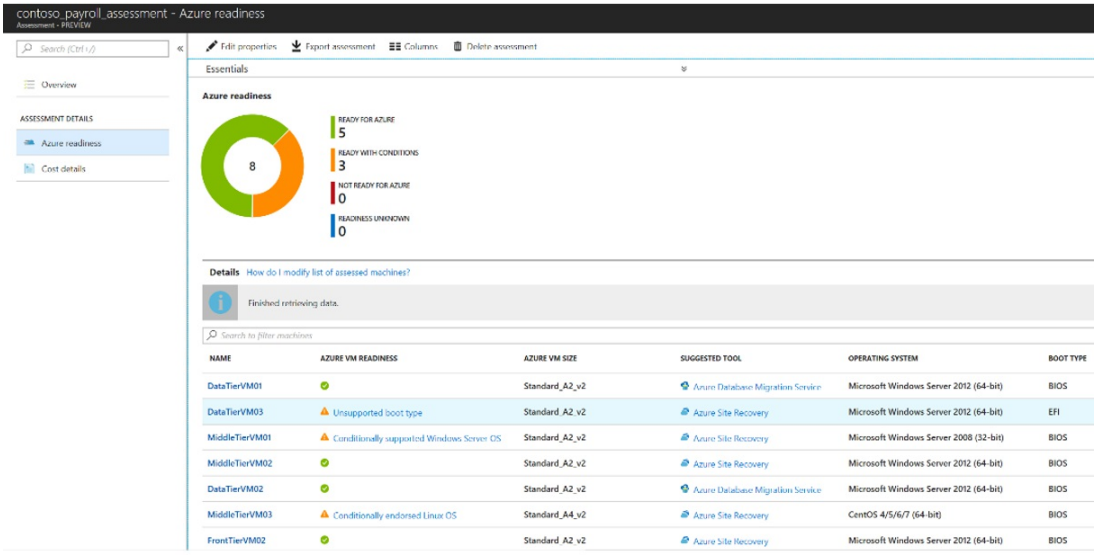
To forecast your monthly bill for Azure resources, there are several tools you can use.

- **Azure pricing calculator:** Select the products you want to estimate, for example VMs and storage, then input costs into the calculator to build an estimate.



Azure pricing calculator.

- **Azure Migrate:** To estimate costs, you need to review and account for all the resources required to run your workloads in Azure. To acquire this data, you create inventory of your assets, including servers, VMs, databases, and storage. You can use Azure Migrate to collect this information.
  - Azure Migrate discovers and assesses your on-premises environment to provide an inventory.
  - Azure Migrate can map and show you dependencies between VMs so that you have a complete picture.
  - An Azure Migrate assessment contains estimated cost.
    - **Compute costs:** Using the Azure VM size recommended when you create an assessment, Azure Migrate uses the Azure Billing APIs to calculate estimated monthly VM costs. The estimate considers the operating system, Software Assurance, Azure Reserved VM Instances, VM uptime, location, and currency settings. It aggregates the cost across all VMs in the assessment, and calculates a total monthly compute cost.
    - **Storage cost:** Azure Migrate calculates total monthly storage costs by aggregating the storage costs of all VMs in an assessment. You can calculate the monthly storage cost for a specific machine by aggregating the monthly cost of all disks attached to it.



Azure Migrate assessment.

Learn more:

- Use the Azure pricing calculator.
- Read the Azure Migrate overview.
- Read about Azure Migrate assessments.
- Learn more about the Azure Database Migration Service.

Best practice: Right-size VMs

You can choose various options when you deploy Azure VMs to support workloads. Each VM type has specific features and different combinations of CPU, memory, and disks. VMs are grouped as shown below:

TYPE	DETAILS	USAGE
General-purpose	Balanced CPU-to-memory.	Good for testing and development, small to midsize databases, low- to medium-volume traffic web servers.
Compute-optimized	High CPU-to-memory.	Good for medium-volume traffic web server, network appliances, batch processes, application servers.
Memory-optimized	High memory-to-CPU.	Good for relational databases, medium- to large-size cache, in-memory analytics.
Storage optimized	High disk throughput and I/O.	Suitable for big data, SQL, and NoSQL databases.
GPU optimized	Specialized VMs. Single or multiple GPUs.	Heavy graphics and video editing.
High performance	Fastest and most powerful CPU. VMs with optional high-throughput network interfaces (RDMA).	Critical high-performance applications.

- It's important to understand the pricing differences between these VMs, and the long-term budget effects.
- Each type has several series of VMs within it.
- Additionally, when you select a VM within a series, you can only scale the VM up and down within that series.

For example, a `DS2_v2` instance can scale up to `DS4_v2`, but it can't be changed to an instance of a different series such as a `F2S_v2` instance.

Learn more:

- Learn more about VM types and sizing, and map sizes to types.
- Plan VM sizing.
- Review a sample assessment for the fictional Contoso company.

Best practice: Select the right storage

Tuning and maintaining on-premises storage (SAN or NAS), and the networks to support them, can be costly and time-consuming. File (storage) data is commonly migrated to the cloud to help alleviate operational and management headaches. Microsoft provides several options for moving data to Azure, and you need to make decisions about those options. Picking the right storage type for data can save your organization several thousands of dollars every month. A few considerations:

- Data that isn't accessed much and isn't business-critical shouldn't be placed on the most expensive storage.
- Conversely, important business-critical data should be located on higher tier storage options.
- During adoption planning, take an inventory of data and classify it by importance, in order to map it to the most suitable storage. Consider budget and costs, as well as performance. Cost shouldn't necessarily be the main decision-making factor. Picking the least expensive option could expose the workload to performance and availability risks.

Storage data types

Azure provides different types of storage data.

DATA TYPE	DETAILS	USAGE
Blobs	Optimized to store massive amounts of unstructured objects, such as text or binary data.	Access data from everywhere over HTTP/HTTPS. Use for streaming and random access scenarios. For example, to serve images and documents directly to a browser, stream video and audio, and store backup and disaster recovery data.
Files	Managed file shares accessed over SMB 3.0.	Use when migrating on-premises file shares, and to provide multiple access and connections to file data.
Disks	Based on page blobs. Disk type (speed): Standard HDD, standard SSD, premium SSD, or ultra disks. Disk management: unmanaged (you manage disk settings and storage) or managed (you select the disk type and Azure manages the disk for you).	Use premium disks for VMs. Use managed disks for simple management and scaling.
Queues	Store and retrieve large numbers of messages accessed via authenticated calls (HTTP or HTTPS).	Connect application components with asynchronous message queuing.

DATA TYPE	DETAILS	USAGE
Tables	Store tables.	Now part of Azure Cosmos DB Table API.

Access tiers

Azure Storage provides different options for accessing block blob data. Selecting the right access tier helps ensure that you store block blob data in the most cost-effective manner.

ACCESS TIER	DETAILS	USAGE
Hot	Higher storage costs, lower access, and transaction costs This is the default access tier.	Use for data in active use that's accessed frequently.
Cool	Lower storage costs, higher access and transaction costs. Store for minimum of 30 days.	Store short-term, data is available but accessed infrequently.
Archive	Used for individual block blobs. Most cost-effective option for storage. Lowest storage costs, highest access and transaction costs.	Use for data that can tolerate several hours of retrieval latency and will reside in the archive tier for at least 180 days.

Storage account types

Azure provides different types of storage accounts and performance tiers.

ACCOUNT TYPE	DETAILS	USAGE
General-purpose v2 Standard tier	Supports blobs (block, page, append), files, disks, queues, and tables. Supports hot, cool, and archive access tiers. Zone-redundant storage (ZRS) is supported.	Use for most scenarios and most types of data. Standard storage accounts can be HDD or SSD-based.
General-purpose v2 Premium tier	Supports Blob Storage data (page blobs). Supports hot, cool, and archive access tiers. ZRS is supported. Stored on SSD.	Microsoft recommends using for all VMs.
General-purpose v1	Access tiering isn't supported. Doesn't support ZRS	Use if applications need the Azure classic deployment model.
Blob	Specialized storage account for storing unstructured objects. Provides block blobs and append blobs only (no file, queue, table, or disk storage services). Provides the same durability, availability, scalability, and performance as general-purpose v2.	You can't store page blobs in these accounts, and therefore can't store VHD files. You can set an access tier to hot or cool.

Storage redundancy options

Storage accounts can use different types of redundancy for resilience and high availability.

TYPE	DETAILS	USAGE
Locally redundant storage (LRS)	Protects against a local outage by replicating within a single storage unit to a separate fault domain and update domain. Keeps multiple copies of your data in one datacenter. Provides at least 99.999999999 percent (eleven 9's) durability of objects over a given year.	Consider whether your application stores data that can be easily reconstructed.
Zone-redundant storage (ZRS)	Protects again a datacenter outage by replicating across three storage clusters in a single region. Each storage cluster is physically separated and located in its own Availability Zone. Provides at least 99.9999999999 percent (twelve 9's) durability of objects over a given year by keeping multiple copies of your data across multiple datacenters or regions.	Consider whether you need consistency, durability, and high availability. Might not protect against a regional disaster when multiple zones are permanently affected.
Geo-redundant storage (GRS)	Protects against an entire region outage by replicating data to a secondary region that's hundreds of miles away from the primary. Provides at least 99.99999999999999 percent (sixteen 9's) durability of objects over a given year.	Replica data isn't available unless Microsoft initiates a failover to the secondary region. If failover occurs, read and write access is available.
Read-access geo-redundant storage (RA-GRS)	Similar to GRS. Provides at least 99.99999999999999 percent (sixteen 9's) durability of objects over a given year.	Provides and 99.99 percent read availability by allowing read access from the second region used for GRS.

Learn more:

- Review Azure Storage pricing.
- Learn to use the Azure Import/Export service to securely import large amounts of data to Azure Blob Storage and Azure Files.
- Compare blobs, files, and disk storage data types.
- Learn more about access tiers.
- Review different types of storage accounts.
- Learn about Azure Storage redundancy, including LRS, ZRS, GRS, and read-access GRS.
- Learn more about Azure Files.

After adoption

Prior to adoption, cost forecasts are dependent upon decisions made by workload owners and the cloud adoption team. While the governance team can aid in influencing those decisions, there's likely to be little action for the governance team to take.

Once resources are in production, data can be aggregated and trends analyzed at an environment level. This data will help the governance team make sizing and usage decisions independently, based on actual usage patterns and current state architecture.

- Analyze data to generate a budget baseline for Azure resource groups and resources.
- Identify patterns of use that would allow you to reduce size and stop or pause resources to further reduce your costs.

Best practices in this section include using Azure Hybrid Benefit and Azure Reserved VM Instances, reduce cloud spending across subscriptions, using Azure Cost Management + Billing for cost budgeting and analysis, monitoring resources and implementing resource group budgets, and optimizing monitoring, storage, and VMs.

Best practice: Take advantage of Azure Hybrid Benefit

Due to years of software investment in systems such as Windows Server and SQL Server, Microsoft is in a unique position to offer customers value in the cloud, with substantial discounts that other cloud providers can't necessarily provide.

An integrated Microsoft on-premises/Azure product portfolio generates competitive and cost advantages. If you currently have an operating system or other software licensing through Software Assurance (SA), you can take those licenses with you to the cloud for with Azure Hybrid Benefit.

Learn more:

- Take a look at the Azure Hybrid Benefit savings calculator.
- Learn more about Azure Hybrid Benefit for Windows Server.
- Review pricing guidance for SQL Server Azure VMs.

Best practice: Use Azure Reserved VM Instances

Most cloud platforms are set up as pay-as-you-go. This model presents disadvantages, since you don't necessarily know how dynamic workloads will be. When you specify clear intentions for a workload, you contribute to infrastructure planning.

Using Azure Reserved VM Instances, you prepay for either a one-year or three-year term for reserved instances.

- Prepayment provides a discount on the resources you use.
- You can significantly reduce costs for VM compute, SQL Database compute, Azure Cosmos DB, or other resources by up to 72% on pay-as-you-go prices.
- Reserved instances provide a billing discount, and don't affect the runtime state of your resources.
- You can cancel reserved instances.

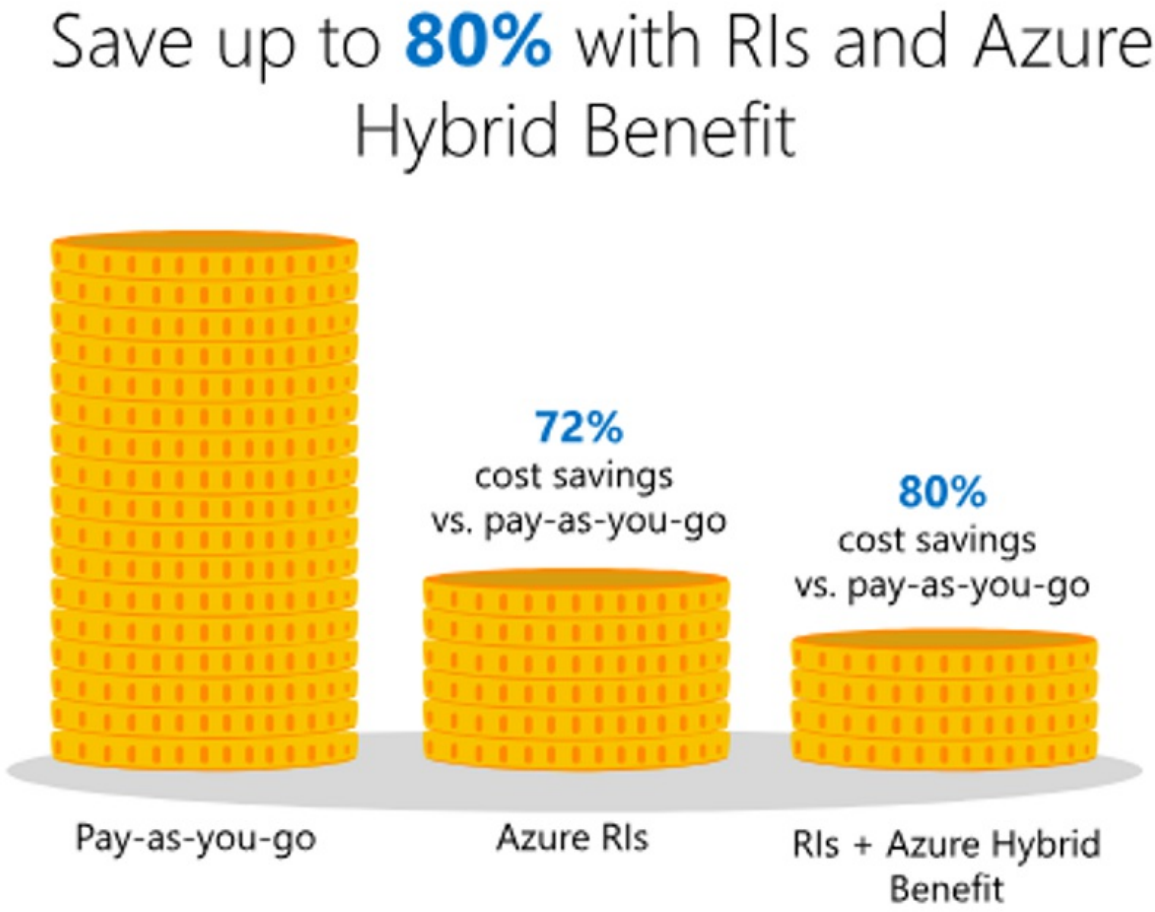


Figure 1: Azure Reserved VM instances.

Learn more:

- Learn about Azure Reservations.
- Read the reserved instances FAQ.
- Review pricing guidance for SQL Server Azure VMs.

Best practice: Aggregate cloud spend across subscriptions

It's inevitable that eventually you'll have more than one Azure subscription. For example, you might need an additional subscription to separate development and production boundaries, or you might have a platform that requires a separate subscription for each client. Having the ability to aggregate data reporting across all the subscriptions into a single platform is a valuable feature.

To do this, you can use Azure Cost Management + Billing APIs. Then, after aggregating data into a single source such as Azure SQL Database, you can use tools like Power BI to surface the aggregated data. You can create aggregated subscription reports, and granular reports. For example, for users who need proactive insights into cost management, you can create specific views of costs, based on department, resource group, or other information. You don't need to provide them with full access to Azure billing data.

Learn more:

- Read the Azure Consumption APIs overview.
- Learn about connecting to Azure Consumption Insights in Power BI Desktop.
- Learn to manage access to billing information for Azure using Azure role-based access control (Azure RBAC).

Best practice: Monitor resource utilization

In Azure you pay for what you use, when resources are consumed, and you don't pay when they aren't. For VMs,

billing occurs when a VM is allocated, and you aren't charged after a VM is deallocated. With this in mind you should monitor VMs in use, and verify VM sizing.

- Continually evaluate your VM workloads to determine baselines.
- For example, if your workload is used heavily Monday through Friday, 8am to 6pm, but hardly used outside those hours, you could downgrade VMs outside peak times. This might mean changing VM sizes, or using virtual machine scale sets to autoscale VMs up or down.
- Some companies "snooze", VMs by putting them on a calendar that specifies when they should be available, and when they're not needed.
- In addition to VM monitoring, you should monitor other networking resources such as ExpressRoute and virtual network gateways for under and over use.
- You can monitor VM usage using Microsoft tools such as Azure Cost Management + Billing, Azure Monitor, and Azure Advisor. Third-party tools are also available.

Learn more:

- Read overviews of Azure Monitor and Azure Advisor.
- Get Azure Advisor cost recommendations.
- Learn how to optimize costs from recommendations, and prevent unexpected charges.
- Learn about the Azure Resource Optimization (ARO) toolkit.

Best practice: Reduce nonproduction costs

Development, testing, and quality assurance (QA) environments are needed during development cycles. Unfortunately, it's common for those environments to stay provisioned long after they cease to be useful. A regular review of unused nonproduction environments can have an immediate impact on costs.

Additionally, consider general cost reductions for any nonproduction environments:

- Reduce nonproduction resources to use lower cost B-series VMs and standard storage.
- Reduce nonproduction compute costs by using Spot VMs.
- Apply Azure policies to require resource level cost-reductions for any nonproduction resources.

Learn more:

- Use tags to identify dev, test, or QA targets for resizing or termination.
- Auto-shutdown VMs sets a nightly termination time for VMs. Using this feature will stop nonproduction VMs each night, requiring developers to restart those VMs when they're ready to resume development.
- Spot VMs allows it to take advantage of unused Azure capacity at a significant cost savings. However, at any point in time when Azure needs the capacity back, the Azure infrastructure will evict Spot VMs.
- Encourage development teams to use Azure DevTest Labs to establish their own cost-control approaches and avoid impact of the standard auto-shutdown timing in the prior step.

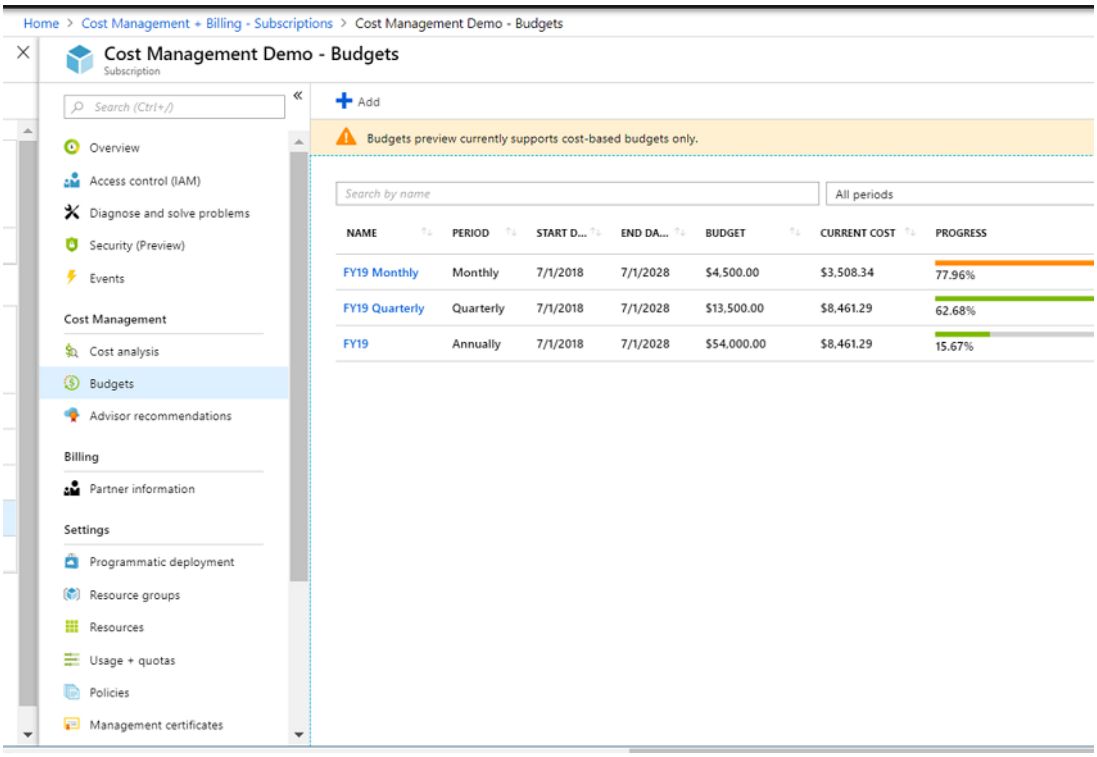
Best practice: Use Azure Cost Management + Billing

Microsoft provides Azure Cost Management + Billing to help you track spending:

- Helps you to monitor and control Azure spending, and optimize use of resources.
- Reviews your entire subscription and all of its resources, and makes recommendations.
- Provides a full API to integrate external tools and financial systems for reporting.
- Tracks resource usage and manage cloud costs with a single, unified view.
- Provides rich operational and financial insights to help you make informed decisions.

In Azure Cost Management + Billing, you can:

- **Create a budget:** Create a budget for financial accountability.
  - You can account for the services you consume or subscribe to for a specific period (monthly, quarterly, annually) and a scope (subscriptions/resource groups). For example, you can create an Azure subscription budget for a monthly, quarterly, or annual period.
    - After you create a budget, it's shown in cost analysis. Viewing your budget against current spending is one of the first steps needed when analyzing your costs and spending.
  - Email notifications can be sent when budget thresholds are reached.
  - You can export costs management data to Azure Storage, for analysis.



Budgets in Azure Cost Management + Billing.

- **Do a cost analysis:** Get a cost analysis to explore and analyze your organizational costs, to help you understand how costs are accrued, and identify spending trends.
  - Cost analysis is available to EA users.
  - You can view cost analysis data for various scopes, including by department, account, subscription, or resource group.
  - You can get a cost analysis that shows total costs for the current month, and accumulated daily costs.

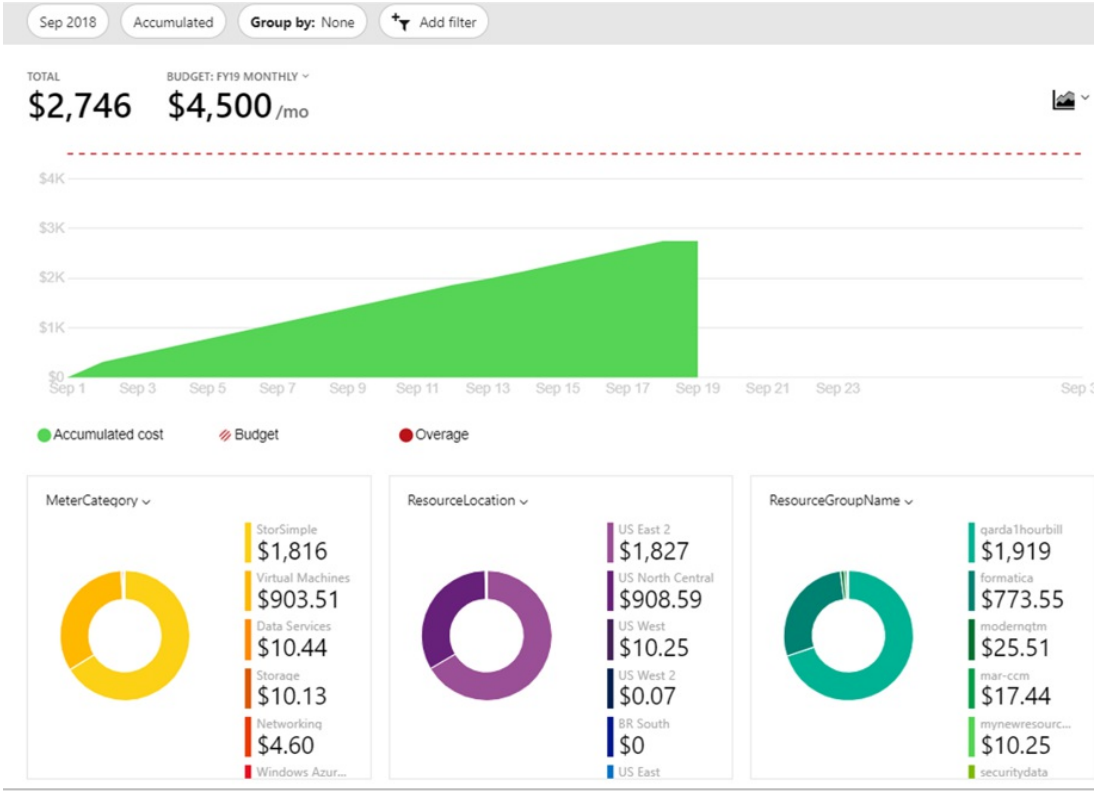


Figure: Azure Cost Management + Billing analysis.

- **Get recommendations:** Get Advisor recommendations that show you how you can optimize and improve efficiency.

Learn more:

- Read the Azure Cost Management + Billing overview.
- Learn how to optimize your cloud investment with Azure Cost Management + Billing.
- Learn how to use Azure Cost Management + Billing reports.
- Review a tutorial on optimizing costs from recommendations.
- Review the Azure Consumption APIs.

Best practice: Implement resource group budgets

Often, resource groups are used to represent cost boundaries. Together with this usage pattern, the Azure team continues to develop new and enhanced ways to track and analyze resource spending at different levels, including the ability to create budgets at the resource group and resources.

- A resource group budget helps you track the costs associated with a resource group.
- You can trigger alerts and run a wide variety of playbooks as the budget is reached or exceeded.

Learn more:

- Learn how to manage costs with Azure budgets.
- Review a tutorial on creating and managing an Azure budget.

Best practice: Review Azure Advisor recommendations

Azure Advisor cost recommendations identify opportunities to reduce costs. When budgets appear high or utilization appears low, use this report to find immediate opportunities to quickly align costs.

Learn more:

- Review Azure Advisor cost recommendations to take immediate actions.

Best practice: Optimize Azure Monitor retention

As you move resources into Azure and enable diagnostic logging for them, you generate a lot of log data. Typically this log data is sent to a storage account that's mapped to a Log Analytics workspace.

- The longer the log data retention period, the more data you'll have.
- Not all log data is equal, and some resources will generate more log data than others.
- Due to regulations and compliance, it's likely that you'll need to retain log data for some resources longer than others.
- Balance between optimizing your log storage costs and keeping the log data you need.
- We recommend evaluating and setting up the logging immediately after completing a migration, so that you aren't spending money retaining logs of no importance.

Learn more:

- Learn about monitoring usage and estimated costs.

Best practice: Optimize storage

If you followed best practices for selecting storage before adoption, you're probably reaping some benefits. You can probably optimize some additional storage costs. Over time, blobs and files become stale. Data might not be used anymore, but regulatory requirements might mean that you need to keep it for a certain period. As such, you might not need to store it on the high-performance storage that you used for the original adoption.

Identifying and moving stale data to cheaper storage areas can have a huge impact on your monthly storage budget and cost savings. Azure provides many ways to help you identify and then store this stale data.

- Take advantage of access tiers for general-purpose v2 storage by moving less important data from hot to cool and archived tiers.
- Use StorSimple to help move stale data based on customized policies.

Learn more:

- Learn more about access tiers.
- Read the StorSimple overview.
- Review StorSimple pricing.

Best practice: Automate VM optimization

The ultimate goal of running a VM in the cloud is to maximize the CPU, memory, and disk that it uses. If you discover VMs that aren't optimized, or have frequent periods when VMs aren't used, it makes sense to either shut them down, or downscale them using virtual machine scale sets.

You can optimize a VM with Azure Automation, virtual machine scale sets, auto-shutdown, and scripted or third-party solutions.

Learn more:

- Learn about vertical autoscaling.
- Review Azure DevTest Labs: Schedule VM auto-start.
- Learn how to start or stop VMs off hours in Azure Automation.
- Get more information about Azure Advisor, and the Azure Resource Optimization (ARO) toolkit.

Best practice: Use Logic Apps and runbooks with Budgets API

Azure provides a REST API that can access your tenant billing information.

- You can use the Budgets API to integrate external systems and workflows that are triggered by metrics that you build from the API data.
- You can pull usage and resource data into your preferred data analysis tools.
- The Azure Resource Usage API and the Azure Resource RateCard API can help you accurately predict and manage your costs.
- The APIs are implemented as a resource provider and are included in the APIs exposed by Azure Resource Manager.
- The Budgets API can be integrated with Azure Logic Apps and Azure Automation runbooks.

Learn more:

- Learn more about the Budgets API.
- Get insights into Azure usage with the Azure Billing APIs.

Next steps

With an understanding of best practices, examine the Cost Management toolchain to identify Azure tools and features to help you execute these best practices.

Cost Management toolchain for Azure

Cost Management tools in Azure

3/6/2021 • 2 minutes to read • Edit Online

The Cost Management discipline is one of the Five Disciplines of Cloud Governance. This discipline focuses on ways of establishing cloud spending plans, allocating cloud budgets, monitoring and enforcing cloud budgets, detecting costly anomalies, and adjusting the cloud governance plan when actual spending is misaligned.

The following is a list of Azure native tools that can help mature the policies and processes that support this discipline.

TOOL	AZURE PORTAL	AZURE COST MANAGEMENT + BILLING	POWER BI DESKTOP CONNECTOR	AZURE POLICY
Budget control	No	Yes	No	Yes
Monitor spending on single resource	Yes	Yes	Yes	No
Monitor spending across multiple resources	No	Yes	Yes	No
Control spending on single resource	Yes, manual sizing	Yes	No	Yes
Enforce spending across multiple resources	No	Yes	No	Yes
Enforce accounting metadata on resources	No	No	No	Yes
Monitor and detect trends	Yes	Yes	Yes	No
Detect spending anomalies	No	Yes	Yes	No
Socialize deviations	No	Yes	Yes	No

Security Baseline discipline overview

3/6/2021 • 2 minutes to read • Edit Online

Security baseline is one of the Five Disciplines of Cloud Governance within the Cloud Adoption Framework governance model. Security is a component of any IT deployment, and the cloud introduces unique security concerns. Many businesses are subject to regulatory requirements that make protecting sensitive data a major organizational priority when considering a cloud transformation. Identifying potential security threats to your cloud environment and establishing processes and procedures for addressing these threats should be a priority for any IT security or cybersecurity team. The Security Baseline discipline ensures technical requirements and security constraints are consistently applied to cloud environments, as those requirements mature.

> **NOTE**
> Security Baseline discipline does not replace the existing IT teams, processes, and procedures that your organization uses to secure cloud-deployed resources. The primary purpose of this discipline is to identify security-related business risks and provide risk-mitigation guidance to the IT staff responsible for security infrastructure. As you develop governance policies and processes make sure to involve relevant IT teams in your planning and review processes.

This article outlines the approach to developing a Security Baseline discipline as part of your cloud governance strategy. The primary audience for this guidance is your organization's cloud architects and other members of your cloud governance team. The decisions, policies, and processes that emerge from this discipline should involve engagement and discussions with relevant members of your IT and security teams, especially those technical leaders responsible for implementing networking, encryption, and identity services.

Making the correct security decisions is critical to the success of your cloud deployments and wider business success. If your organization lacks in-house expertise in cybersecurity, consider engaging external security consultants as a component of this discipline. Also consider engaging Microsoft Consulting Services, the Microsoft FastTrack cloud adoption service, or other external cloud adoption experts to discuss concerns related to this discipline.

Policy statements

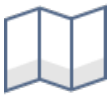
Actionable policy statements and the resulting architecture requirements serve as the foundation of a Security Baseline discipline. Use sample policy statements as a starting point for defining your Security Baseline policies.

Caution

The sample policies come from common customer experiences. To better align these policies to specific cloud governance needs, execute the following steps to create policy statements that meet your unique business needs.

Develop governance policy statements

The following steps offer examples and potential options to consider when developing your Security Baseline discipline. Use each step as a starting point for discussions within your cloud governance team and with affected business, IT, and security teams across your organization to establish the policies and processes needed to manage security-related risks.

	Security Baseline discipline template: Download the template for documenting a Security Baseline discipline.
	Business risks: Understand the motives and risks commonly associated with the Security Baseline discipline.
	Indicators and metrics: Indicators to understand whether it's the right time to invest in the Security Baseline discipline.
	Policy adherence processes: Suggested processes for supporting policy compliance in the Security Baseline discipline.
	Maturity: Align cloud management maturity with phases of cloud adoption.
	Toolchain: Azure services that can be implemented to support the Security Baseline discipline.

Next steps

Get started by evaluating business risks in a specific environment.

Understand business risks

Security Baseline discipline template

3/6/2021 • 2 minutes to read • Edit Online

The first step to implementing change is communicating what is desired. The same is true when changing governance practices. The following template provides a starting point for documenting and communicating policy statements that govern security related issues in the cloud.

As your discussions progress, use this template's structure as a model for capturing the business risks, risk tolerances, compliance processes, and tooling needed to define your organization's Security Baseline policy statements.

> **IMPORTANT**
> This template is a limited sample. Before updating this template to reflect your requirements, you should review the subsequent steps for defining an effective Security Baseline discipline within your cloud governance strategy.

Download the Security Baseline discipline template

Next steps

Solid governance practices start with an understanding of business risk. Review the article on business risks and begin to document the business risks that align with your current cloud adoption plan.

Understand business risks

Motivations and business risks in the Security Baseline discipline

3/6/2021 • 2 minutes to read • Edit Online

This article discusses the reasons that customers typically adopt a Security Baseline discipline within a cloud governance strategy. It also provides a few examples of potential business risks that can drive policy statements.

Relevance

Security is a key concern for any IT organization. Cloud deployments face many of the same security risks as workloads hosted in traditional on-premises datacenters. The nature of public cloud platforms, with a lack of direct ownership of the physical hardware storing and running your workloads, means cloud security requires its own policy and processes.

One of the primary things that sets cloud security governance apart from traditional security policy is the ease with which resources can be created, potentially adding vulnerabilities if security isn't considered before deployment. The flexibility that technologies like Software Defined Networking (SDN) provide for rapidly changing your cloud-based network topology can also easily modify your overall network attack surface in unforeseen ways. Cloud platforms also provide tools and features that can improve your security capabilities in ways not always possible in on-premises environments.

The amount you invest into security policy and processes will depend a great deal on the nature of your cloud deployment. Initial test deployments may only need the most basic of security policies in place, while a mission-critical workload will entail addressing complex and extensive security needs. All deployments will need to engage with the discipline at some level.

The Security Baseline discipline covers the corporate policies and manual processes that you can put in place to protect your cloud deployment against security risks.

> **NOTE**
> While it's important to understand the Identity Baseline discipline in the context of the Security Baseline discipline and how that relates to access control, the Five Disciplines of Cloud Governance treats it as a separate discipline.

Business risk

The Security Baseline discipline attempts to address core security-related business risks. Work with your business to identify these risks and monitor each of them for relevance as you plan for and implement your cloud deployments.

Risks differ between organizations. Use this list of common security-related risks as a starting point for discussions within your cloud governance team:

- **Data breach:** Inadvertent exposure or loss of sensitive cloud-hosted data can lead to losing customers, contractual issues, or legal consequences.
- **Service disruption:** Outages and other performance issues due to insecure infrastructure interrupts normal operations and can result in lost productivity or lost business.

Next steps

Use the Security Baseline discipline template to document business risks that are likely to be introduced by the current cloud adoption plan.

Once an understanding of realistic business risks is established, the next step is to document the business's tolerance for risk and the indicators and key metrics to monitor that tolerance.

Understand indicators, metrics, and risk tolerance

Risk tolerance metrics and indicators in the Security Baseline discipline

3/6/2021 • 4 minutes to read • Edit Online

Learn to quantify business risk tolerance associated with the Security Baseline discipline. Defining metrics and indicators helps to create a business case for investing in the maturity of this discipline.

Metrics

The Security Baseline discipline generally focuses on identifying potential vulnerabilities in your cloud deployments. As part of your risk analysis you'll want to gather data related to your security environment to determine how much risk you face, and how important investment in your Security Baseline discipline is for your planned cloud deployments.

Every organization has different security environments and requirements and different potential sources of security data. The following are examples of useful metrics that you should gather to help evaluate risk tolerance within the Security Baseline discipline:

- **Data classification:** Number of cloud-stored data and services that are unclassified according to on your organization's privacy, compliance, or business impact standards.
- **Number of sensitive data stores:** Number of storage endpoints or databases that contain sensitive data and should be protected.
- **Number of unencrypted data stores:** Number of sensitive data stores that are not encrypted.
- **Attack surface:** How many total data sources, services, and applications will be cloud-hosted. What percentage of these data sources are classified as sensitive? What percentage of these applications and services are mission-critical?
- **Covered standards:** Number of security standards defined by the security team.
- **Covered resources:** Deployed assets that are covered by security standards.
- **Overall standards compliance:** Ratio of compliance adherence to security standards.
- **Attacks by severity:** How many coordinated attempts to disrupt your cloud-hosted services, such as through distributed denial of service (DDoS) attacks, does your infrastructure experience? What is the size and severity of these attacks?
- **Malware protection:** Percentage of deployed virtual machines (VMs) that have all required anti-malware, firewall, or other security software installed.
- **Patch latency:** How long has it been since VMs have had OS and software patches applied.
- **Security health recommendations:** Number of security software recommendations for resolving health standards for deployed resources, organized by severity.

Risk tolerance indicators

Cloud platforms provide a baseline set of features that enable small deployment teams to configure basic security settings without extensive additional planning. As a result, small dev/test or experimental first workloads that do not include sensitive data represent a relatively low level of risk, and will likely not need much in the way of formal Security Baseline policy. As soon as important data or mission-critical functionality is moved to the cloud, security risks increase, while tolerance for those risks diminishes rapidly. As more of your data and functionality is deployed to the cloud, the more likely you need an increased investment in the Security Baseline discipline.

In the early stages of cloud adoption, work with your IT security team and business stakeholders to identify business risks related to security, then determine an acceptable baseline for security risk tolerance. This section of the Cloud Adoption Framework provides examples, but the detailed risks and baselines for your company or deployments may be different.

Once you have a baseline, establish minimum benchmarks representing an unacceptable increase in your identified risks. These benchmarks act as triggers for when you need to take action to remediate these risks. The following are a few examples of how security metrics, such as those discussed above, can justify an increased investment in the Security Baseline discipline.

- **Mission-critical workloads trigger.** A company deploying mission-critical workloads to the cloud should invest in the Security Baseline discipline to prevent potential disruption of service or sensitive data exposure.
- **Protected data trigger.** A company hosting data on the cloud that can be classified as confidential, private, or otherwise subject to regulatory concerns. They need a Security Baseline discipline to ensure that this data is not subject to loss, exposure, or theft.
- **External attacks trigger.** A company that experiences serious attacks against their network infrastructure *x* times per month could benefit from the Security Baseline discipline.
- **Standards compliance trigger.** A company with more than *x%* of resources out of security standards compliance should invest in the Security Baseline discipline to ensure standards are applied consistently across your IT infrastructure.
- **Cloud estate size trigger.** A company hosting more than *x* applications, services, or data sources. Large cloud deployments can benefit from investment in the Security Baseline discipline to ensure that their overall attack surface is properly protected against unauthorized access or other external threats.
- **Security software compliance trigger.** A company where less than *x%* of deployed virtual machines have all required security software installed. A Security Baseline discipline can be used to ensure software is installed consistently on all software.
- **Patching trigger.** A company where deployed virtual machines or services where OS or software patches have not been applied in the last *x* days. A Security Baseline discipline can be used to ensure patching is kept up-to-date within a required schedule.
- **Security-focused.** Some companies will have strong security and data confidentiality requirements even for test and experimental workloads. These companies will need to invest in the Security Baseline discipline before any deployments can begin.

The exact metrics and triggers you use to gauge risk tolerance and the level of investment in the Security Baseline discipline will be specific to your organization, but the examples above should serve as a useful base for discussion within your cloud governance team.

Next steps

Use the Security Baseline discipline template to document metrics and tolerance indicators that align to the current cloud adoption plan.

Review sample Security Baseline policies as a starting point to develop your own policies to address specific business risks aligned with your cloud adoption plans.

Review sample policies

Security Baseline sample policy statements

3/6/2021 • 4 minutes to read • Edit Online

Individual cloud policy statements are guidelines for addressing specific risks identified during your risk assessment process. These statements should provide a concise summary of risks and plans to deal with them. Each statement definition should include these pieces of information:

- **Technical risk:** A summary of the risk this policy will address.
- **Policy statement:** A clear summary explanation of the policy requirements.
- **Technical options:** Actionable recommendations, specifications, or other guidance that IT teams and developers can use when implementing the policy.

The following sample policy statements address common security-related business risks. These statements are examples you can reference when drafting policy statements to address your organization's needs. These examples are not meant to be proscriptive, and there are potentially several policy options for dealing with each identified risk. Work closely with business, security, and IT teams to identify the best policies for your unique set of risks.

Asset classification

Technical risk: Assets that are not correctly identified as mission-critical or involving sensitive data may not receive sufficient protections, leading to potential data leaks or business disruptions.

Policy statement: All deployed assets must be categorized by criticality and data classification. Classifications must be reviewed by the cloud governance team and the application owner before deployment to the cloud.

Potential design option: Establish resource tagging standards and ensure IT staff apply them consistently to any deployed resources using Azure resource tags.

Data encryption

Technical risk: There is a risk of protected data being exposed during storage.

Policy statement: All protected data must be encrypted when at rest.

Potential design option: See the Azure encryption overview article for a discussion of how data at rest encryption is performed on the Azure platform. Additional controls such as in account data encryption and control over how storage account settings can be changed should also be considered.

Network isolation

Technical risk: Connectivity between networks and subnets within networks introduces potential vulnerabilities that can result in data leaks or disruption of mission-critical services.

Policy statement: Network subnets containing protected data must be isolated from any other subnets. Network traffic between protected data subnets is to be audited regularly.

Potential design option: In Azure, network and subnet isolation is managed through Azure Virtual Network.

Secure external access

Technical risk: Allowing access to workloads from the public internet introduces a risk of intrusion resulting in

unauthorized data exposure or business disruption.

Policy statement: No subnet containing protected data can be directly accessed over public internet or across datacenters. Access to those subnets must be routed through intermediate subnets. All access into those subnets must come through a firewall solution capable of performing packet scanning and blocking functions.

Potential design option: In Azure, secure public endpoints by deploying a perimeter network between the public internet and your cloud-based network. Consider deployment, configuration, and automation of Azure Firewall.

DDoS protection

Technical risk: Distributed denial of service (DDoS) attacks can result in a business interruption.

Policy statement: Deploy automated DDoS mitigation mechanisms to all publicly accessible network endpoints. No public-facing web site backed by IaaS should be exposed to the internet without DDoS.

Potential design option: Use Azure DDoS Protection Standard to minimize disruptions caused by DDoS attacks.

Secure on-premises connectivity

Technical risk: Unencrypted traffic between your cloud network and on-premises over the public internet is vulnerable to interception, introducing the risk of data exposure.

Policy statement: All connections between the on-premises and cloud networks must take place either through a secure encrypted VPN connection or a dedicated private WAN link.

Potential design option: In Azure, use ExpressRoute or Azure VPN to establish private connections between your on-premises and cloud networks.

Network monitoring and enforcement

Technical risk: Changes to network configuration can lead to new vulnerabilities and data exposure risks.

Policy statement: Governance tooling must audit and enforce network configuration requirements defined by the security baseline team.

Potential design option: In Azure, network activity can be monitored using Azure Network Watcher, and Azure Security Center can help identify security vulnerabilities. Azure Policy allows you to restrict network resources and resource configuration policy according to limits defined by the security team.

Security review

Technical risk: Over time, new security threats and attack types emerge, increasing the risk of exposure or disruption of your cloud resources.

Policy statement: Trends and potential exploits that could affect cloud deployments should be reviewed regularly by the security team to provide updates to Security Baseline tools used in the cloud.

Potential design option: Establish a regular security review meeting that includes relevant IT and governance team members. Review existing security data and metrics to establish gaps in current policy and Security Baseline tools, and update policy to remediate any new risks. Use Azure Advisor and Azure Security Center to gain actionable insights on emerging threats specific to your deployments.

Next steps

Use the samples mentioned in this article as a starting point to develop policies that address specific security risks that align with your cloud adoption plans.

To begin developing your own custom Security Baseline policy statements, download the Security Baseline discipline template.

To accelerate adoption of this discipline, choose the actionable governance guide that most closely aligns with your environment. Then modify the design to incorporate your specific corporate policy decisions.

Building on risks and tolerance, establish a process for governing and communicating Security Baseline policy adherence.

Establish policy compliance processes

Security Baseline policy compliance processes

3/6/2021 • 5 minutes to read • Edit Online

This article discusses an approach to policy adherence processes that govern the Security Baseline discipline. Effective governance of cloud security starts with recurring manual processes designed to detect vulnerabilities and impose policies to remediate those security risks. This requires regular involvement of the cloud governance team and interested business and IT stakeholders to review and update policy and ensure policy compliance. In addition, many ongoing monitoring and enforcement processes can be automated or supplemented with tooling to reduce the overhead of governance and allow for faster response to policy deviation.

Planning, review, and reporting processes

The best Security Baseline tools in the cloud are only as good as the processes and policies that they support. The following is a set of example processes commonly involved in the Security Baseline discipline. Use these examples as a starting point when planning the processes that will allow you to continue to update security policy based on business change and feedback from the security and IT teams tasked with turning governance guidance into action.

Initial risk assessment and planning: As part of your initial adoption of the Security Baseline discipline, identify your core business risks and tolerances related to cloud security. Use this information to discuss specific technical risks with members of your IT and security teams and develop a baseline set of security policies for mitigating these risks to establish your initial governance strategy.

Deployment planning: Before deploying any workload or asset, perform a security review to identify any new risks and ensure all access and data security policy requirements are met.

Deployment testing: As part of the deployment process for any workload or asset, the cloud governance team, in cooperation with your corporate security teams, will be responsible for reviewing the deployment to validate security policy compliance.

Annual planning: On an annual basis, perform a high-level review of Security Baseline strategy. Explore future corporate priorities and updated cloud adoption strategies to identify potential risk increase and other emerging security needs. Also use this time to review the latest security baseline best practices and integrate these into your policies and review processes.

Quarterly review and planning: On a quarterly basis perform a review of security audit data and incident reports to identify any changes required in security policy. As part of this process, review the current cybersecurity landscape to proactively anticipate emerging threats, and update policy as appropriate. After the review is complete, align design guidance with updated policy.

This planning process is also a good time to evaluate the current membership of your cloud governance team for knowledge gaps related to new or changing policy and risks related to security. Invite relevant IT staff to participate in reviews and planning as either temporary technical advisors or permanent members of your team.

Education and training: On a bimonthly basis, offer training sessions to make sure IT staff and developers are up-to-date on the latest security policy requirements. As part of this process review and update any documentation, guidance, or other training assets to ensure they're in sync with the latest corporate policy statements.

Monthly audit and reporting reviews: On a monthly basis, perform an audit on all cloud deployments to

assure their continued alignment with security policy. Review security related activities with IT staff and identify any compliance issues not already handled as part of the ongoing monitoring and enforcement process. The result of this review is a report for the cloud strategy team and each cloud adoption team to communicate overall adherence to policy. The report is also stored for auditing and legal purposes.

Processes for ongoing monitoring

A successful Security Baseline strategy is successful depends on visibility into the current and past state of your cloud infrastructure. Without the ability to analyze the relevant metrics and data of your cloud resources security health and activity, you cannot identify changes in your risks or detect violations of your risk tolerances. The ongoing governance processes discussed above require quality data to ensure policy can be modified to better protect your infrastructure against changing threats and security requirements.

Ensure that your security and IT teams have implemented automated monitoring systems for your cloud infrastructure that capture the relevant logs data you need to evaluate risk. Be proactive in monitoring these systems to ensure prompt detection and mitigation of potential policy violation, and ensure your monitoring strategy is in line with security needs.

Violation triggers and enforcement actions

Because security noncompliance can lead to critical and data exposure and service disruption risks, the cloud governance team should have visibility into serious policy violations. Ensure IT staff have clear escalation paths for reporting security issues to the governance team members best suited to identify and verify that policy issues are mitigated.

When violations are detected, you should take actions to realign with policy as soon as possible. Your IT team can automate most violation triggers using the tools outlined in the Security Baseline toolchain for Azure.

The following triggers and enforcement actions provide examples you can reference when planning how to use monitoring data to resolve policy violations:

- **Increase in attacks detected.** If any resource experiences a 25% increase in brute force or DDoS attacks, discuss with IT security staff and workload owner to determine remedies. Track issue and update guidance if policy revision is necessary to prevent future incidents.
- **Unclassified data detected.** Any data source without an appropriate privacy, security, or business impact classification will have external access denied until the classification is applied by the data owner and the appropriate level of data protection applied.
- **Security health issue detected.** Disable access to any virtual machines (VMs) that have known access or malware vulnerabilities identified until appropriate patches or security software can be installed. Update policy guidance to account for any newly detected threats.
- **Network vulnerability detected.** Access to any resource not explicitly allowed by the network access policies should trigger an alert to IT security staff and the relevant workload owner. Track issue and update guidance if policy revision is necessary to mitigate future incidents.

Next steps

Use the Security Baseline discipline template to document the processes and triggers that align to the current cloud adoption plan.

For guidance on executing cloud management policies in alignment with adoption plans, see the article on discipline improvement.

Security Baseline discipline improvement

Security Baseline discipline improvement

5/26/2021 • 5 minutes to read • Edit Online

The Security Baseline discipline focuses on ways of establishing policies that protect the network, assets, and most importantly the data that will reside on a cloud provider's solution. Within the Five Disciplines of Cloud Governance, the Security Baseline discipline includes classification of the digital estate and data. It also includes documentation of risks, business tolerance, and mitigation strategies associated with the security of the data, assets, and network. From a technical perspective, this also includes involvement in decisions regarding encryption, network requirements, hybrid identity strategies, and the processes used to develop Security Baseline policies for the cloud.

This article outlines some potential tasks your company can engage in to better develop and mature the Security Baseline discipline. These tasks can be broken down into planning, building, adopting, and operating phases of implementing a cloud solution, which are then iterated on allowing the development of an incremental approach to cloud governance.

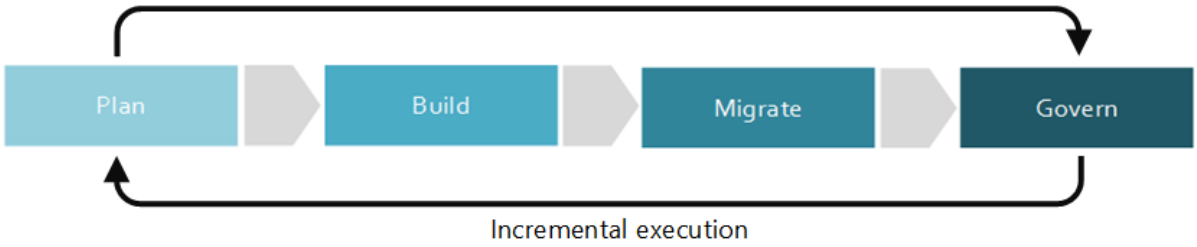


Figure 1: Phases of an incremental approach to cloud governance.

It's impossible for any one document to account for the requirements of all businesses. As such, this article outlines suggested minimum and potential example activities for each phase of the governance maturation process. The initial objective of these activities is to help you build a policy MVP and establish a framework for incremental policy improvement. Your cloud governance team will need to decide how much to invest in these activities to improve your Security Baseline discipline.

Caution

Neither the minimum or potential activities outlined in this article are aligned to specific corporate policies or third-party compliance requirements. This guidance is designed to help facilitate the conversations that will lead to alignment of both requirements with a cloud governance model.

Planning and readiness

This phase of governance maturity bridges the divide between business outcomes and actionable strategies. During this process, the leadership team defines specific metrics, maps those metrics to the digital estate, and begins planning the overall migration effort.

Minimum suggested activities:

- Evaluate your Security Baseline toolchain options.
- Develop a draft architecture guidelines document and distribute to key stakeholders.
- Educate and involve the people and teams affected by the development of architecture guidelines.
- Add prioritized security tasks to your migration backlog.

Potential activities:

- Define a data classification schema.

- Conduct a digital estate planning process to inventory the current IT assets powering your business processes and supporting operations.
- Conduct a policy review to begin the process of modernizing existing corporate IT security policies, and define MVP policies addressing known risks.
- Review your cloud platform's security guidelines. For Azure these can be found in the Microsoft Service Trust Portal.
- Determine whether your Security Baseline policy includes a security development lifecycle.
- Evaluate network, data, and asset-related business risks based on the next one to three releases, and gauge your organization's tolerance for those risks.
- Review Microsoft's top trends in cybersecurity report for an overview of the current security landscape.
- Consider developing a DevSecOps role in your organization.

Build and predeployment

Several technical and nontechnical prerequisites are required to successful migrate an environment. This process focuses on the decisions, readiness, and core infrastructure that proceeds a migration.

Minimum suggested activities:

- Implement your Security Baseline toolchain by rolling out in a predeployment phase.
- Update the architecture guidelines document and distribute to key stakeholders.
- Implement security tasks on your prioritized migration backlog.
- Develop educational materials and documentation, awareness communications, incentives, and other programs to help drive user adoption.

Potential activities:

- Determine your organization's encryption strategy for cloud-hosted data.
- Evaluate your cloud deployment's identity strategy. Determine how your cloud-based identity solution will coexist or integrate with on-premises identity providers.
- Determine network boundary policies for your Software Defined Networking (SDN) design to ensure secure virtualized networking capabilities.
- Evaluate your organization's least-privilege access policies, and use task-based roles to provide access to specific resources.
- Apply security and monitoring mechanisms to all cloud services and virtual machines.
- Automate security policies where possible.
- Review your Security Baseline policy and determine whether you need to modify your plans according to best practices guidance such as those outlined in the security development lifecycle.

Adopt and migrate

Migration is an incremental process that focuses on the movement, testing, and adoption of applications or workloads in an existing digital estate.

Minimum suggested activities:

- Migrate your Security Baseline toolchain from predeployment to production.
- Update the architecture guidelines document and distribute to key stakeholders.
- Develop educational materials and documentation, awareness communications, incentives, and other programs to help drive user adoption.

Potential activities:

- Review the latest security baseline and threat information to identify any new business risks.

- Gauge your organization's tolerance to handle new security risks that may arise.
- Identify deviations from policy, and enforce corrections.
- Adjust security and access control automation to ensure maximum policy compliance.
- Validate that the best practices defined during the build and predeployment phases are properly executed.
- Review your least-privilege access policies and adjust access controls to maximize security.
- Test your Security Baseline toolchain against your workloads to identify and resolve any vulnerabilities.

Operate and post-implementation

Once the transformation is complete, governance and operations must live on for the natural lifecycle of an application or workload. This phase of governance maturity focuses on the activities that commonly come after the solution is implemented and the transformation cycle begins to stabilize.

Minimum suggested activities:

- Validate and refine your Security Baseline toolchain.
- Customize notifications and reports to alert you of potential security issues.
- Refine the architecture guidelines to guide future adoption processes.
- Communicate and educate the affected teams periodically to ensure ongoing adherence to architecture guidelines.

Potential activities:

- Discover patterns and behavior for your workloads and configure your monitoring and reporting tools to identify and notify you of any abnormal activity, access, or resource usage.
- Continuously update your monitoring and reporting policies to detect the latest vulnerabilities, exploits, and attacks.
- Have procedures in place to quickly stop unauthorized access and disable resources that may have been compromised by an attacker.
- Regularly review the latest security best practices and apply recommendations to your security policy, automation, and monitoring capabilities where possible.

Next steps

Now that you understand the concept of cloud security governance, move on to learn more about what security and best practices guidance Microsoft provides for Azure.

Learn about security guidance for Azure Introduction to Azure security Learn about logging, reporting, and monitoring

Cloud-native Security Baseline policy

3/6/2021 • 6 minutes to read • Edit Online

The Security Baseline discipline is one of the Five Disciplines of Cloud Governance. This discipline focuses on general security topics including protection of the network, digital assets, and data. This article discusses a cloud-native sample policy for the Security Baseline discipline.

> **NOTE**
> Microsoft is in no position to dictate corporate or IT policy. This article will help you prepare for an internal policy review. It's assumed that this sample policy will be extended, validated, and tested against your corporate policy before attempting to use it. Any use of this sample policy as-is is discouraged.

Policy alignment

This sample policy synthesizes a cloud-native scenario, meaning that the tools and platforms provided by Azure are sufficient to manage business risks involved in a deployment. In this scenario, it's assumed that a simple configuration of the default Azure services provides sufficient asset protection.

Cloud security and compliance

Security is integrated into every aspect of Azure, offering unique security advantages derived from global security intelligence, sophisticated customer-facing controls, and a secure, hardened infrastructure. This powerful combination helps protect your applications and data, support your compliance efforts, and provide cost-effective security for organizations of all sizes. This approach creates a strong starting position for any security policy, but does not negate the need for equally strong security practices related to the security services being used.

Built-in security controls

It's hard to maintain a strong security infrastructure when security controls are not intuitive and need to be configured separately. Azure includes built-in security controls across a variety of services that help you protect data and workloads quickly and manage risk across hybrid environments. Integrated partner solutions also let you easily transition existing protections to the cloud.

Cloud-native identity policies

Identity is becoming the new boundary control plane for security, taking over that role from the traditional network-centric perspective. Network perimeters have become increasingly porous and that perimeter defense cannot be as effective as it was before the advent of bring your own device (BYOD) and cloud applications. Azure identity management and access control enable seamless secure access to all your applications.

A sample cloud-native policy for identity across cloud and on-premises directories, could include requirements like the following:

- Authorized access to resources with Azure role-based access control (Azure RBAC), multi-factor authentication, and single sign-on (SSO).
- Quick mitigation of user identities suspected of compromise.
- Just-in-time (JIT), just-enough access granted on a task-by-task basis to limit exposure of overprivileged admin credentials.
- Extended user identity and access to policies across multiple environments through Azure Active Directory.

While it's important to understand the Identity Baseline discipline in the context of the Security Baseline discipline, the Five Disciplines of Cloud Governance treats it as a separate discipline.

Network access policies

Network control includes the configuration, management, and securing of network elements such as virtual networking, load balancing, DNS, and gateways. The controls provide a means for services to communicate and interoperate. Azure includes a robust and secure networking infrastructure to support your application and service connectivity requirements. Network connectivity is possible between resources located in Azure, between on-premises and Azure hosted resources, and to and from the internet and Azure.

A cloud-native policy for network controls may include requirements like the following:

- Hybrid connections to on-premises resources, might not be allowed in a cloud-native policy. Should a hybrid connection prove necessary, a more robust enterprise security policy sample would be a more relevant reference.
- Users can establish secure connections to and within Azure using virtual networks and network security groups.
- The native Windows Azure Firewall protects hosts from malicious network traffic by limiting port access. A good example of this policy is a requirement to block or not enable traffic directly to a VM over SSH/RDP.
- Services like the Azure Web Application Firewall (WAF) on Azure Application Gateway and Azure DDoS protection safeguard applications and ensure availability for virtual machines running in Azure. These features should not be disabled.

Data protection

One of the keys to data protection in the cloud is accounting for the possible states in which your data may occur, and what controls are available for each state. For the purpose of Azure data security and encryption best practices, recommendations focus on the following data states:

- Data encryption controls are built into services from virtual machines to storage and SQL Database.
- As data moves between clouds and customers, it can be protected using industry-standard encryption protocols.
- Azure Key Vault enables users to safeguard and control cryptographic keys, passwords, connection strings and certificates used by cloud applications and services.
- Azure Information Protection will help classify, label, and protect your sensitive data within applications.

While these features are built into Azure, each of the above requires configuration and could increase costs. Alignment of each cloud-native feature with a data classification strategy is highly suggested.

Security monitoring

Security monitoring is a proactive strategy that audits your resources to identify systems that do not meet organizational standards or best practices. Azure Security Center provides unified security baseline and Microsoft Defender for Identity across hybrid cloud workloads. With Security Center, you can apply security policies across your workloads, limit your exposure to threats, and detect and respond to attacks, including:

- Unified view of security across all on-premises and cloud workloads with Azure Security Center.
- Continuous monitoring and security assessments to ensure compliance and remediate any vulnerabilities.
- Interactive tools and contextual threat intelligence for streamlined investigation.
- Extensive logging and integration with existing security information.
- Reduces the need for expensive, nonintegrated, one off security solutions.

Extend cloud-native policies

Using the cloud can reduce some of the security burden. Microsoft provides physical security for Azure datacenters and helps protect the cloud platform against infrastructure threats such as a DDoS attack. Given that Microsoft has thousands of cybersecurity specialists working on security every day, the resources to detect,

prevent, or mitigate cyberattacks are considerable. In fact, while organizations used to worry about whether the cloud was secure, most now understand that the level of investment in people and specialized infrastructure made by vendors like Microsoft makes the cloud more secure than most on-premises datacenters.

Using the cloud can reduce some of the security burden. Microsoft provides physical security for Azure datacenters and helps protect the cloud platform against infrastructure threats such as a DDoS attack. Given that Microsoft has thousands of cybersecurity specialists working on security every day, the resources to detect, prevent, or mitigate cyberattacks are considerable. In fact, while organizations used to worry about whether the cloud was secure, most now understand that the level of investment in people and specialized infrastructure made by vendors like Microsoft makes the cloud more secure than most on-premises datacenters.

Even with this investment in a cloud-native security baseline, it's suggested that any Security Baseline policy extend the default cloud-native policies. The following are examples of extended policies that should be considered, even in a cloud-native environment:

- **Secure VMs.** Security should be every organization's top priority, and doing it effectively requires several things. You must assess your security state, protect against security threats, and then detect and respond rapidly to threats that occur.
- **Protect VM contents.** Setting up regular automated backups is essential to protect against user errors. This isn't enough, though; you must also make sure that your backups are safe from cyberattacks and are available when you need them.
- **Monitor applications.** This pattern encompasses several tasks, including getting insight into the health of your VMs, understanding interactions among them, and establishing ways to monitor the applications these VMs run. All of these tasks are essential in keeping your applications running around the clock.
- **Secure and audit data access.** Organizations should audit all data access and use advanced machine learning capabilities to call out deviations from regular access patterns.
- **Failover practice.** Cloud operations that have low tolerances for failure must be capable of failing over or recovering from a cybersecurity or platform incident. These procedures must not simply be documented, but should be practiced quarterly.

Next steps

Now that you've reviewed the sample Security Baseline policy for cloud-native solutions, return to the policy review guide to start building on this sample to create your own policies for cloud adoption.

Build your own policies using the policy review guide

Microsoft security guidance

3/6/2021 • 5 minutes to read • Edit Online

Tools

The Microsoft Service Trust Portal and Compliance Manager can help meet these needs:

- Overcome compliance management challenges.
- Fulfill responsibilities of meeting regulatory requirements.
- Conduct self-service audits and risk assessments of enterprise cloud service utilization.

These tools are designed to help organizations meet complex compliance obligations and improve data protection capabilities when choosing and using Microsoft cloud services.

The **Microsoft Service Trust Portal** provides in-depth information and tools to help meet your needs for using Microsoft cloud services, including Azure, Microsoft 365, Dynamics 365, and Windows. The portal is a one-stop shop for security, regulatory, compliance, and privacy information related to the Microsoft cloud. It's where we publish the information and resources needed to perform self-service risk assessments of cloud services and tools. The portal was created to help track regulatory compliance activities within Azure, including:

- **Compliance Manager:** Compliance Manager, a workflow-based risk assessment tool in the Microsoft Service Trust Portal, enables you to track, assign, and verify your organization's regulatory compliance activities related to Microsoft cloud services, such as Microsoft 365, Dynamics 365, and Azure. You can find more details in the next section.
- **Trust documents:** Three categories of guides provide abundant resources to assess the Microsoft cloud, learn about Microsoft operations in security, compliance, and privacy, and help you act on improving your data protection capabilities. These guides include:
- **Audit reports:** Audit reports allow you to stay current on the latest privacy, security, and compliance-related information for Microsoft cloud services. This information includes ISO, SOC, FedRAMP, and other audit reports, bridge letters, and materials related to independent third-party audits of Microsoft cloud services such as Azure, Microsoft 365, Dynamics 365, and others.
- **Data protection guides:** Data protection guides provide information about how Microsoft cloud services protect your data, and how you can manage cloud data security and compliance for your organization. These guides include detailed white papers about the design and operation of Microsoft cloud services, FAQ documents, reports of end-of-year security assessments, penetration test results, and guidance to help you conduct risk assessment and improve your data protection capabilities.
- **Azure security and compliance blueprint:** Blueprints provide resources to assist you in building and launching cloud-powered applications that help you comply with stringent regulations and standards. With more certifications than any other cloud provider, you can have confidence deploying your critical workloads to Azure, with blueprints that include:
  - Industry-specific overview and guidance.
  - Customer responsibilities matrix.
  - Reference architectures with threat models.
  - Control implementation matrices.
  - Automation to deploy reference architectures.
  - Privacy resources. Documentation for data protection impact assessments, data subject requests, and data breach notification is provided to incorporate into your own accountability program in support of the General Data Protection Regulation (GDPR).
- **Get started with GDPR:** Microsoft products and services help organizations meet GDPR requirements

while collecting or processing personal data. The Microsoft Service Trust Portal is designed to give you information about the capabilities in Microsoft services that you can use to address specific requirements of the GDPR. The documentation can help your GDPR accountability and your understanding of technical and organizational measures. Documentation for data protection impact assessments, data subject requests, and data breach notification is provided to incorporate into your own accountability program in support of the GDPR.

  - **Data subject requests:** The GDPR grants individuals (or data subjects) certain rights in connection with the processing of their personal data. These rights include the right to correct inaccurate data, erase data, or restrict its processing, as well as the right to receive their data and fulfill a request to transmit their data to another controller.
  - **Data breach:** The GDPR mandates notification requirements for data controllers and processors if a breach of personal data occurs. The Microsoft Service Trust Portal provides you with information about how Microsoft works to prevent breaches, how Microsoft detects a breach, and how Microsoft will respond and notify you as a data controller if a breach occurs.
  - **Data protection impact assessment:** Microsoft helps controllers complete GDPR data protection impact assessments (DPIAs). The GDPR provides an inexhaustive list of cases in which DPIAs must be performed, such as automated processing for the purposes of profiling and similar activities; processing on a large scale of special categories of personal data, and systematic monitoring of a publicly accessible area on a large scale.
  - **Other resources:** In addition to tools guidance discussed in the above sections, the Microsoft Service Trust Portal also provides other resources including regional compliance, additional resources for the security and compliance center, and frequently asked questions about the Microsoft Service Trust Portal, Compliance Manager, privacy, and GDPR.
- **Regional compliance:** The Microsoft Service Trust Portal provides numerous compliance documents and guidance for Microsoft online services to meet compliance requirements for different regions including Czech Republic, Poland, and Romania.

Unique intelligent insights

As the volume and complexity of security signals grow, determining if those signals are credible threats, and then acting, takes far too long. Microsoft offers an unparalleled breadth of security intelligence delivered at cloud scale to help quickly detect and remediate threats. For more information, see the Azure Security Center overview.

Azure threat intelligence

By using the threat intelligence option available in Security Center, IT administrators can identify security threats against the environment. For example, they can identify whether a particular computer is part of a botnet. Computers can become nodes in a botnet when attackers illicitly install malware that secretly connects the computer to the command and control. Threat intelligence can also identify potential threats coming from underground communication channels, such as the dark web.

To build this threat intelligence, Security Center uses data from multiple sources within Microsoft. Security Center uses this data to identify potential threats against your environment. The threat intelligence pane is composed of three major options:

- Detected threat types
- Threat origin
- Threat intelligence map

Machine learning in Azure Security Center

Azure Security Center deeply analyzes a wealth of data from a variety of Microsoft and partner solutions to help

you achieve greater security. To take advantage of this data, the company use data science and machine learning for threat prevention, detection, and eventually investigation.

Broadly, Azure Machine Learning helps achieve two outcomes:

Next-generation detection

Attackers are increasingly automated and sophisticated. They use data science too. They reverse-engineer protections and build systems that support mutations in behavior. They masquerade their activities as noise, and learn quickly from mistakes. Machine learning helps us respond to these developments.

Simplified security baseline

Making effective security decisions is not easy. It requires security experience and expertise. While some large organizations have such experts on staff, many companies don't. Azure Machine Learning enables customers to benefit from the wisdom of other organizations when making security decisions.

Behavioral analytics

Behavioral analytics is a technique that analyzes and compares data to a collection of known patterns. These patterns are not simple signatures. They're determined through complex machine learning algorithms that are applied to massive data sets. They're also determined through careful analysis of malicious behaviors by expert analysts. Azure Security Center can use behavioral analytics to identify compromised resources based on analysis of virtual machine logs, virtual network device logs, fabric logs, crash dumps, and other sources.

Security Baseline tools in Azure

3/6/2021 • 2 minutes to read • Edit Online

The Security Baseline discipline is one of the Five Disciplines of Cloud Governance. This discipline focuses on ways of establishing policies that protect the network, assets, and most importantly the data that will reside on a cloud provider's solution. Within the Five Disciplines of Cloud Governance, the Security Baseline discipline involves classification of the digital estate and data. It also involves documentation of risks, business tolerance, and mitigation strategies associated with the security of data, assets, and networks. From a technical perspective, this discipline also includes involvement in decisions regarding encryption, network requirements, hybrid identity strategies, and tools to automate enforcement of security policies across resource groups.

The following list of Azure tools can help mature the policies and processes that support this discipline.

TOOL	AZURE PORTAL AND AZURE RESOURCE MANAGER	AZURE KEY VAULT	AZURE AD	AZURE POLICY	AZURE SECURITY CENTER	AZURE MONITOR
Apply access controls to resources and resource creation	Yes	No	Yes	No	No	No
Secure virtual networks	Yes	No	No	Yes	No	No
Encrypt virtual drives	No	Yes	No	No	No	No
Encrypt PaaS storage and databases	No	Yes	No	No	No	No
Manage hybrid identity services	No	No	Yes	No	No	No
Restrict allowed types of resource	No	No	No	Yes	No	No
Enforce geo-regional restrictions	No	No	No	Yes	No	No
Monitor security health of networks and resources	No	No	No	No	Yes	Yes

TOOL	AZURE PORTAL AND AZURE RESOURCE MANAGER	AZURE KEY VAULT	AZURE AD	AZURE POLICY	AZURE SECURITY CENTER	AZURE MONITOR
Detect malicious activity	No	No	No	No	Yes	Yes
Preemptively detect vulnerabilities	No	No	No	No	Yes	No
Configure backup and disaster recovery	Yes	No	No	No	No	No

For a complete list of Azure security tools and services, see Security services and technologies available on Azure.

Customers commonly use third-party tools to enable Security Baseline discipline activities. For more information, see the article integrate security solutions in Azure Security Center.

In addition to security tools, Compliance Management in Microsoft 365 provides extensive guidance, reports, and related documentation that can help you perform risk assessments as part of your migration planning process.

Identity Baseline discipline overview

3/6/2021 • 2 minutes to read • Edit Online

Identity baseline is one of the Five Disciplines of Cloud Governance within the Cloud Adoption Framework governance model. Identity is increasingly considered the primary security perimeter in the cloud, which is a shift from the traditional focus on network security. Identity services provide the core mechanisms supporting access control and organization within IT environments, and the Identity Baseline discipline complements the Security Baseline discipline by consistently applying authentication and authorization requirements across cloud adoption efforts.

> **NOTE**
> Identity Baseline discipline does not replace the existing IT teams, processes, and procedures that allow your organization to manage and secure identity services. The primary purpose of this discipline is to identify potential identity-related business risks and provide risk-mitigation guidance to IT staff that are responsible for implementing, maintaining, and operating your identity management infrastructure. As you develop governance policies and processes make sure to involve relevant IT teams in your planning and review processes.

This section of the Cloud Adoption Framework outlines the approach to developing an Identity Baseline discipline as part of your cloud governance strategy. The primary audience for this guidance is your organization's cloud architects and other members of your cloud governance team. The decisions, policies, and processes that emerge from this discipline should involve engagement and discussions with relevant members of the IT teams responsible for implementing and managing your organization's identity management solutions.

If your organization lacks in-house expertise in identity and security, consider engaging external consultants as a part of this discipline. Also consider engaging Microsoft Consulting Services, the Microsoft FastTrack cloud adoption service, or other external cloud adoption partners to discuss concerns related to this discipline.

Policy statements

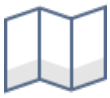
Actionable policy statements and the resulting architecture requirements serve as the foundation of an Identity Baseline discipline. Use sample policy statements as a starting point for defining your Identity Baseline policies.

Caution

The sample policies come from common customer experiences. To better align these policies to specific cloud governance needs, execute the following steps to create policy statements that meet your unique business needs.

Develop governance policy statements

The following steps offer examples and potential options to consider when developing your Identity Baseline discipline. Use each step as a starting point for discussions within your cloud governance team and with affected business, and IT teams across your organization to establish the policies and processes needed to manage identity-related risks.

	Identity Baseline discipline template: Download the template for documenting an Identity Baseline discipline.
	Business risks: Understand the motives and risks commonly associated with the Identity Baseline discipline.
	Indicators and metrics: Indicators to understand whether it's the right time to invest in the Identity Baseline discipline.
	Policy adherence processes: Suggested processes for supporting policy compliance in the Identity Baseline discipline.
	Maturity: Align cloud management maturity with phases of cloud adoption.
	Toolchain: Azure services that can be implemented to support the Identity Baseline discipline.

Next steps

Get started by evaluating business risks in a specific environment.

Understand business risks

Identity Baseline discipline template

3/6/2021 • 2 minutes to read • Edit Online

The first step to implementing change is communicating the desired change. The same is true when changing governance practices. The following template serves as a starting point for documenting and communicating policy statements that govern identity services in the cloud.

As your discussions progress, use this template's structure as a model for capturing the business risks, risk tolerances, compliance processes, and tooling needed to define your organization's Identity Baseline policy statements.

> **IMPORTANT**
> This template is a limited sample. Before updating this template to reflect your requirements, you should review the subsequent steps for defining an effective Identity Baseline discipline within your cloud governance strategy.

Download the Identity Baseline discipline template

Next steps

Solid governance practices start with an understanding of business risk. Review the article on business risks and begin to document the business risks that align with your current cloud adoption plan.

Understand business risks

Motivations and business risks in the Identity Baseline discipline

3/6/2021 • 2 minutes to read • Edit Online

This article discusses the reasons that customers typically adopt an Identity Baseline discipline within a cloud governance strategy. It also provides a few examples of business risks that drive policy statements.

Relevance

Traditional on-premises directories are designed to allow businesses to strictly control permissions and policies for users, groups, and roles within their internal networks and datacenters. These directories typically support single-tenant implementations, with services applicable only within the on-premises environment.

Cloud identity services expand an organization's authentication and access control capabilities to the internet. They support multitenancy and can be used to manage users and access policy across cloud applications and deployments. Public cloud platforms have cloud-native identity services supporting management and deployment tasks and are capable of varying levels of integration with your existing on-premises identity solutions. All of these features can result in cloud identity policy being more complicated than your traditional on-premises solutions require.

The importance of the Identity Baseline discipline to your cloud deployment will depend on the size of your team and need to integrate your cloud-based identity solution with an existing on-premises identity service. Initial test deployments may not require much in the way of user organization or management, but as your cloud estate matures, you will likely need to support more complicated organizational integration and centralized management.

Business risk

The Identity Baseline discipline attempts to address core business risks related to identity services and access control. Work with your business to identify these risks and monitor each of them for relevance as you plan for and implement your cloud deployments.

Risks will differ between organization, but the following serve as common identity-related risks that you can use as a starting point for discussions within your cloud governance team:

- **Unauthorized access.** Sensitive data and resources that can be accessed by unauthorized users can lead to data leaks or service disruptions, violating your organization's security perimeter and risking business or legal liabilities.
- **Inefficiency due to multiple identity solutions.** Organizations with multiple identity services tenants can require multiple accounts for users. This can lead to inefficiency for users who need to remember multiple sets of credentials and for IT in managing accounts across multiple systems. If user access assignments are not updated across identity solutions as staff, teams, and business goals change, your cloud resources may be vulnerable to unauthorized access or users unable to access required resources.
- **Inability to share resources with external partners.** Difficulty adding external business partners to your existing identity solutions can prevent efficient resource sharing and business communication.
- **On-premises identity dependencies.** Legacy authentication mechanisms or third-party multi-factor authentication might not be available in the cloud, requiring either migrating workloads to be retooled, or additional identity services to be deployed to the cloud. Either requirement could delay or prevent migration, and increase costs.

Next steps

Use the Identity Baseline discipline template to document business risks that are likely to be introduced by the current cloud adoption plan.

Once an understanding of realistic business risks is established, the next step is to document the business's tolerance for risk and the indicators and key metrics to monitor that tolerance.

Understand indicators, metrics, and risk tolerance

Identity baseline metrics, indicators, and risk tolerance

3/6/2021 • 4 minutes to read • Edit Online

Learn to quantify business risk tolerance associated with the Identity Baseline discipline. Defining metrics and indicators helps to create a business case for investing in the maturity of this discipline.

Metrics

Identity management focuses on identifying, authenticating, and authorizing individuals, groups of users, or automated processes, and providing them appropriate access to resources in your cloud deployments. As part of your risk analysis you'll want to gather data related to your identity services to determine how much risk you face, and how important investment in your Identity Baseline discipline is for your planned cloud deployments.

The following are examples of useful metrics that you should gather to help evaluate risk tolerance within the Identity Baseline discipline:

- **Identity systems size.** Total number of users, groups, or other objects managed through your identity systems.
- **Overall size of directory services infrastructure.** Number of directory forests, domains, and tenants used by your organization.
- **Dependency on legacy or on-premises authentication mechanisms.** Number of workloads that depend on legacy or third-party or multi-factor authentication mechanisms.
- **Extent of cloud-deployed directory services.** Number of directory forests, domains, and tenants you've deployed to the cloud.
- **Cloud-deployed Active Directory servers.** Number of Active Directory servers deployed to the cloud.
- **Cloud-deployed organizational units.** Number of Active Directory organizational units (OUs) deployed to the cloud.
- **Extent of federation.** Number of identity management systems federated with your organization's systems.
- **Elevated users.** Number of user accounts with elevated access to resources or management tools.
- **Use of Azure role-based access control.** Number of subscriptions, resource groups, or individual resources not managed through Azure role-based access control (Azure RBAC) via groups.
- **Authentication claims.** Number of successful and failed user authentication attempts.
- **Authorization claims.** Number of successful and failed attempts by users to access resources.
- **Compromised accounts.** Number of user accounts that have been compromised.

Risk tolerance indicators

Risks related to identity baseline are largely related to the complexity of your organization's identity infrastructure. If all your users and groups are managed using a single directory or cloud-native identity provider using minimal integration with other services, your risk level will likely be small. As your business needs grow, your identity management systems may need to support more complicated scenarios, such as multiple directories to support your internal organization or federation with external identity providers. As these systems become more complex, risk increases.

In the early stages of cloud adoption, work with your IT security team and business stakeholders to identify business risks related to identity, then determine an acceptable baseline for identity risk tolerance. This section

of the Cloud Adoption Framework provides examples, but the detailed risks and baselines for your company or deployments may be different.

Once you have a baseline, establish minimum benchmarks representing an unacceptable increase in your identified risks. These benchmarks act as triggers for when you need to take action to address these risks. The following are a few examples of how identity related metrics, such as those discussed above, can justify an increased investment in the Identity Baseline discipline.

- **User account number trigger.** A company with more than *x* users, groups, or other objects managed in your identity systems could benefit from investment in the Identity Baseline discipline to ensure efficient governance over a large number of accounts.
- **On-premises identity dependency trigger.** A company planning to migrate workloads to the cloud that require legacy authentication capabilities or third-party multi-factor authentication should invest in the Identity Baseline discipline to reduce risks related to refactoring or additional cloud infrastructure deployment.
- **Directory services complexity trigger.** A company maintaining more than *x* individual forests, domains, or directory tenants should invest in the Identity Baseline discipline to reduce risks related with account management and the efficiency issues related to multiple user credentials spread across multiple systems.
- **Cloud-hosted directory services trigger.** A company hosting *x* Active Directory server virtual machines (VMs) hosted in the cloud, or having *x* organizational units (OUs) managed on these cloud-based servers, can benefit from investment in the Identity Baseline discipline to optimize integration with any on-premises or other external identity services.
- **Federation trigger.** A company implementing identity federation with *x* external identity management systems can benefit from investing in the Identity Baseline discipline to ensure consistent organizational policy across federation members.
- **Elevated access trigger.** A company with more than *x%* of users with elevated permissions to management tools and resources should consider investing in the Identity Baseline discipline to minimize the risk of inadvertent overprovisioning of access to users.
- **Azure RBAC trigger.** A company with less than *x%* of resources using Azure role-based access control methods should consider investing in the Identity Baseline discipline to identify optimized ways to assign user access to resources.
- **Authentication failure trigger.** A company where authentication failures represent more than *x%* of attempts should invest in the Identity Baseline discipline to ensure that authentication methods are not under external attack, and that users can authenticate properly.
- **Authorization failure trigger.** A company where access attempts are rejected more than *x%* of the time should invest in the Identity Baseline discipline to improve the application and updating of access controls, and identify potentially malicious access attempts.
- **Compromised account trigger.** A company with more than 1 compromised account should invest in the Identity Baseline discipline to improve the strength and security of authentication mechanisms and improve mechanisms to remediate risks related to compromised accounts.

The exact metrics and triggers you use to gauge risk tolerance and the level of investment in the Identity Baseline discipline will be specific to your organization, but the examples above should serve as a useful base for discussion within your cloud governance team.

Next steps

Use the Identity Baseline discipline template to document metrics and tolerance indicators that align to the current cloud adoption plan.

Review sample Identity Baseline policies as a starting point to develop your own policies to address specific business risks aligned with your cloud adoption plans.

Review sample policies

Identity Baseline sample policy statements

3/6/2021 • 3 minutes to read • Edit Online

Individual cloud policy statements are guidelines for addressing specific risks identified during your risk assessment process. These statements should provide a concise summary of risks and plans to deal with them. Each statement definition should include these pieces of information:

- **Technical risk:** A summary of the risk this policy will address.
- **Policy statement:** A clear summary explanation of the policy requirements.
- **Design options:** Actionable recommendations, specifications, or other guidance that IT teams and developers can use when implementing the policy.

The following sample policy statements address common identity-related business risks. These statements are examples you can reference when drafting policy statements to address your organization's needs. These examples are not meant to be proscriptive, and there are potentially several policy options for dealing with each identified risk. Work closely with business and IT teams to identify the best policies for your unique set of risks.

Lack of access controls

Technical risk: Insufficient or ad hoc access control settings can introduce risk of unauthorized access to sensitive or mission-critical resources.

Policy statement: All assets deployed to the cloud should be controlled using identities and roles approved by current governance policies.

Potential design options: Azure Active Directory Conditional Access is the default access control mechanism in Azure.

Overprovisioned access

Technical risk: Users and groups with control over resources beyond their area of responsibility can result in unauthorized modifications leading to outages or security vulnerabilities.

Policy statement: The following policies will be implemented:

- A least-privilege access model will be applied to any resources involved in mission-critical applications or protected data.
- Elevated permissions should be an exception, and any such exceptions must be recorded with the cloud governance team. Exceptions will be audited regularly.

Potential design options: Consult the Azure identity management best practices to implement an Azure role-based access control (Azure RBAC) strategy that restricts access based on the need to know and least-privilege security principles.

Lack of shared management accounts between on-premises and the cloud

Technical risk: IT management or administrative staff with accounts on your on-premises Active Directory may not have sufficient access to cloud resources might not be able to efficiently resolve operational or security issues.

Policy statement: All groups in the on-premises Active Directory infrastructure that have elevated privileges

should be mapped to an approved Azure role.

Potential design options: Implement a hybrid identity solution between your cloud-based Azure Active Directory and your on-premises Active Directory, and add the required on-premises groups to the Azure roles necessary to do their work.

Weak authentication mechanisms

Technical risk: Identity management systems with insufficiently secure user authentication methods, such as basic user/password combinations, can lead to compromised or hacked passwords, providing a major risk of unauthorized access to secure cloud systems.

Policy statement: All accounts are required to sign in to secured resources using a multi-factor authentication method.

Potential design options: For Azure Active Directory, implement Azure Multi-Factor Authentication as part of your user authorization process.

Isolated identity providers

Technical risk: Incompatible identity providers can result in the inability to share resources or services with customers or other business partners.

Policy statement: Deployment of any applications that require customer authentication must use an approved identity provider that is compatible with the primary identity provider for internal users.

Potential design options: Implement federation with Azure Active Directory between your internal and customer identity providers or use Azure Active Directory B2B

Identity reviews

Technical risk: As business changes over time, the addition of new cloud deployments or other security concerns can increase the risks of unauthorized access to secure resources.

Policy statement: Cloud governance processes must include quarterly review with identity management teams to identify malicious actors or usage patterns that should be prevented by cloud asset configuration.

Potential design options: Establish a quarterly security review meeting that includes both governance team members and IT staff responsible for managing identity services. Review existing security data and metrics to establish gaps in current identity management policy and tooling, and update policy to remediate any new risks.

Next steps

Use the samples mentioned in this article as a starting point for developing policies to address specific business risks that align with your cloud adoption plans.

To begin developing your own custom Identity Baseline policy statements, download the Identity Baseline discipline template.

To accelerate adoption of this discipline, choose the actionable governance guide that most closely aligns with your environment. Then modify the design to incorporate your specific corporate policy decisions.

Building on risks and tolerance, establish a process for governing and communicating Identity Baseline policy adherence.

Establish policy compliance processes

Identity Baseline policy compliance processes

3/6/2021 • 4 minutes to read • Edit Online

This article discusses an approach to policy adherence processes that govern the Identity Baseline discipline. Effective governance of identity starts with recurring manual processes that guide identity policy adoption and revisions. This requires regular involvement of the cloud governance team and interested business and IT stakeholders to review and update policy and ensure policy compliance. In addition, many ongoing monitoring and enforcement processes can be automated or supplemented with tooling to reduce the overhead of governance and allow for faster response to policy deviation.

Planning, review, and reporting processes

Identity management tools offer capabilities and features that greatly assist user management and access control within a cloud deployment. They also require well-considered processes and policies to support your organization's goals. The following is a set of example processes commonly involved in the Identity Baseline discipline. Use these examples as a starting point when planning the processes that will allow you to continue to update identity policy based on business change and feedback from the IT teams tasked with turning governance guidance into action.

Initial risk assessment and planning: As part of your initial adoption of the Identity Baseline discipline, identify your core business risks and tolerances related to cloud identity management. Use this information to discuss specific technical risks with members of your IT teams responsible for managing identity services and develop a baseline set of security policies for mitigating these risks to establish your initial governance strategy.

Deployment planning: Before any deployment, review the access needs for any workloads and develop an access control strategy that aligns with established corporate identity policy. Document any gaps between needs and current policy to determine whether policy updates are required, and modify policy as needed.

Deployment testing: As part of the deployment, the cloud governance team, in cooperation with IT teams responsible for identity services, will be responsible for reviewing the deployment to validate identity policy compliance.

Annual planning: On an annual basis, perform a high-level review of identity management strategy. Explore planned changes to the identity services environment and updated cloud adoption strategies to identify potential risk increase or need to modify current identity infrastructure patterns. Also use this time to review the latest identity management best practices and integrate these into your policies and review processes.

Quarterly planning: On a quarterly basis perform a general review of identity and access control audit data, and meet with the cloud adoption teams to identify any potential new risks or operational requirements that would require updates to identity policy or changes in access control strategy.

This planning process is also a good time to evaluate the current membership of your cloud governance team for knowledge gaps related to new or changing policy and risks related to identity. Invite relevant IT staff to participate in reviews and planning as either temporary technical advisors or permanent members of your team.

Education and training: On a bimonthly basis, offer training sessions to make sure IT staff and developers are up-to-date on the latest identity policy requirements. As part of this process review and update any documentation, guidance, or other training assets to ensure they're in sync with the latest corporate policy statements.

Monthly audit and reporting reviews: On a monthly basis, perform an audit on all cloud deployments to

assure their continued alignment with identity policy. Use this review to check user access against business change to ensure users have correct access to cloud resources, and ensure access strategies such as Azure RBAC are being followed consistently. Identify any privileged accounts and document their purpose. This review process produces a report for the cloud strategy team and each cloud adoption team detailing overall adherence to policy. The report is also stored for auditing and legal purposes.

Processes for ongoing monitoring

A successful Identity Baseline strategy depends on visibility into the current and past state of your identity systems. Without the ability to analyze your cloud deployment's relevant metrics and related data, you cannot identify changes in your risks or detect violations of your risk tolerances. The ongoing governance processes discussed above require quality data to ensure policy can be modified to support the changing needs of your business.

Ensure that your IT teams have implemented automated monitoring systems for your identity services that capture the logs and audit information you need to evaluate risk. Be proactive in monitoring these systems to ensure prompt detection and mitigation of potential policy violation, and ensure any changes to your identity infrastructure are reflected in your monitoring strategy.

Violation triggers and enforcement actions

Violations of identity policy can result in unauthorized access to sensitive data and lead to serious disruption of mission-critical application and services. When violations are detected, you should take actions to realign with policy as soon as possible. Your IT team can automate most violation triggers using the tools outlined in the Identity Baseline toolchain.

The following triggers and enforcement actions provide examples you can reference when planning how to use monitoring data to resolve policy violations:

- **Suspicious activity detected:** User logins detected from anonymous proxy IP addresses, unfamiliar locations, or successive logins from impossibly distant geographical locations may indicate a potential account breach or malicious access attempt. Login will be blocked until user identity can be verified and password reset.
- **Leaked user credentials:** Accounts that have their username and password leaked to the internet will be disabled until user identity can be verified and password reset.
- **Insufficient access controls detected:** Any protected assets where access restrictions do not meet security requirements will have access blocked until the resource is brought into compliance.

Next steps

Use the Identity Baseline discipline template to document the processes and triggers that align to the current cloud adoption plan.

For guidance on executing cloud management policies in alignment with adoption plans, see the article on discipline improvement.

Identity Baseline discipline improvement

Identity Baseline discipline improvement

3/6/2021 • 6 minutes to read • Edit Online

The Identity Baseline discipline focuses on ways of establishing policies that ensure consistency and continuity of user identities regardless of the cloud provider that hosts the application or workload. Within the Five Disciplines of Cloud Governance, the Identity Baseline discipline includes decisions regarding the hybrid identity strategy, evaluation and extension of identity repositories, implementation of single sign-on (same sign-on), auditing and monitoring for unauthorized use or malicious actors. In some cases, it may also involve decisions to modernize, consolidate, or integrate multiple identity providers.

This article outlines some potential tasks your company can engage in to better develop and mature the Identity Baseline discipline. These tasks can be broken down into planning, building, adopting, and operating phases of implementing a cloud solution, which are then iterated on allowing the development of an incremental approach to cloud governance.

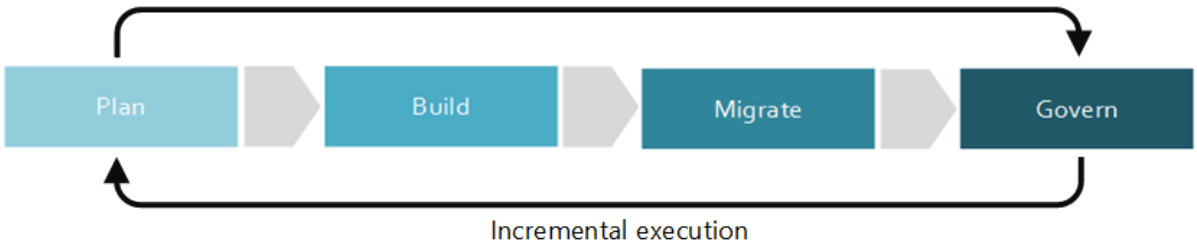


Figure 1: Phases of an incremental approach to cloud governance.

It's impossible for any one document to account for the requirements of all businesses. As such, this article outlines suggested minimum and potential example activities for each phase of the governance maturation process. The initial objective of these activities is to help you build a policy MVP and establish a framework for incremental policy improvement. Your cloud governance team will need to decide how much to invest in these activities to improve your Identity Baseline discipline.

Caution

Neither the minimum or potential activities outlined in this article are aligned to specific corporate policies or third-party compliance requirements. This guidance is designed to help facilitate the conversations that will lead to alignment of both requirements with a cloud governance model.

Planning and readiness

This phase of governance maturity bridges the divide between business outcomes and actionable strategies. During this process, the leadership team defines specific metrics, maps those metrics to the digital estate, and begins planning the overall migration effort.

Minimum suggested activities:

- Evaluate your Identity Baseline toolchain options and implement a hybrid strategy that is appropriate to your organization.
- Develop a draft architecture guidelines document and distribute to key stakeholders.
- Educate and involve the people and teams affected by the development of architecture guidelines.

Potential activities:

- Define roles and assignments that will govern identity and access management in the cloud.
- Define your on-premises groups and map to corresponding cloud-based roles.

- Inventory identity providers (including database-driven identities used by custom applications).
- Consolidate and integrate identity providers where duplication exists, to simplify the overall identity solution and reduce risk.
- Evaluate hybrid compatibility of existing identity providers.
- For identity providers that are not hybrid compatible, evaluate consolidation or replacement options.

Build and predeployment

Several technical and nontechnical prerequisites are required to successfully migrate an environment. This process focuses on the decisions, readiness, and core infrastructure that proceeds a migration.

Minimum suggested activities:

- Consider a pilot test before implementing your Identity Baseline toolchain, making sure it simplifies the user experience as much as possible.
- Apply feedback from pilot tests into the predeployment. Repeat until results are acceptable.
- Update the architecture guidelines document to include deployment and user adoption plans, and distribute to key stakeholders.
- Consider establishing an early adopter program and rolling out to a limited number of users.
- Continue to educate the people and teams most affected by the architecture guidelines.

Potential activities:

- Evaluate your logical and physical architecture and determine a hybrid identity strategy.
- Map identity access management policies, such as login ID assignments, and choose the appropriate authentication method for Azure AD.
  - If federated, enable tenant restrictions for administrative accounts.
- Integrate your on-premises and cloud directories.
- Consider using the following access models:
  - Least-privilege administrative access model.
  - Privileged Identity Management access model.
- Finalize all preintegration details and review identity management and access control security best practices.
  - Enable single-identity single-sign-on (SSO), also called seamless SSO.
  - Configure multi-factor authentication for administrators.
  - Consolidate or integrate identity providers, where necessary.
  - Implement tooling necessary to centralize management of identities.
  - Enable just-in-time (JIT) access and role change alerting.
  - Conduct a risk analysis of key admin activities for assignment to built-in roles.
  - Consider an updated rollout of stronger authentication for all users.
  - Enable Privileged Identity Management (PIM) for JIT (using time-limited activation) for additional administrative roles.
  - Separate user accounts from global admin accounts, to ensure that administrators do not inadvertently open emails or run programs associated with their global admin accounts).

Adopt and migrate

Migration is an incremental process that focuses on the movement, testing, and adoption of applications or workloads in an existing digital estate.

Minimum suggested activities:

- Migrate your Identity Baseline toolchain from development to production.

- Update the architecture guidelines document and distribute to key stakeholders.
- Develop educational materials and documentation, awareness communications, incentives, and other programs to help drive user adoption.

Potential activities:

- Validate that the best practices defined during the build predeployment phases are properly executed.
- Validate and refine your hybrid identity strategy.
- Ensure that each application or workload continues to align with the identity strategy before release.
- Validate that single sign-on (SSO) and seamless SSO is working as expected for your applications.
- Reduce or eliminate the number of alternative identity stores.
- Scrutinize the need for any in-app or in-database identity stores. Identities that fall outside of a proper identity provider (first-party or third-party) can represent risk to the application and the users.
- Enable conditional access for on-premises federated applications.
- Distribute identity across global regions in multiple hubs with synchronization between regions.
- Establish central Azure role-based access control (Azure RBAC) federation.

Operate and post-implementation

Once the transformation is complete, governance and operations must live on for the natural lifecycle of an application or workload. This phase of governance maturity focuses on the activities that commonly come after the solution is implemented and the transformation cycle begins to stabilize.

Minimum suggested activities:

- Customize your Identity Baseline toolchain based on your organization's changing needs.
- Automate notifications and reports to alert you of potential malicious threats.
- Monitor and report on system usage and user adoption progress.
- Report on post-deployment metrics and distribute to stakeholders.
- Refine the architecture guidelines to guide future adoption processes.
- Communicate and continually educate the affected teams on a periodic basis to ensure ongoing adherence to architecture guidelines.

Potential activities:

- Conduct periodic audits of identity policies and adherence practices.
- Ensure sensitive user accounts (such as accounts of corporate executives) are always enabled for multi-factor authentication and anomalous login detection.
- Scan for malicious actors and data breaches regularly, particularly those related to identity fraud, such as potential admin account takeovers.
- Configure a monitoring and reporting tool.
- Consider integrating more closely with security and fraud-prevention systems.
- Regularly review access rights for elevated users or roles.
  - Identify every user who is eligible to activate admin privilege.
- Review onboarding, offboarding, and credential update processes.
- Investigate increasing levels of automation and communication between identity access management (IAM) modules.
- Consider implementing a development security operations (DevSecOps) approach.
- Conduct an impact analysis to gauge results on costs, security, and user adoption.
- Periodically produce an impact report that shows the changes in metrics created by the system and estimate the business impacts of the hybrid identity strategy.

- Establish integrated monitoring recommended by the Azure Security Center.

Next steps

Now that you understand the concept of cloud identity governance, examine the Identity Baseline toolchain to identify Azure tools and features that you'll need when developing the Identity Baseline discipline on the Azure platform.

Identity Baseline toolchain for Azure

Identity Baseline tools in Azure

7/14/2021 • 4 minutes to read • Edit Online

The Identity Baseline discipline is one of the Five Disciplines of Cloud Governance. This discipline focuses on ways of establishing policies that ensure consistency and continuity of user identities regardless of the cloud provider that hosts the application or workload.

The following tools are included in the discovery guide for hybrid identity.

Active Directory (on-premises): Active Directory is the identity provider most frequently used in the enterprise to store and validate user credentials.

Azure Active Directory: A software as a service (SaaS) equivalent to Active Directory, capable of federating with an on-premises Active Directory.

Active Directory (IaaS): An instance of the Active Directory application running in a virtual machine in Azure.

Identity is the control plane for IT security. So authentication is an organization's access guard to the cloud. Organizations need an identity control plane that strengthens their security and keeps their cloud applications safe from intruders.

Cloud authentication

Choosing the correct authentication method is the first concern for organizations wanting to move their applications to the cloud.

When you choose this method, Azure AD handles users' sign-in process. Coupled with seamless single sign-on (SSO), users can sign in to cloud applications without having to reenter their credentials. With cloud authentication, you can choose from two options:

Azure AD password hash synchronization: The simplest way to enable authentication for on-premises directory objects in Azure AD. This method can also be used with any method as a back-up failover authentication method in case your on-premises server goes down.

Azure AD Pass-through Authentication: Provides a persistent password validation for Azure AD authentication services by using a software agent that runs on one or more on-premises servers.

> **NOTE**
> Companies with a security requirement to immediately enforce on-premises user account states, password policies, and sign-in hours should consider the pass-through authentication method.

Federated authentication:

When you choose this method, Azure AD passes the authentication process to a separate trusted authentication system, such as on-premises Active Directory Federation Services (AD FS) or a trusted third-party federation provider, to validate the user's password.

For a decision tree that helps you choose the best solution for your organization, see Choose the right authentication method for Azure Active Directory.

The following table lists the native tools that can help mature the policies and processes that support this discipline.

CONSIDERATION	PASSWORD HASH SYNCHRONIZATION + SEAMLESS SSO	PASS-THROUGH AUTHENTICATION + SEAMLESS SSO	FEDERATION WITH AD FS
Where does authentication happen?	In the cloud	In the cloud after a secure password verification exchange with the on-premises authentication agent	On-premises
What are the on-premises server requirements beyond the provisioning system: Azure AD Connect?	None	One server for each additional authentication agent	Two or more AD FS servers Two or more WAP servers in the perimeter network
What are the requirements for on-premises internet and networking beyond the provisioning system?	None	Outbound internet access from the servers running authentication agents	Inbound internet access to WAP servers in the perimeter Inbound network access to AD FS servers from WAP servers in the perimeter Network load balancing
Is there an SSL certificate requirement?	No	No	Yes
Is there a health monitoring solution?	Not required	Agent status provided by Azure Active Directory admin center	Azure AD Connect Health
Do users get single sign-on to cloud resources from domain-joined devices within the company network?	Yes with Seamless SSO	Yes with Seamless SSO	Yes
What sign-in types are supported?	UserPrincipalName + password Integrated Windows authentication by using Seamless SSO Alternate login ID	UserPrincipalName + password Integrated Windows authentication by using Seamless SSO Alternate login ID	UserPrincipalName + password SamAccountName + password Integrated Windows authentication Certificate and smart card authentication Alternate login ID
Is Windows Hello for Business supported?	Key trust model Certificate trust model with Intune	Key trust model Certificate trust model with Intune	Key trust model Certificate trust model

CONSIDERATION	PASSWORD HASH SYNCHRONIZATION + SEAMLESS SSO	PASS-THROUGH AUTHENTICATION + SEAMLESS SSO	FEDERATION WITH AD FS
What are the multi-factor authentication options?	Azure Multi-Factor Authentication Custom controls with Azure AD Conditional Access*	Azure Multi-Factor Authentication Custom controls with Azure AD Conditional Access*	Azure Multi-Factor Authentication Azure Multi-Factor Authentication server Third-party multi-factor authentication Custom controls with Azure AD access
What user account states are supported?	Disabled accounts (Up to 30-minute delay)	Disabled accounts Account locked out Account expired Password expired Sign-in hours	Disabled accounts Account locked out Account expired Password expired Sign-in hours
What are the conditional access options in Azure AD?	Azure AD Conditional Access	Azure AD Conditional Access	Azure AD Conditional Access AD FS claim rules
Is blocking legacy protocols supported?	Yes	Yes	Yes
Can you customize the logo, image, and description on the sign-in pages?	Yes, with Azure AD Premium	Yes, with Azure AD Premium	Yes
What advanced scenarios are supported?	Smart password lockout Leaked credentials reports	Smart password lockout	Multisite low-latency authentication system AD FS extranet lockout Integration with third-party identity systems

NOTE

Custom controls in Azure AD Conditional Access does not currently support device registration.

Next steps

The Hybrid Identity Digital Transformation Framework white paper outlines combinations and solutions for choosing and integrating each of these components.

The Azure AD Connect tool helps you to integrate your on-premises directories with Azure AD.

Resource Consistency discipline overview

3/6/2021 • 2 minutes to read • Edit Online

Resource consistency is one of the Five Disciplines of Cloud Governance within the Cloud Adoption Framework governance model. This discipline focuses on ways of establishing policies related to the operational management of an environment, application, or workload. IT operations teams often provide monitoring of applications, workload, and asset performance. They also commonly execute the tasks required to meet scale demands, remediate performance service-level agreement (SLA) violations, and proactively avoid performance SLA violations through automated remediation. Within the Five Disciplines of Cloud Governance, the Resource Consistency discipline ensures resources are consistently configured in such a way that they can be discoverable by IT operations, are included in recovery solutions, and can be onboarded into repeatable operations processes.

> **NOTE**
>
> Resource Consistency discipline does not replace the existing IT teams, processes, and procedures that allow your organization to effectively manage cloud-based resources. The primary purpose of this discipline is to identify potential business risks and provide risk-mitigation guidance to the IT staff that are responsible for managing your resources in the cloud. As you develop governance policies and processes make sure to involve relevant IT teams in your planning and review processes.

This section of the Cloud Adoption Framework outlines how to develop a Resource Consistency discipline as part of your cloud governance strategy. The primary audience for this guidance is your organization's cloud architects and other members of your cloud governance team. The decisions, policies, and processes that emerge from this discipline should involve engagement and discussions with relevant members of the IT teams responsible for implementing and managing your organization's resource consistency solutions.

If your organization lacks in-house expertise in resource consistency strategies, consider engaging external consultants as a part of this discipline. Also consider engaging Microsoft Consulting Services, the Microsoft FastTrack cloud adoption service, or other external cloud adoption experts for discussing how best to organize, track, and optimize your cloud-based assets.

Policy statements

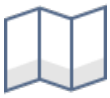
Actionable policy statements and the resulting architecture requirements serve as the foundation of a Resource Consistency discipline. Use sample policy statements. These samples can serve as a starting point for defining your Resource Consistency policies.

Caution

The sample policies come from common customer experiences. To better align these policies to specific cloud governance needs, execute the following steps to create policy statements that meet your unique business needs.

Develop governance policy statements

The following steps offer examples and potential options to consider when developing your Resource Consistency discipline. Use each step as a starting point for discussions within your cloud governance team and with affected business, and IT teams across your organization to establish the policies and processes needed to manage Resource Consistency discipline risks.

	Resource Consistency discipline template: Download the template for documenting a Resource Consistency discipline.
	Business risks: Understand the motives and risks commonly associated with the Resource Consistency discipline.
	Indicators and metrics: Indicators to understand whether it's the right time to invest in the Resource Consistency discipline.
	Policy adherence processes: Suggested processes for supporting policy compliance in the Resource Consistency discipline.
	Maturity: Align cloud management maturity with phases of cloud adoption.
	Toolchain: Azure services that can be implemented to support the Resource Consistency discipline.

Next steps

Get started by evaluating business risks in a specific environment.

Understand business risks

Resource Consistency discipline template

3/6/2021 • 2 minutes to read • Edit Online

The first step to implementing change is communicating what is desired. The same is true when changing governance practices. The following template serves as a starting point for documenting and communicating policy statements that govern IT operations and management in the cloud.

As your discussions progress, use this template's structure as a model for capturing the business risks, risk tolerances, compliance processes, and tooling needed to define your organization's Resource Consistency policy statements.

> **IMPORTANT**
> This template is a limited sample. Before updating this template to reflect your requirements, you should review the subsequent steps for defining an effective Resource Consistency discipline within your cloud governance strategy.

Download the Resource Consistency discipline template

Next steps

Solid governance practices start with an understanding of business risk. Review the article on business risks and begin to document the business risks that align with your current cloud adoption plan.

Understand business risks

Motivations and business risks in the Resource Consistency discipline

3/6/2021 • 2 minutes to read • Edit Online

This article discusses the reasons that customers typically adopt a Resource Consistency discipline within a cloud governance strategy. It also provides a few examples of potential business risks that can drive policy statements.

Relevance

When it comes to deploying resources and workloads, the cloud offers increased agility and flexibility over most traditional on-premises datacenters. These potential cloud-based advantages also come with potential management drawbacks that can seriously jeopardize the success of your cloud adoption. What assets have you deployed? What teams own what assets? Do you have enough resources supporting a workload? How do you know whether workloads are healthy?

Resource consistency is crucial to ensure that resources are deployed, updated, and configured consistently in a repeatable manner, and that service disruptions are minimized and remedied in as little time as possible.

The Resource Consistency discipline is concerned with identifying and mitigating business risks related to the operational aspects of your cloud deployment. Resource consistency includes monitoring of applications, workloads, and asset performance. It also includes the tasks required to meet scale demands, provide disaster recovery capabilities, mitigate performance service-level agreement (SLA) violations, and proactively avoid those SLA violations through automated remediation.

Initial test deployments may not require much beyond adopting some cursory naming and tagging standards to support your resource consistency needs. As your cloud adoption matures and you deploy more complicated and mission-critical assets, the need to invest in the Resource Consistency discipline increases rapidly.

Business risk

The Resource Consistency discipline attempts to address core operational business risks. Work with your business and IT teams to identify these risks and monitor each of them for relevance as you plan for and implement your cloud deployments.

Risks will differ between organization, but the following serve as common risks that you can use as a starting point for discussions within your cloud governance team:

- **Unnecessary operational cost.** Obsolete or unused resources, or resources that are overprovisioned during times of low demand, add unnecessary operational costs.
- **Underprovisioned resources.** Resources that experience higher than anticipated demand can result in business disruption as cloud resources are overwhelmed by demand.
- **Management inefficiencies.** Lack of consistent naming and tagging metadata associated with resources can lead to IT staff having difficulty finding resources for management tasks or identifying ownership and accounting information related to assets. This results in management inefficiencies that can increase cost and slow IT responsiveness to service disruption or other operational issues.
- **Business interruption.** Service disruptions that result in violations of your organization's established service-level agreements (SLAs) can result in loss of business or other financial impacts to your company.

Next steps

Use the Resource Consistency discipline template to document business risks that are likely to be introduced by the current cloud adoption plan.

Once an understanding of realistic business risks is established, the next step is to document the business's tolerance for risk and the indicators and key metrics to monitor that tolerance.

Understand indicators, metrics, and risk tolerance

Risk tolerance metrics and indicators in the Resource Consistency discipline

3/6/2021 • 5 minutes to read • Edit Online

Learn to quantify business risk tolerance associated with the Resource Consistency discipline. Defining metrics and indicators helps to create a business case for investing in the maturity of this discipline.

Metrics

Resource consistency focuses on addressing risks related to the operational management of your cloud deployments. As part of your risk analysis you'll want to gather data related to your IT operations to determine how much risk you face, and how important investment in your Resource Consistency discipline is for your planned cloud deployments.

Every organization has different operational scenarios, but the following items represent useful examples of the metrics you should gather when evaluating risk tolerance within the Resource Consistency discipline:

- **Cloud assets.** Total number of cloud-deployed resources.
- **Untagged resources.** Number of resources without required accounting, business impact, or organizational tags.
- **Underused assets.** Number of resources where memory, CPU, or network capabilities are all consistently underutilized.
- **Resource depletion.** Number of resources where memory, CPU, or network capabilities are exhausted by load.
- **Resource age.** Time since resource was last deployed or modified.
- **VMs in critical condition.** Number of deployed VMs where one or more critical issues are detected that must addressed in order to restore normal functionality.
- **Alerts by severity.** Total number of alerts on a deployed asset, broken down by severity.
- **Unhealthy network links.** Number of resources with network connectivity issues.
- **Unhealthy service endpoints.** Number of issues with external network endpoints.
- **Cloud provider service health incidents.** Number of disruptions or performance incidents caused by the cloud provider.
- **Service-level agreements.** This can include both Microsoft's commitments for uptime and connectivity of Azure services, as well as commitments made by the business to its external and internal customers.
- **Service availability.** Percentage of actual uptime cloud-hosted workloads compared to the expected uptime.
- **Recovery time objective (RTO).** The maximum acceptable time that an application can be unavailable after an incident.
- **Recovery point objective (RPO).** The maximum duration of data loss that is acceptable during a disaster. For example, if you store data in a single database, with no replication to other databases, and perform hourly backups, you could lose up to an hour of data.
- **Mean time to recover (MTTR).** The average time required to restore a component after a failure.
- **Mean time between failures (MTBF).** The duration that a component can reasonably expect to run between outages. This metric can help you calculate how often a service will become unavailable.
- **Backup health.** Number of backups actively being synchronized.
- **Recovery health.** Number of recovery operations successfully performed.

Risk tolerance indicators

Cloud platforms offer a baseline set of features that allow deployment teams to effectively manage small deployments without extensive additional planning or processes. As a result, small dev/test or experimental first workloads that include a relatively small amount of cloud-based assets represent low level of risk, and will likely not need much in the way of a formal Resource Consistency policy.

As the size of your cloud estate grows the complexity of managing your assets becomes significantly more difficult. With more assets on the cloud, the ability identify ownership of resources and control resource useful becomes critical to minimizing risks. As more mission-critical workloads are deployed to the cloud, service uptime becomes more critical, and tolerance for service disruption potential cost overruns diminishes rapidly.

In the early stages of cloud adoption, work with your IT operations team and business stakeholders to identify business risks related to resource consistency, then determine an acceptable baseline for risk tolerance. This section of the Cloud Adoption Framework provides examples, but the detailed risks and baselines for your company or deployments may be different.

Once you have a baseline, establish minimum benchmarks representing an unacceptable increase in your identified risks. These benchmarks act as triggers for when you need to take action to remediate these risks. The following are a few examples of how operational metrics, such as those discussed above, can justify an increased investment in the Resource Consistency discipline.

- **Tagging and naming trigger.** A company with more than *x* resources lacking required tagging information or not obeying naming standards should consider investing in the Resource Consistency discipline to help refine these standards and ensure consistent application of them to cloud-deployed assets.
- **Overprovisioned resources trigger.** If a company has more than *x%* of assets regularly using small amounts of their available memory, CPU, or network capabilities, investment in the Resource Consistency discipline is suggested to help optimize resources usage for these items.
- **Underprovisioned resources trigger.** If a company has more than *x%* of assets regularly exhausting most of their available memory, CPU, or network capabilities, investment in the Resource Consistency discipline is suggested to help ensure these assets have the resources necessary to prevent service interruptions.
- **Resource age trigger.** A company with more than *x* resources that haven't been updated in over *y* months could benefit from investment in the Resource Consistency discipline aimed at ensuring active resources are patched and healthy, while retiring obsolete or otherwise unused assets.
- **Service-level agreement trigger.** A company that cannot meet its service-level agreements to its external customers or internal partners should invest in the Deployment Acceleration discipline to reduce system downtime.
- **Recovery time triggers.** If a company exceeds the required thresholds for recovery time following a system failure, it should invest in improving its Deployment Acceleration discipline and systems design to reduce or eliminate failures or the effect of individual component downtime.
- **VM health trigger.** A company that has more than *x%* of VMs experiencing a critical health issue should invest in the Resource Consistency discipline to identify issues and improve VM stability.
- **Network health trigger.** A company that has more than *x%* of network subnets or endpoints experiencing connectivity issues should invest in the Resource Consistency discipline to identify and resolve network issues.
- **Backup coverage trigger.** A company with *x%* of mission-critical assets without up-to-date backups in place would benefit from an increased investment in the Resource Consistency discipline to ensure a consistent backup strategy.
- **Backup health trigger.** A company experiencing more than *x%* failure of restore operations should invest in the Resource Consistency discipline to identify problems with backup and ensure important resources are protected.

The exact metrics and triggers you use to gauge risk tolerance and the level of investment in the Resource

Consistency discipline will be specific to your organization, but the examples above should serve as a useful base for discussion within your cloud governance team.

Next steps

Use the Resource Consistency discipline template to document metrics and tolerance indicators that align to the current cloud adoption plan.

Review sample Resource Consistency policies as a starting point to develop your own policies to address specific business risks aligned with your cloud adoption plans.

Review sample policies

Resource Consistency sample policy statements

3/6/2021 • 4 minutes to read • Edit Online

Individual cloud policy statements are guidelines for addressing specific risks identified during your risk assessment process. These statements should provide a concise summary of risks and plans to deal with them. Each statement definition should include these pieces of information:

- **Technical risk:** A summary of the risk this policy will address.
- **Policy statement:** A clear summary explanation of the policy requirements.
- **Design options:** Actionable recommendations, specifications, or other guidance that IT teams and developers can use when implementing the policy.

The following sample policy statements address common business risks related to resource consistency. These statements are examples you can reference when drafting policy statements to address your organization's needs. These examples are not meant to be proscriptive, and there are potentially several policy options for dealing with each identified risk. Work closely with business and IT teams to identify the best policies for your unique set of risks.

Tagging

Technical risk: Without proper metadata tagging associated with deployed resources, IT operations cannot prioritize support or optimization of resources based on required SLA, importance to business operations, or operational cost. This can result in mis-allocation of IT resources and potential delays in incident resolution.

Policy statement: The following policies will be implemented:

- Deployed assets should be tagged with the following values:
  - Cost
  - Criticality
  - SLA
  - Environment
- Governance tooling must validate tagging related to cost, criticality, SLA, application, and environment. All values must align to predefined values managed by the governance team.

Potential design options: In Azure, standard name-value metadata tags are supported on most resource types. Azure Policy is used to enforce specific tags as part of resource creation.

Ungoverned subscriptions

Technical risk: Arbitrary creation of subscriptions and management groups can lead to isolated sections of your cloud estate that are not properly subject to your governance policies.

Policy statement: Creation of new subscriptions or management groups for any mission-critical applications or protected data will require a review from the cloud governance team. Approved changes will be integrated into a proper blueprint assignment.

Potential design options: Lock down administrative access to your organizations Azure management groups to only approved governance team members who will control the subscription creation and access control process.

Manage updates to virtual machines

Technical risk: Virtual machines (VMs) that are not up-to-date with the latest updates and software patches are vulnerable to security or performance issues, which can result in service disruptions.

Policy statement: Governance tooling must enforce that automatic updates are enabled on all deployed VMs. Violations must be reviewed with operational management teams and remediated in accordance with operations policies. Assets that are not automatically updated must be included in processes owned by IT operations.

Potential design options: For Azure hosted VMs, you can provide consistent update management using the Update Management solution in Azure Automation.

Deployment compliance

Technical risk: Deployment scripts and automation tooling that is not fully vetted by the cloud governance team can result in resource deployments that violate policy.

Policy statement: The following policies will be implemented:

- Deployment tooling must be approved by the cloud governance team to ensure ongoing governance of deployed assets.
- Deployment scripts must be maintained in central repository accessible by the cloud governance team for periodic review and auditing.

Potential design options: Consistent use of Azure Blueprints to manage automated deployments allows consistent deployments of Azure resources that adhere to your organization's governance standards and policies.

Monitoring

Technical risk: Improperly implemented or inconsistently instrumented monitoring can prevent the detection of workload health issues or other policy compliance violations.

Policy statement: The following policies will be implemented:

- Governance tooling must validate that all assets are included in monitoring for resource depletion, security, compliance, and optimization.
- Governance tooling must validate that the appropriate level of logging data is being collected for all applications and data.

Potential design options: Azure Monitor is the default monitoring service in Azure, and consistent monitoring can be enforced via Azure Blueprints when deploying resources.

Disaster recovery

Technical risk: Resource failure, deletions, or corruption can result in disruption of mission-critical applications or services and the loss of sensitive data.

Policy statement: All mission-critical applications and protected data must have backup and recovery solutions implemented to minimize business impact of outages or system failures.

Potential design options: The Azure Site Recovery service provides backup, recovery, and replication capabilities that minimize outage duration in business continuity and disaster recovery (BCDR) scenarios.

Next steps

Use the samples mentioned in this article as a starting point to develop policies that address specific business risks that align with your cloud adoption plans.

To begin developing your own custom Resource Consistency policy statements, download the Resource Consistency discipline template.

To accelerate adoption of this discipline, choose the actionable governance guide that most closely aligns with your environment. Then modify the design to incorporate your specific corporate policy decisions.

Building on risks and tolerance, establish a process for governing and communicating Resource Consistency policy adherence.

Establish policy compliance processes

Resource Consistency policy compliance processes

3/6/2021 • 5 minutes to read • Edit Online

This article discusses an approach to policy adherence processes that govern resource consistency. Effective cloud resource consistency governance starts with recurring manual processes designed to identify operational inefficiencies, improve management of deployed resources, and ensure mission-critical workloads have minimal disruptions. These manual processes are supplemented with monitoring, automation, and tooling to help reduce the overhead of governance and allow for faster response to policy deviation.

Planning, review, and reporting processes

Cloud platforms provide an array of management tools and features that you can use to organize, provision, scale, and minimize downtime. Using these tools to effectively structure and operate your cloud deployments in ways that remediate potential risks requires well-considered processes and policies in addition to close cooperation with IT operations teams and business stakeholders.

The following is a set of example processes commonly involved in the Resource Consistency discipline. Use these examples as a starting point when planning the processes that will allow you to continue to update Resource Consistency policy based on business change and feedback from the development and IT teams tasked with turning guidance into action.

Initial risk assessment and planning: As part of your initial adoption of the Resource Consistency discipline, identify your core business risks and tolerances related to operations and IT management. Use this information to discuss specific technical risks with members of your IT teams and workload owners to develop a baseline set of Resource Consistency policies designed to remediate these risks, establishing your initial governance strategy.

Deployment planning: Before deploying any asset, perform a review to identify any new operational risks. Establish resource requirements and expected demand patterns, and identify scalability needs and potential usage optimization opportunities. Also ensure backup and recovery plans are in place.

Deployment testing: As part of deployment, the cloud governance team, in cooperation with your cloud operations teams, will be responsible for reviewing the deployment to validate Resource Consistency policy compliance.

Annual planning: On an annual basis, perform a high-level review of Resource Consistency strategy. Explore future corporate expansion plans or priorities and update cloud adoption strategies to identify potential risk increase or other emerging resource consistency needs. Also use this time to review the latest best practices for cloud resource consistency and integrate these into your policies and review processes.

Quarterly review and planning: On a quarterly basis perform a review of operational data and incident reports to identify any changes required in Resource Consistency policy. As part of this process, review changes in resource usage and performance to identify assets that require increases or decreases in resource allocation, and identify any workloads or assets that are candidates for retirement.

This planning process is also a good time to evaluate the current membership of your cloud governance team for knowledge gaps related to new or changing policy and risks associated with the Resource Consistency discipline. Invite relevant IT staff to participate in reviews and planning as either temporary technical advisors or permanent members of your team.

Education and training: On a bimonthly basis, offer training sessions to make sure IT staff and developers are up-to-date on the latest Resource Consistency policy requirements and guidance. As part of this process review and update any documentation or other training assets to ensure they're in sync with the latest corporate policy

statements.

Monthly audit and reporting reviews: On a monthly basis, perform an audit on all cloud deployments to assure their continued alignment with Resource Consistency policy. Review related activities with IT staff and identify any compliance issues not already handled as part of the ongoing monitoring and enforcement process. The result of this review is a report for the cloud strategy team and each cloud adoption team to communicate overall performance and adherence to policy. The report is also stored for auditing and legal purposes.

Processes for ongoing monitoring

A successful Resource Consistency strategy depends on visibility into the current and past state of your cloud infrastructure. Without the ability to analyze the relevant metrics and data of your cloud environment's health and activity, you cannot identify changes in your risks or detect violations of your risk tolerances. The ongoing governance processes discussed above require quality data to ensure policy can be modified to optimize your cloud resource usage and improve overall performance of cloud-hosted workloads.

Ensure that your IT teams have implemented automated monitoring systems for your cloud infrastructure that capture the relevant logs data you need to evaluate risks. Be proactive in monitoring these systems to ensure prompt detection and mitigation of potential policy violation, and ensure your monitoring strategy is in line with your operational needs.

Violation triggers and enforcement actions

Because Resource Consistency policy compliance can lead to critical service disruption or significant cost overruns risks, the cloud governance team should have visibility into noncompliance incidents. Ensure IT staff have clear escalation paths for reporting these issues to the governance team members best suited to identify and verify that policy issues are mitigated when detected.

When violations are detected, you should take actions to realign with policy as soon as possible. Your IT team can automate most violation triggers using the tools outlined in the Resource Consistency toolchain for Azure.

The following triggers and enforcement actions provide examples you can reference when planning how to use monitoring data to resolve policy violations:

- **Overprovisioned resource detected.** Resources detected using less than 60% of CPU or memory capacity should automatically scale down or deprovisioning resources to reduce costs.
- **Underprovisioned resource detected.** Resources detected using more than 80% of CPU or memory capacity should automatically scale up or provisioning additional resources to provide additional capacity.
- **Untagged resource creation.** Any request to create a resource without required meta tags will be rejected automatically.
- **Critical resource outage detected.** IT staff are notified on all detected outages of mission-critical outages. If outage is not immediately resolvable, staff will escalate the issue and notify workload owners and the cloud governance team. The cloud governance team will track the issue until resolution and update guidance if policy revision is necessary to prevent future incidents.
- **Configuration drift.** Resources detected that do not conform to established baselines should trigger alerts and be automatically remediated using configuration management tools like Azure Automation, Chef, Puppet, or Ansible.

Next steps

Use the Resource Consistency discipline template to document the processes and triggers that align to the current cloud adoption plan.

For guidance on executing cloud management policies in alignment with adoption plans, see the article on discipline improvement.

Resource Consistency discipline improvement

Resource Consistency discipline improvement

3/6/2021 • 6 minutes to read • Edit Online

The Resource Consistency discipline focuses on ways of establishing policies related to the operational management of an environment, application, or workload. Within the Five Disciplines of Cloud Governance, the Resource Consistency discipline includes the monitoring of application, workload, and asset performance. It also includes the tasks required to meet scale demands, remediate performance service-level agreement (SLA) violations, and proactively avoid SLA violations through automated remediation.

This article outlines some potential tasks your company can engage in to better develop and mature the Resource Consistency discipline. These tasks can be broken down into planning, building, adopting, and operating phases of implementing a cloud solution, which are then iterated on allowing the development of an incremental approach to cloud governance.

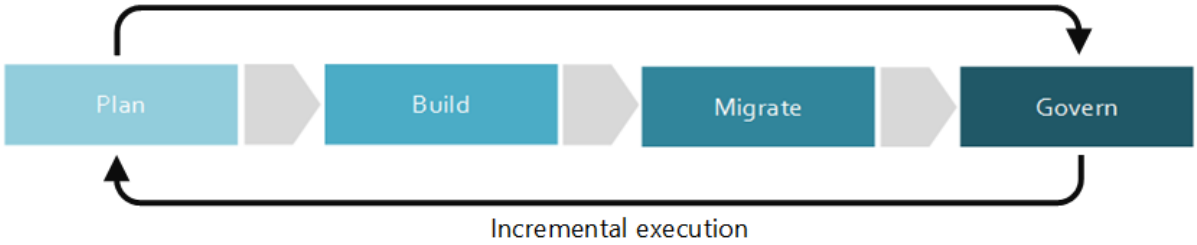


Figure 1: Phases of an incremental approach to cloud governance.

It's impossible for any one document to account for the requirements of all businesses. As such, this article outlines suggested minimum and potential example activities for each phase of the governance maturation process. The initial objective of these activities is to help you build a policy MVP and establish a framework for incremental policy improvement. Your cloud governance team will need to decide how much to invest in these activities to improve your Resource Consistency discipline.

Caution

Neither the minimum or potential activities outlined in this article are aligned to specific corporate policies or third-party compliance requirements. This guidance is designed to help facilitate the conversations that will lead to alignment of both requirements with a cloud governance model.

Planning and readiness

This phase of governance maturity bridges the divide between business outcomes and actionable strategies. During this process, the leadership team defines specific metrics, maps those metrics to the digital estate, and begins planning the overall migration effort.

Minimum suggested activities:

- Evaluate your Resource Consistency toolchain options.
- Understand the licensing requirements for your cloud strategy.
- Develop a draft architecture guidelines document and distribute to key stakeholders.
- Become familiar with the Resource Manager you use to deploy, manage, and monitor all the resources for your solution as a group.
- Educate and involve the people and teams affected by the development of architecture guidelines.
- Add prioritized resource deployment tasks to your migration backlog.

Potential activities:

- Work with the business stakeholders and your cloud strategy team to understand the desired cloud accounting approach and cost accounting practices within your business units and organization as a whole.
- Define your monitoring and policy enforcement requirements.
- Examine the business value and cost of outage to define remediation policy and SLA requirements.
- Determine whether you'll deploy a simple workload or multiple team governance strategy for your resources.
- Determine scalability requirements for your planned workloads.

Build and predeployment

Several technical and nontechnical prerequisites are required to successful migrate an environment. This process focuses on the decisions, readiness, and core infrastructure that proceeds a migration.

Minimum suggested activities:

- Implement your Resource Consistency toolchain by rolling out in a predeployment phase.
- Update the architecture guidelines document and distribute to key stakeholders.
- Implement resource deployment tasks on your prioritized migration backlog.
- Develop educational materials and documentation, awareness communications, incentives, and other programs to help drive user adoption.

Potential activities:

- Decide on a subscription design strategy, choosing the subscription patterns that best fit your organization and workload needs.
- Use a resource consistency strategy to enforce architecture guidelines over time.
- Implement resource naming, and tagging standards for your resources to match your organizational and accounting requirements.
- To create proactive point-in-time governance, use deployment templates and automation to enforce common configurations and a consistent grouping structure when deploying resources and resource groups.
- Establish a least-privilege permissions model, where users have no permissions by default.
- Determine who in your organization owns each workload and account, and who will need to access to maintain or modify these resources. Define cloud roles and responsibilities that match these needs and use these roles as the basis for access control.
- Define dependencies between resources.
- Implement automated resource scaling to match requirements defined during the planning phase.
- Conduct access performance to measure the quality of services received.
- Consider deploying Azure Policy to manage SLA enforcement using configuration settings and resource creation rules.

Adopt and migrate

Migration is an incremental process that focuses on the movement, testing, and adoption of applications or workloads in an existing digital estate.

Minimum suggested activities:

- Migrate your Resource Consistency toolchain from predeployment to production.
- Update the architecture guidelines document and distribute to key stakeholders.
- Develop educational materials and documentation, awareness communications, incentives, and other programs to help drive user adoption.
- Migrate any existing automated remediation scripts or tools to support defined SLA requirements.

Potential activities:

- Complete and test monitoring and reporting data with your chosen on-premises, cloud gateway, or hybrid solution.
- Determine whether changes need to be made to SLA or management policy for resources.
- Improve operations tasks by implementing query capabilities to efficiently find resource across your cloud estate.
- Align resources to changing business needs and governance requirements.
- Ensure that your virtual machines, virtual networks, and storage accounts reflect actual resource access needs during each release, and adjust as necessary.
- Verify automated scaling of resources meets access requirements.
- Review user access to resources, resource groups, and Azure subscriptions, and adjust access controls as necessary.
- Monitor changes in resource access plans and validate with stakeholders if additional sign-offs are needed.
- Update changes to the architecture guidelines document to reflect actual costs.
- Determine whether your organization requires clearer financial alignment to P&Ls for business units.
- For global organizations, implement your SLA compliance or sovereignty requirements.
- For cloud aggregation, deploy a gateway solution to a cloud provider.
- For tools that don't allow for hybrid or gateway options, tightly couple monitoring with an operational monitoring tool that spans all datacenters and clouds.

Operate and post-implementation

Once the transformation is complete, governance and operations must live on for the natural lifecycle of an application or workload. This phase of governance maturity focuses on the activities that commonly come after the solution is implemented and the transformation cycle begins to stabilize.

Minimum suggested activities:

- Customize your Resource Consistency toolchain based on your organization's changing needs.
- Consider automating any notifications and reports to reflect actual resource usage.
- Refine architecture guidelines to guide future adoption processes.
- Educate affected teams periodically to ensure ongoing adherence to the architecture guidelines.

Potential activities:

- Adjust plans quarterly to reflect changes to actual resources.
- Automatically apply and enforce governance requirements during future deployments.
- Evaluate underused resources and determine whether they're worth continuing.
- Detect misalignments and anomalies between planned and actual resource usage.
- Assist the cloud adoption teams and the cloud strategy team in understanding and resolving these anomalies.
- Determine whether changes need to be made to your Resource Consistency discipline for billing and SLAs.
- Evaluate logging and monitoring tools to determine whether your on-premises, cloud gateway, or hybrid solution needs adjusting.
- For business units and geographically distributed groups, determine whether your organization should consider using additional cloud management features such as Azure management groups to better apply centralized policy and meet SLA requirements.

Next steps

Now that you understand the concept of cloud resource governance, move on to learn more about how

resource access is managed in Azure in preparation for learning how to design a governance model for a simple workload or for multiple teams.

Learn about resource access management in Azure Learn about service-level agreements for Azure Learn about logging, reporting, and monitoring

Resource Consistency tools in Azure

3/6/2021 • 2 minutes to read • Edit Online

Resource consistency is one of the Five Disciplines of Cloud Governance. This discipline focuses on ways of establishing policies related to the operational management of an environment, application, or workload. Within the Five Disciplines of Cloud Governance, the Resource Consistency discipline involves monitoring of application, workload, and asset performance. It also involves the tasks required to meet scale demands, remediate performance SLA violations, and proactively avoid performance SLA violations through automated remediation.

The following is a list of Azure tools that can help mature the policies and processes that support this discipline.

TOOL	AZURE PORTAL	AZURE RESOURCE MANAGER	AZURE BLUEPRINTS	AZURE AUTOMATION	AZURE AD	AZURE BACKUP	AZURE SITE RECOVERY
Deploy resources	Yes	Yes	Yes	Yes	No	No	No
Manage resources	Yes	Yes	Yes	Yes	No	No	No
Deploy resources using templates	No	Yes	No	Yes	No	No	No
Orchestrated environment deployment	No	No	Yes	No	No	No	No
Define resource groups	Yes	Yes	Yes	No	No	No	No
Manage workload and account owners	Yes	Yes	Yes	No	No	No	No
Manage conditional access to resources	Yes	Yes	Yes	No	No	No	No
Configure Azure RBAC users	Yes	No	No	No	Yes	No	No

TOOL	AZURE PORTAL	AZURE RESOURCE MANAGER	AZURE BLUEPRINTS	AZURE AUTOMATION	AZURE AD	AZURE BACKUP	AZURE SITE RECOVERY
Assign roles and permissions to resources	Yes	Yes	Yes	No	Yes	No	No
Define dependencies between resources	No	Yes	Yes	No	No	No	No
Apply access control	Yes	Yes	Yes	No	Yes	No	No
Assess availability and scalability	No	No	No	Yes	No	No	No
Apply tags to resources	Yes	Yes	Yes	No	No	No	No
Assign Azure Policy rules	Yes	Yes	Yes	No	No	No	No
Apply automated remediation	No	No	No	Yes	No	No	No
Manage billing	Yes	No	No	No	No	No	No
Plan resources for disaster recovery	Yes	Yes	Yes	No	No	Yes	Yes
Recover data during an outage or SLA violation	No	No	No	No	No	Yes	Yes
Recover applications and data during an outage or SLA violation	No	No	No	No	No	Yes	Yes

Along with these Resource Consistency tools and features, you will need to monitor your deployed resources for performance and health issues. Azure Monitor is the default monitoring and reporting solution in Azure. Azure Monitor provides features for monitoring your cloud resources. This list shows which feature addresses common monitoring requirements.

TOOL	AZURE PORTAL	APPLICATION INSIGHTS	LOG ANALYTICS	AZURE MONITOR REST API
Log virtual machine telemetry data	No	No	Yes	No
Log virtual networking telemetry data	No	No	Yes	No
Log PaaS services telemetry data	No	No	Yes	No
Log application telemetry data	No	Yes	No	No
Configure reports and alerts	Yes	No	No	Yes
Schedule regular reports or custom analysis	No	No	No	No
Visualize and analyze log and performance data	Yes	No	No	No
Integrate with on-premises or third-party monitoring solution	No	No	No	Yes

When planning your deployment, you will need to consider where logging data is stored and how you integrate cloud-based reporting and monitoring services with your existing processes and tools.

> **NOTE**
> Organizations also use third-party DevOps tools to monitor workloads and resources. For more information, see DevOps tool integrations.

Next steps

Learn to create, assign, and manage policy definitions in Azure.

Resource access management in Azure

3/6/2021 • 4 minutes to read • Edit Online

The Govern methodology outlines the Five Disciplines of Cloud Governance, which includes resource management. What is resource access governance furthers explains how resource access management fits into the resource management discipline. Before you move on to learn how to design a governance model, it's important to understand the resource access management controls in Azure. The configuration of these resource access management controls forms the basis of your governance model.

Begin by taking a closer look at how resources are deployed in Azure.

What is an Azure resource?

In Azure, the term *resource* refers to an entity managed by Azure. For example, virtual machines, virtual networks, and storage accounts are all referred to as Azure resources.

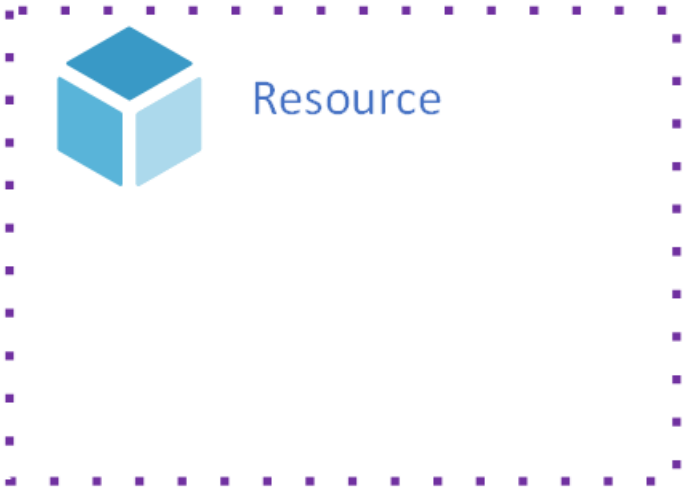


Figure 1: A resource.

What is an Azure resource group?

Each resource in Azure must belong to a resource group. A resource group is simply a logical construct that groups multiple resources together so they can be managed as a single entity **based on lifecycle and security**. For example, resources that share a similar lifecycle, such as the resources for an n-tier application may be created or deleted as a group. In other words, everything that is born together, gets managed together, and deprecates together, goes together in a resource group.

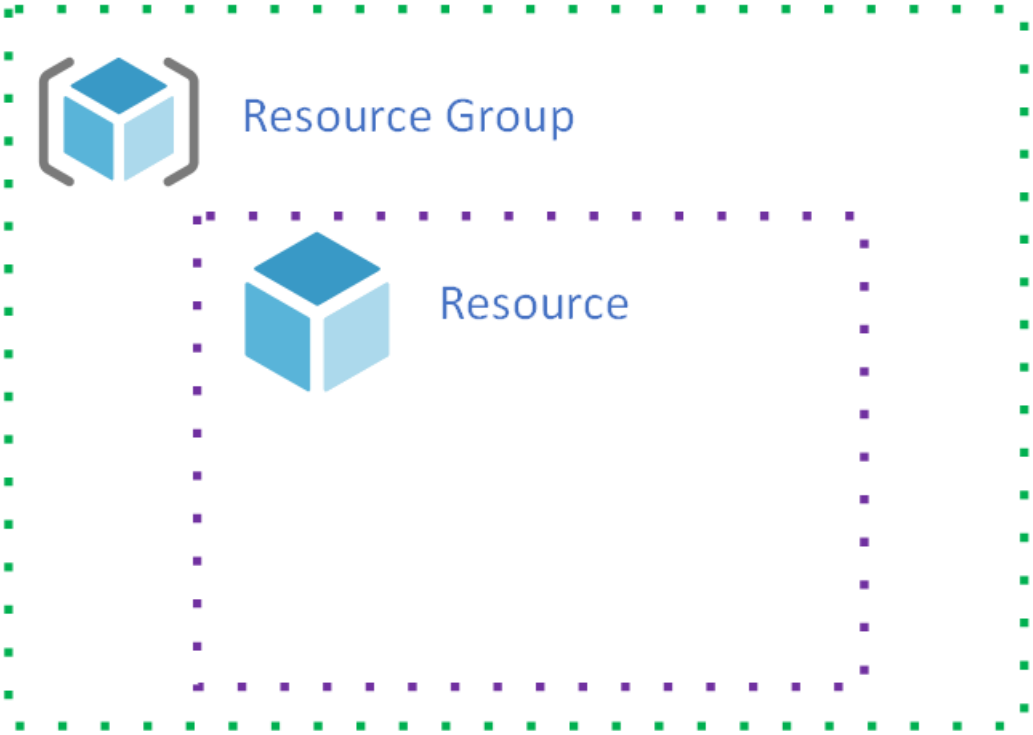


Figure 2: A resource group contains a resource.

Resource groups and the resources they contain are associated with an Azure subscription.

What is an Azure subscription?

An Azure *subscription* is similar to a resource group in that it's a logical construct that groups together resource groups and their resources. An Azure subscription is also associated with the controls used by Azure Resource Manager. Take a closer look at Azure Resource Manager to learn about the relationship between it and an Azure subscription.

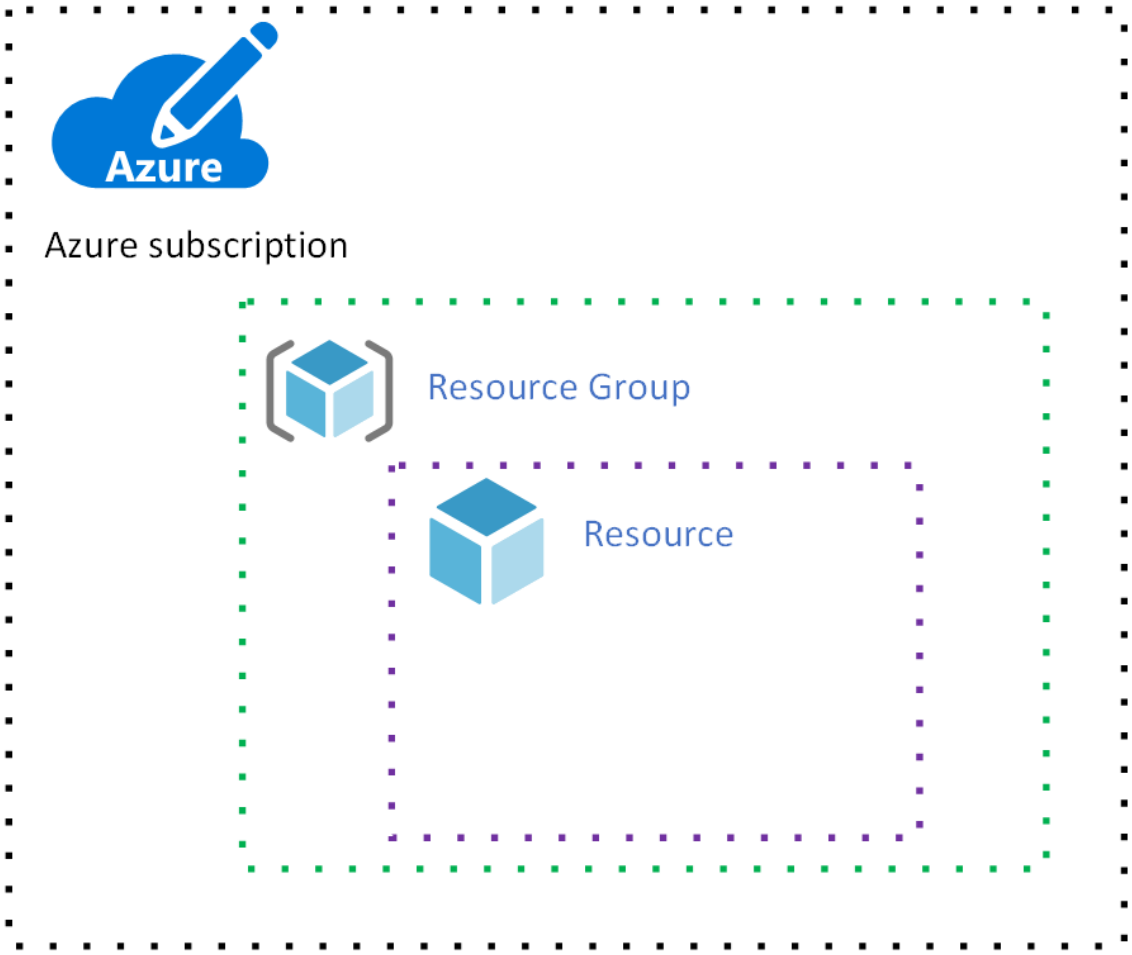


Figure 3: An Azure subscription.

What is Azure Resource Manager?

In How does Azure work? you learned that Azure includes a front end with many services that orchestrate all the functions of Azure. One of these services is Azure Resource Manager, and this service hosts the RESTful API used by clients to manage resources.

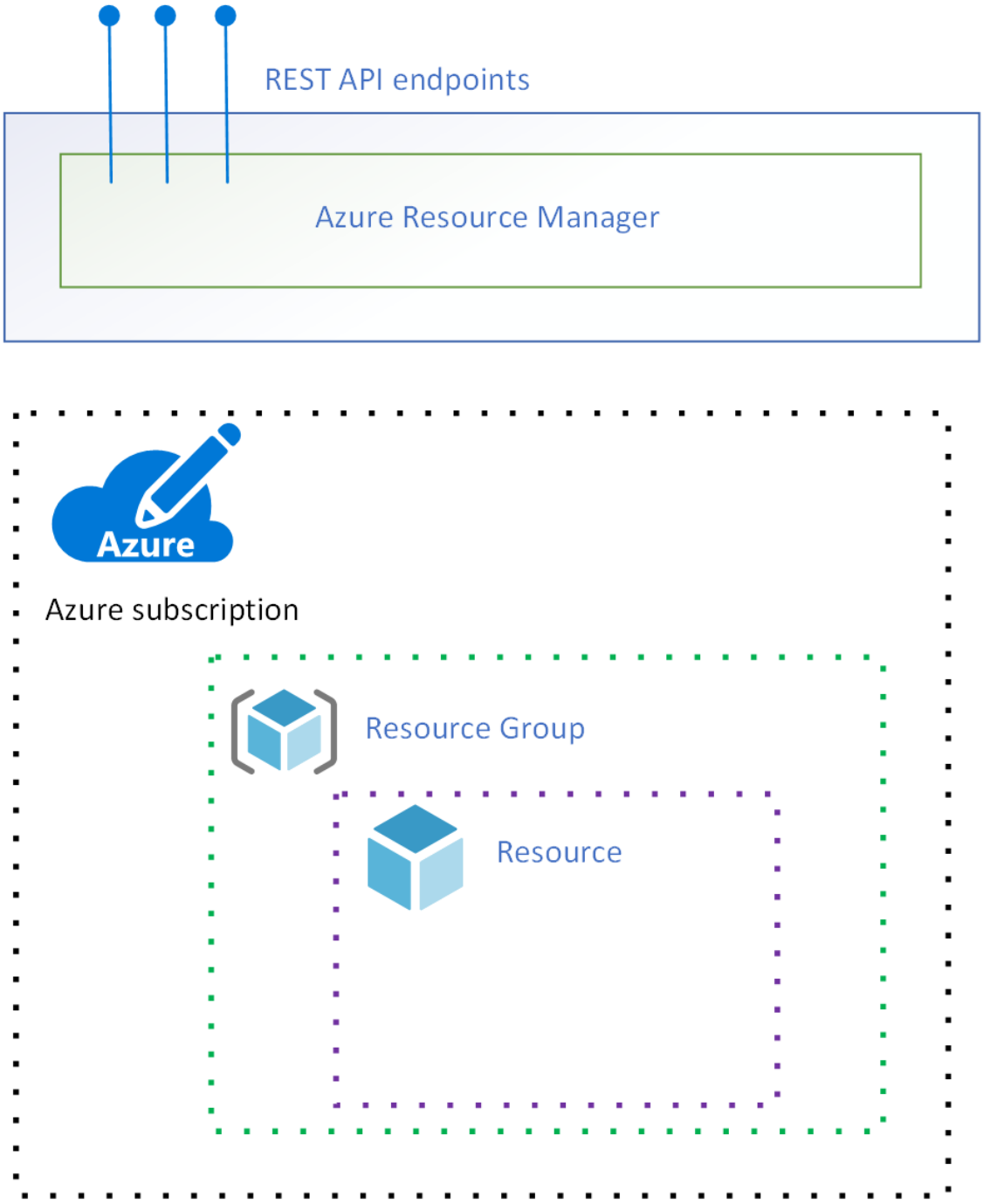


Figure 4: Azure Resource Manager.

The following figure shows three clients: PowerShell, the Azure portal, and the Azure CLI:

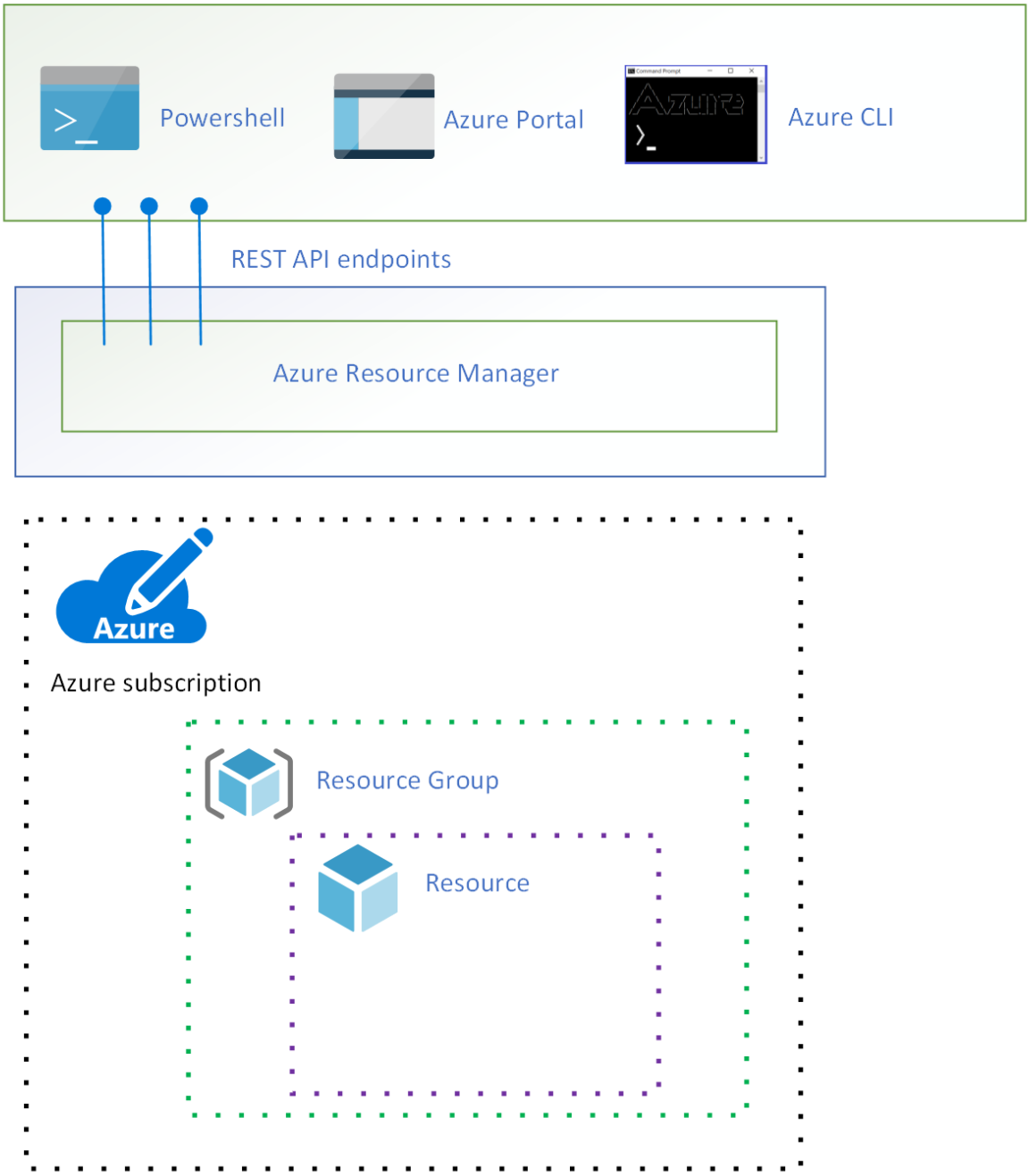


Figure 5: Azure clients connect to the Resource Manager REST API.

While these clients connect to Resource Manager using the REST API, Resource Manager does not include functionality to manage resources directly. Rather, most resource types in Azure have their own resource provider.

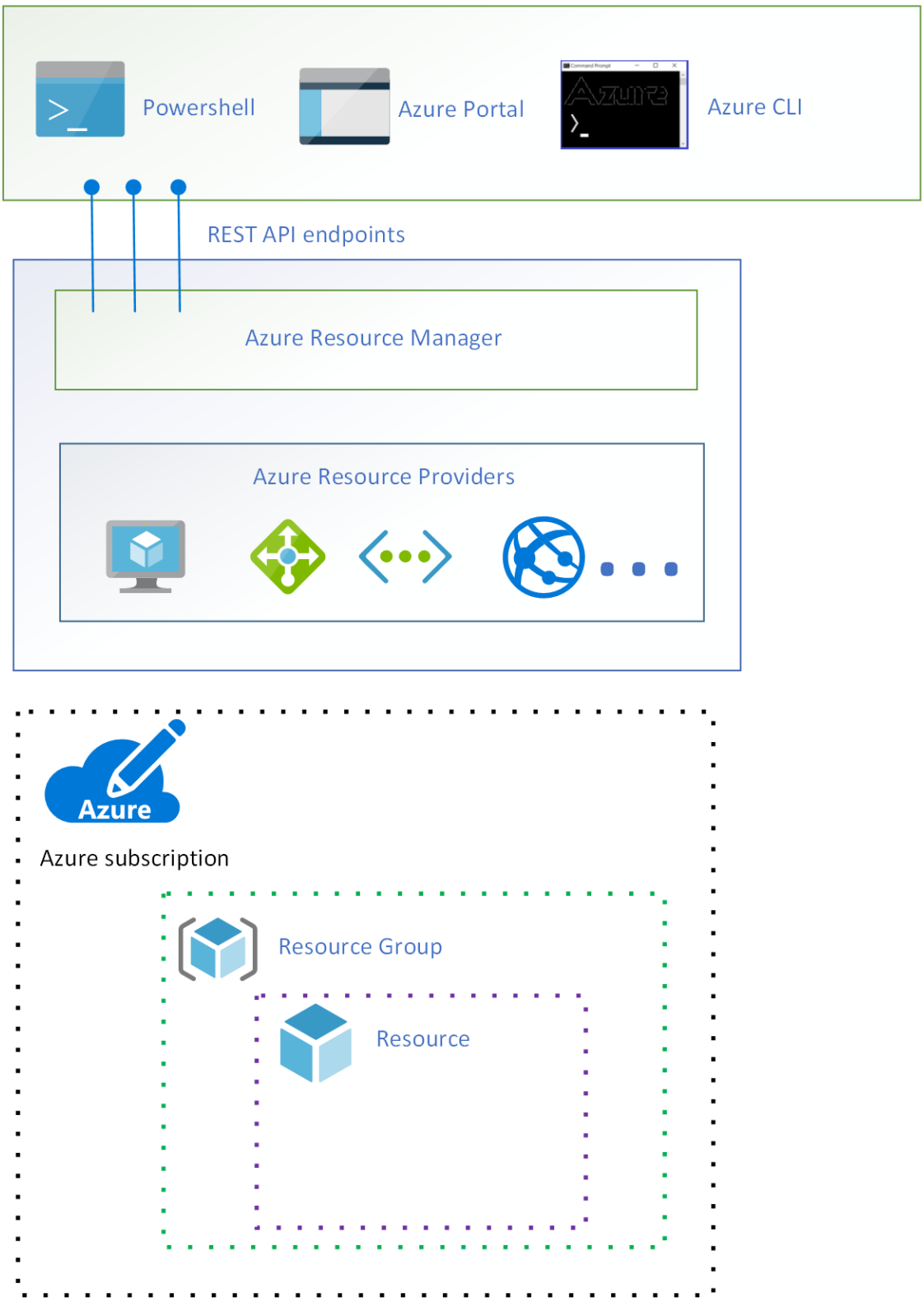


Figure 6: Azure resource providers.

When a client makes a request to manage a specific resource, Azure Resource Manager connects to the resource provider for that resource type to complete the request. For example, if a client makes a request to manage a virtual machine resource, Azure Resource Manager connects to the `Microsoft.Compute` resource provider.

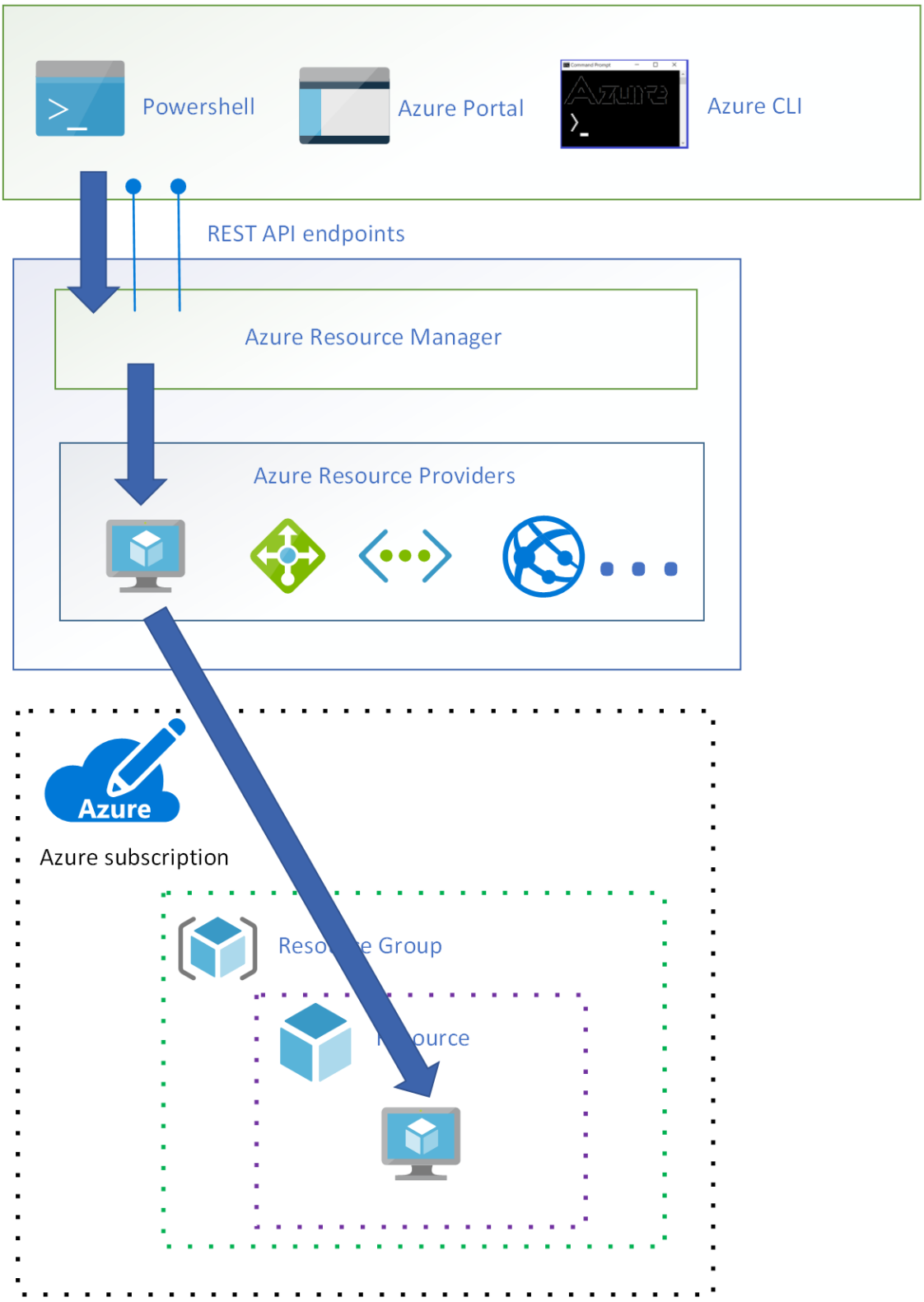


Figure 7: Azure Resource Manager connects to the `Microsoft.Compute` resource provider to manage the resource specified in the client request.

Azure Resource Manager requires the client to specify an identifier for both the subscription and the resource group in order to manage the virtual machine resource.

Now that you have an understanding of how Azure Resource Manager works, return to the discussion of how an Azure subscription is associated with the controls used by Azure Resource Manager. Before any resource management request can be executed by Azure Resource Manager, a set of controls are checked.

The first control is that a request must be made by a validated user, and Azure Resource Manager has a trusted

relationship with Azure Active Directory (Azure AD) to provide user identity functionality.

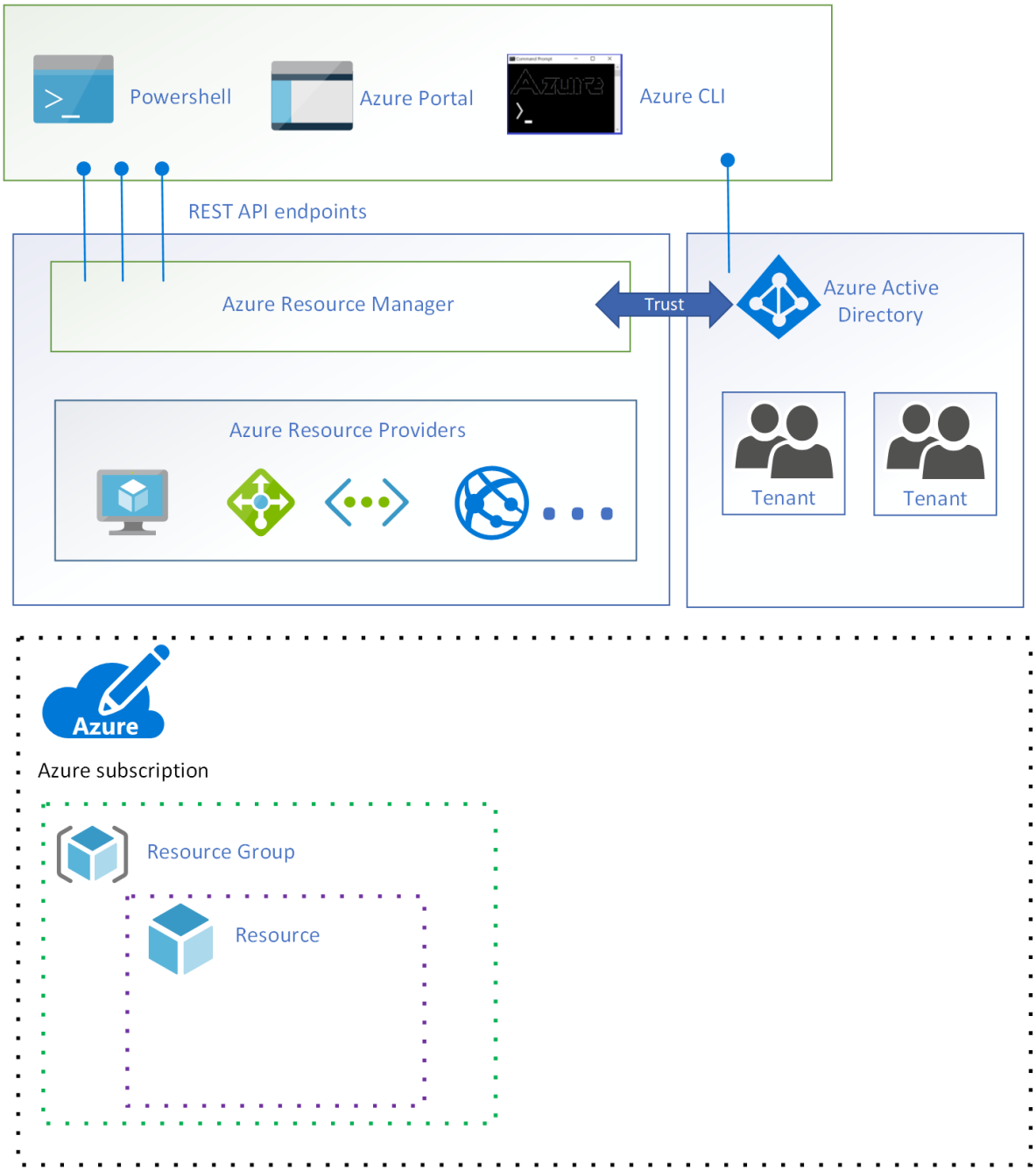


Figure 8: Azure Active Directory.

In Azure AD, users are segmented into tenants. A *tenant* is a logical construct that represents a secure, dedicated instance of Azure AD typically associated with an organization. Each subscription is associated with an Azure AD tenant.

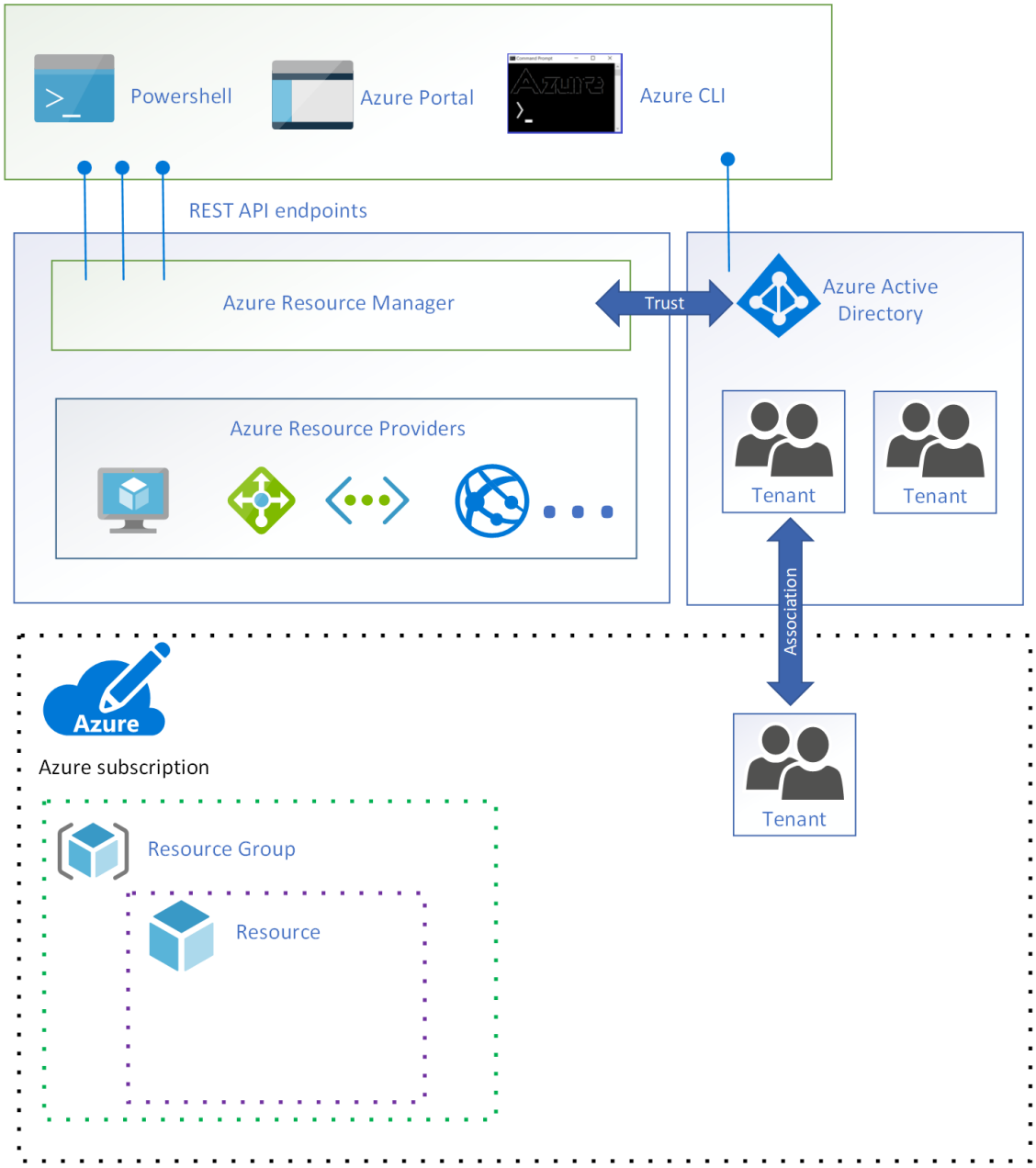


Figure 9: An Azure AD tenant associated with a subscription.

Each client request to manage a resource in a particular subscription requires that the user has an account in the associated Azure AD tenant.

The next control is a check that the user has sufficient permission to make the request. Permissions are assigned to users using Azure role-based access control (Azure RBAC).

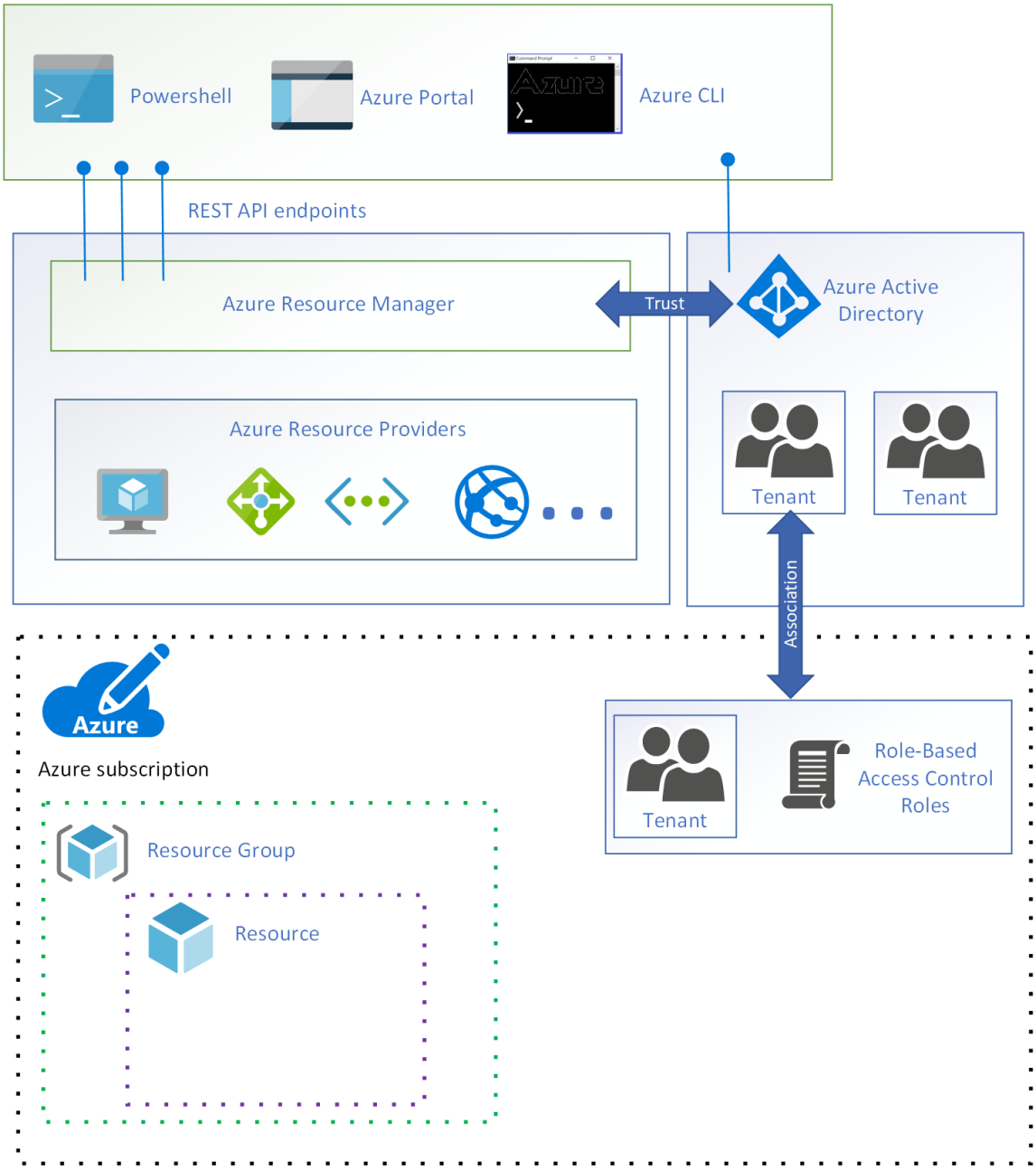


Figure 10: Each user in the tenant is assigned one or more Azure roles.

An Azure role specifies a set of permissions a user may take on a specific resource. When the role is assigned to the user, those permissions are applied. For example, the built-in `owner` role allows a user to perform any action on a resource.

The next control is a check that the request is allowed under the settings specified for Azure resource policy. Azure resource policies specify the operations allowed for a specific resource. For example, an Azure resource policy can specify that users are only allowed to deploy a specific type of virtual machine.

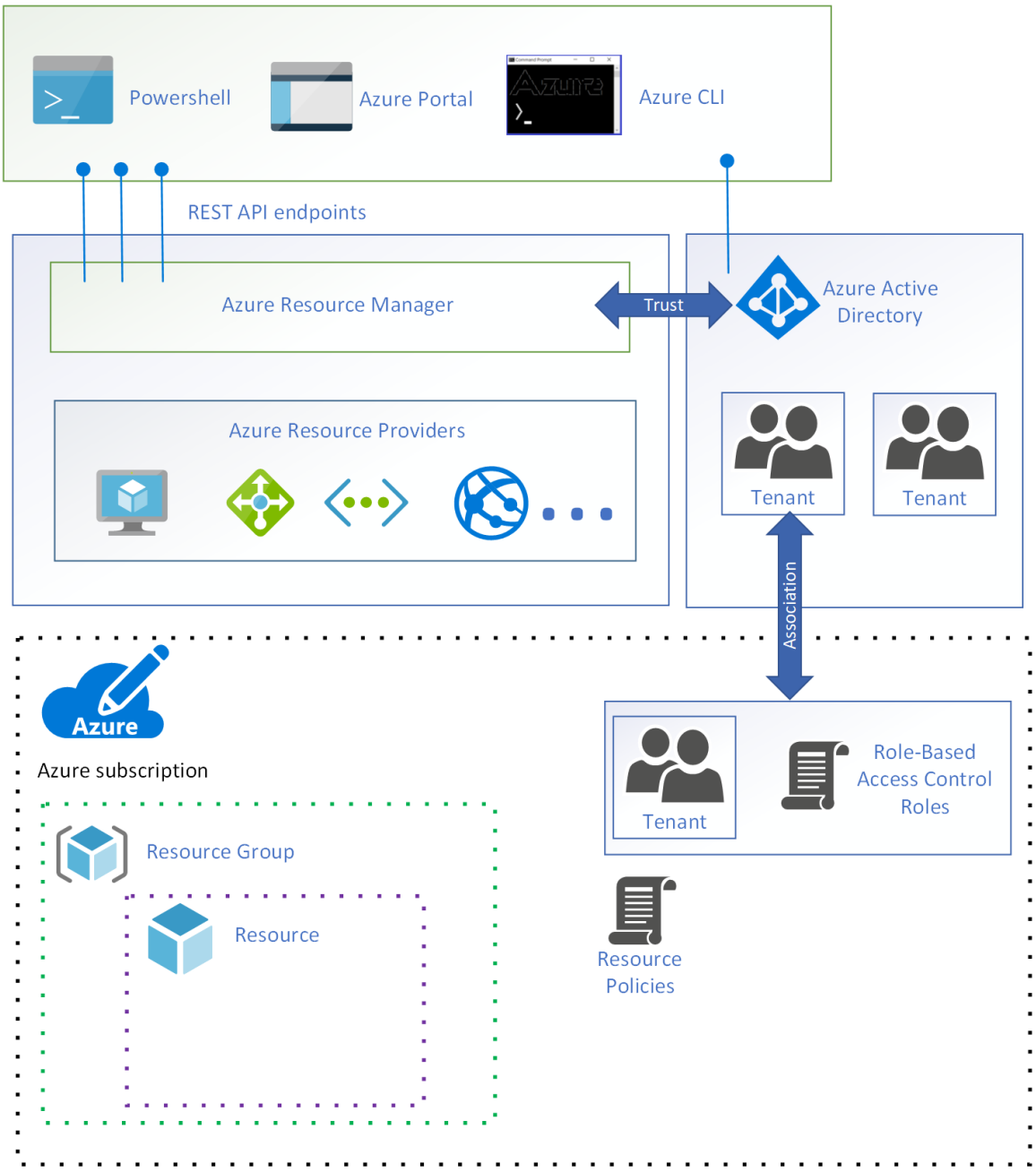


Figure 11: Azure resource policy.

The next control is a check that the request does not exceed an Azure subscription limit. For example, each subscription has a limit of 980 resource groups per subscription. If a request is received to deploy an additional resource group when the limit has been reached, it's denied.

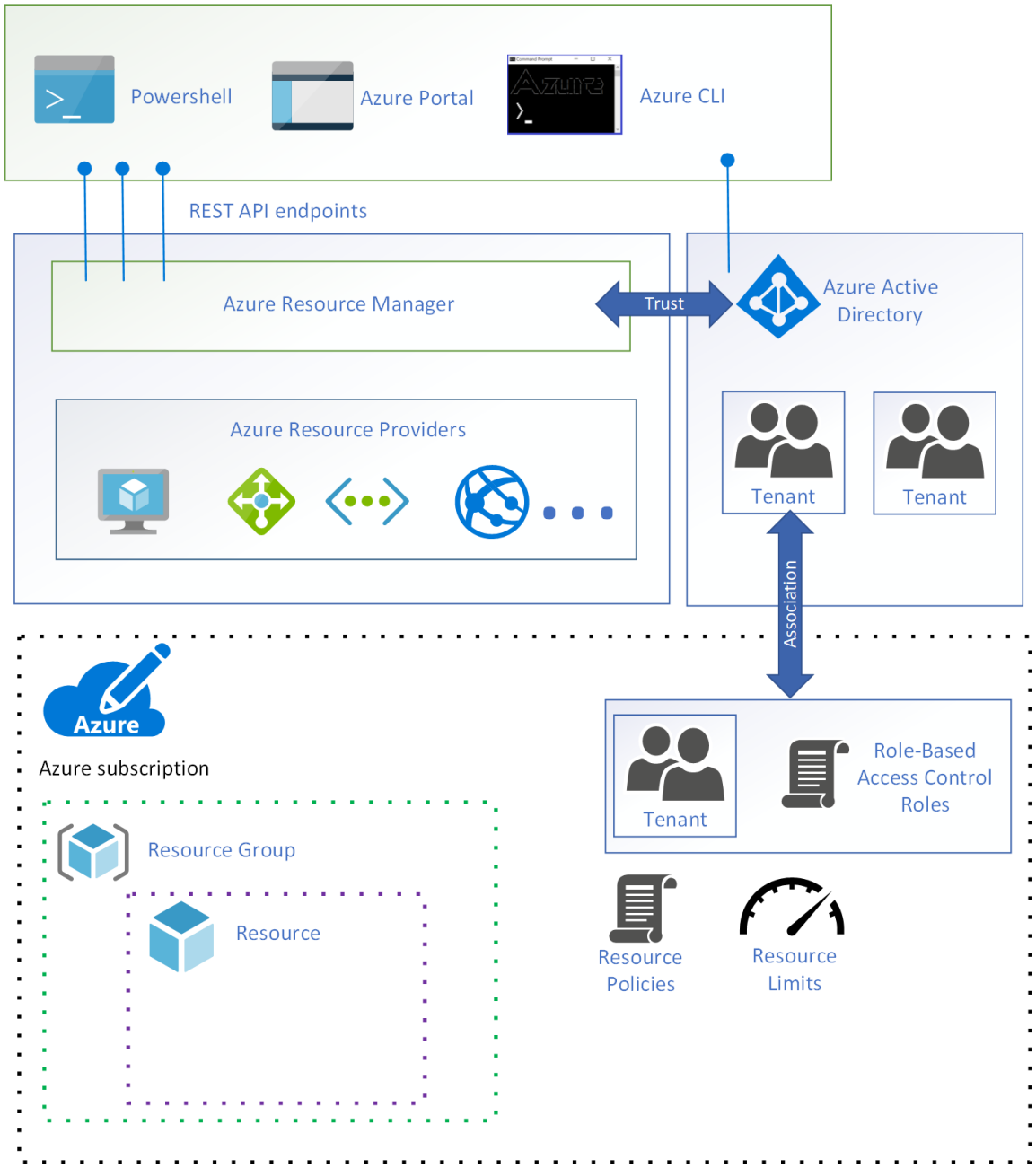


Figure 12: Azure resource limits.

The final control is a check that the request is within the financial commitment associated with the subscription. For example, if the request is to deploy a virtual machine, Azure Resource Manager verifies that the subscription has sufficient payment information.

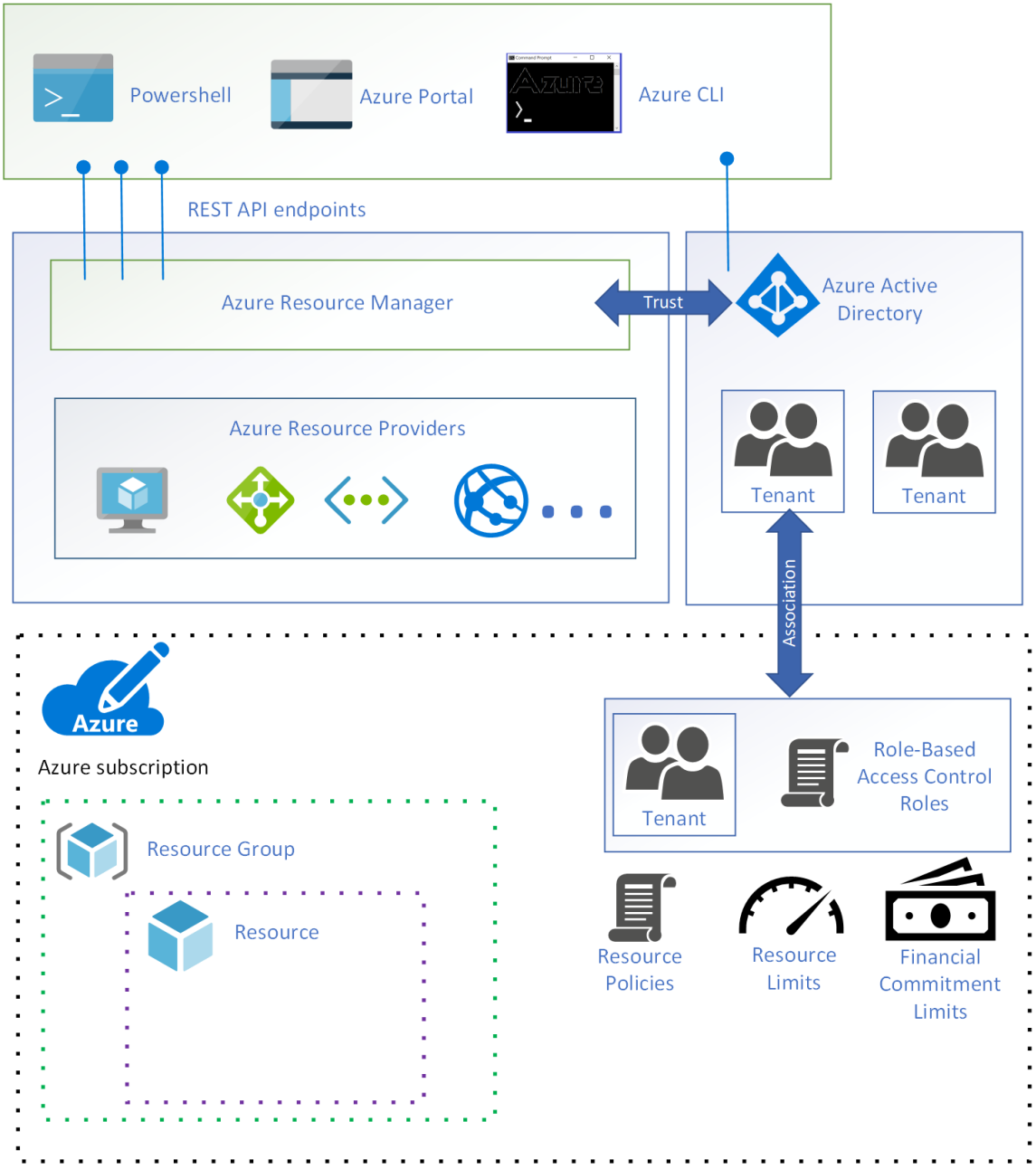


Figure 13: A financial commitment is associated with a subscription.

Summary

In this article, you learned about how resource access is managed in Azure using Azure Resource Manager.

Next steps

Now that you understand how to manage resource access in Azure, move on to learn how to design a governance model for a simple workload or for multiple teams using these services.

An overview of governance

Governance design for a simple workload

3/6/2021 • 6 minutes to read • Edit Online

The goal of this guidance is to help you learn the process for designing a resource governance model in Azure to support a single team and a simple workload. You'll look at a set of hypothetical governance requirements, then go through several example implementations that satisfy those requirements.

In the foundational adoption stage, our goal is to deploy a simple workload to Azure. This results in the following requirements:

- Identity management for a single **workload owner** who is responsible for deploying and maintaining the simple workload. The workload owner requires permission to create, read, update, and delete resources as well as permission to delegate these rights to other users in the identity management system.
- Manage all resources for the simple workload as a single management unit.

Azure licensing

Before you begin designing our governance model, it's important to understand how Azure is licensed. This is because the administrative accounts associated with your Azure license have the highest level of access to your Azure resources. These administrative accounts form the basis of your governance model.

> **NOTE**
> If your organization has an existing Microsoft Enterprise Agreement that does not include Azure, Azure can be added by making an upfront monetary commitment. For more information, see Licensing Azure for the enterprise.

When Azure was added to your organization's Enterprise Agreement, your organization was prompted to create an **Azure account**. During the account creation process, an **Azure account owner** was created, as well as an Azure Active Directory (Azure AD) tenant with a **global administrator** account. An Azure AD tenant is a logical construct that represents a secure, dedicated instance of Azure AD.

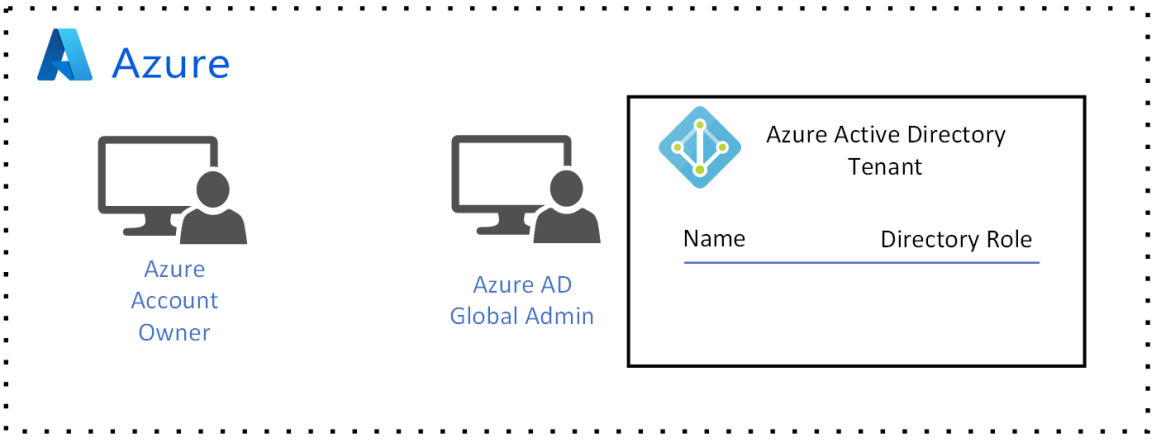


Figure 1: An Azure account with an Azure account owner and Azure AD global administrator.

Identity management

Azure only trusts Azure AD to authenticate users and authorize user access to resources, so Azure AD is our identity management system. The Azure AD Global Administrator has the highest level of permissions and can

perform all actions related to identity, including creating users and assigning permissions.

Our requirement is identity management for a single **workload owner** who is responsible for deploying and maintaining the simple workload. The workload owner requires permission to create, read, update, and delete resources as well as permission to delegate these rights to other users in the identity management system.

Our Azure AD Global Administrator will create the **workload owner** account for the workload owner:

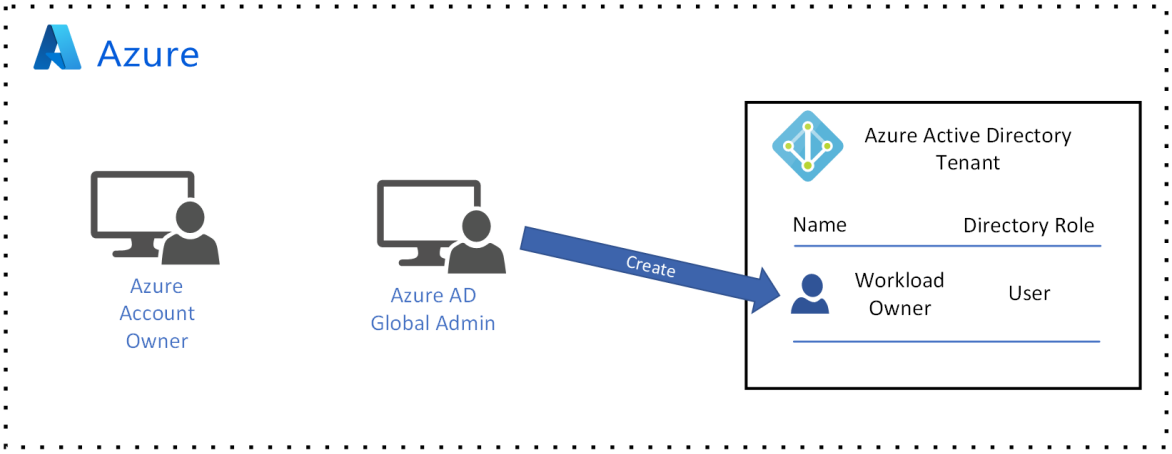


Figure 2: The Azure AD global administrator creates the workload owner user account.

You can't assign resource access permission until this user is added to a **subscription**, so you'll do that in the next two sections.

Resource management scope

As the number of resources deployed by your organization grows, the complexity of governing those resources grows as well. Azure implements a logical container hierarchy to enable your organization to manage your resources in groups at various levels of granularity, also known as **scope**.

The top level of resource management scope is the **subscription** level. A subscription is created by the Azure **account owner**, who establishes the financial commitment and is responsible for paying for all Azure resources associated with the subscription:

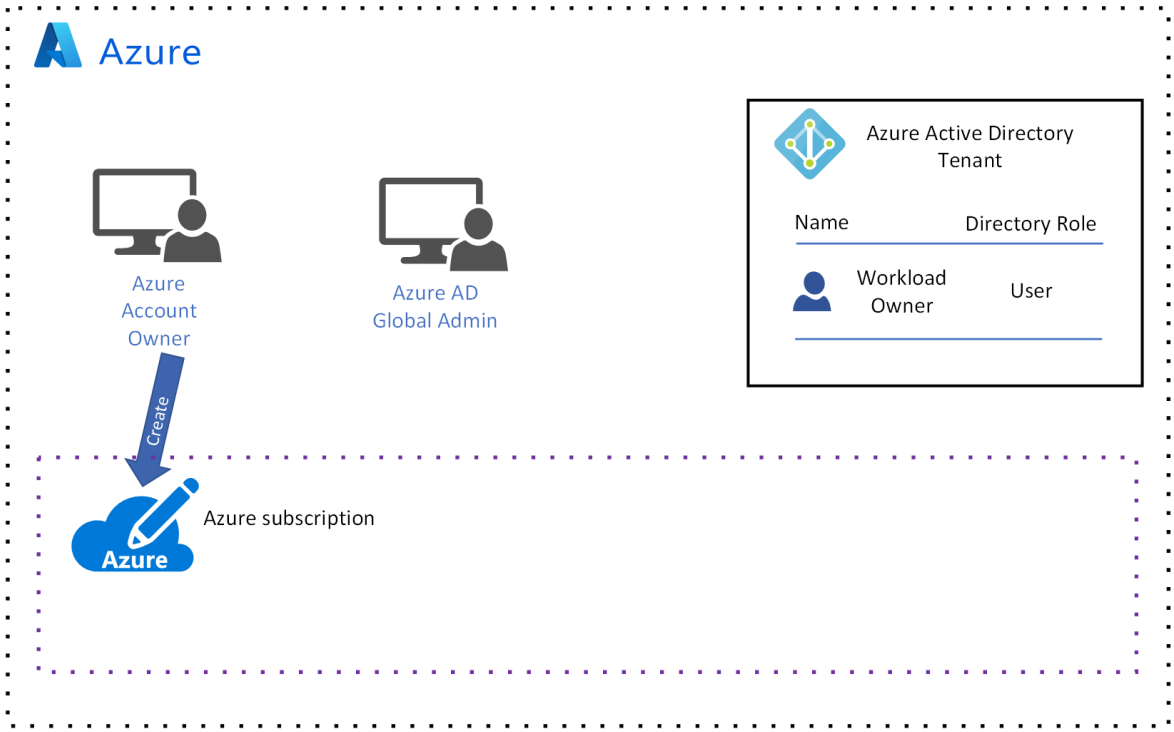


Figure 3: The Azure account owner creates a subscription.

When the subscription is created, the Azure **account owner** associates an Azure AD tenant with the subscription, and this Azure AD tenant is used for authenticating and authorizing users:

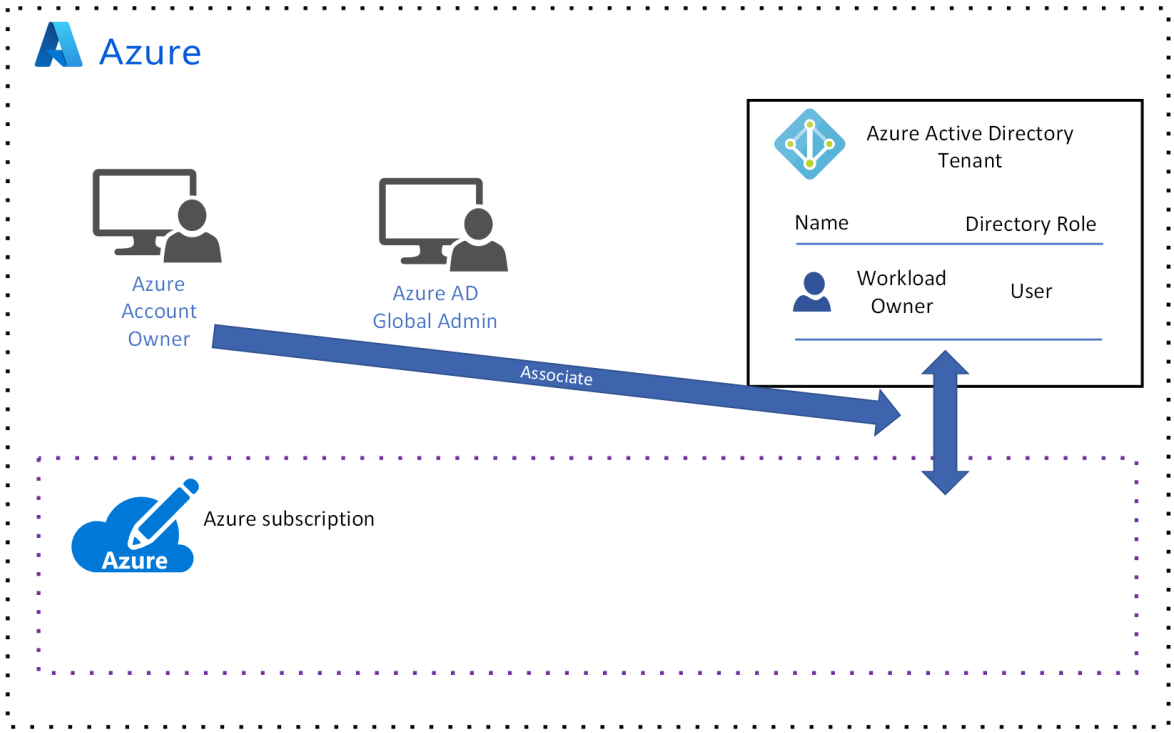


Figure 4: The Azure account owner associates the Azure AD tenant with the subscription.

You may have noticed that there is currently no user associated with the subscription, which means that no one has permission to manage resources. In practice, the **account owner** is the owner of the subscription and has permission to take any action on a resource in the subscription. In practical terms, the **account owner** is more than likely a finance person in your organization and is not responsible for creating, reading, updating, and deleting resources. Those tasks will be performed by the **workload owner**, so you need to add the **workload**

owner to the subscription and assign permissions.

Since the **account owner** is currently the only user with permission to add the **workload owner** to the subscription, they add the **workload owner** to the subscription:

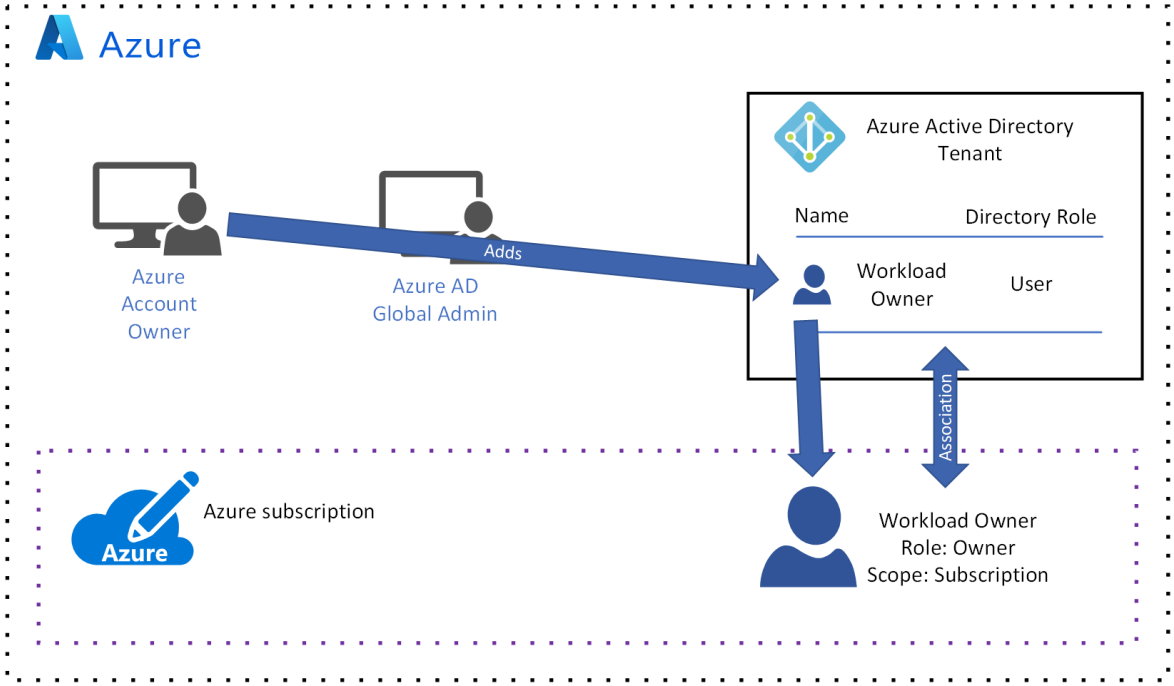


Figure 5: The Azure account owner adds the workload owner to the subscription.

The Azure **account owner** grants permissions to the **workload owner** by assigning an Azure role. The Azure role specifies a set of permissions that the **workload owner** has for an individual resource type or a set of resource types.

Notice that in this example, the **account owner** has assigned the built-in **owner** role:

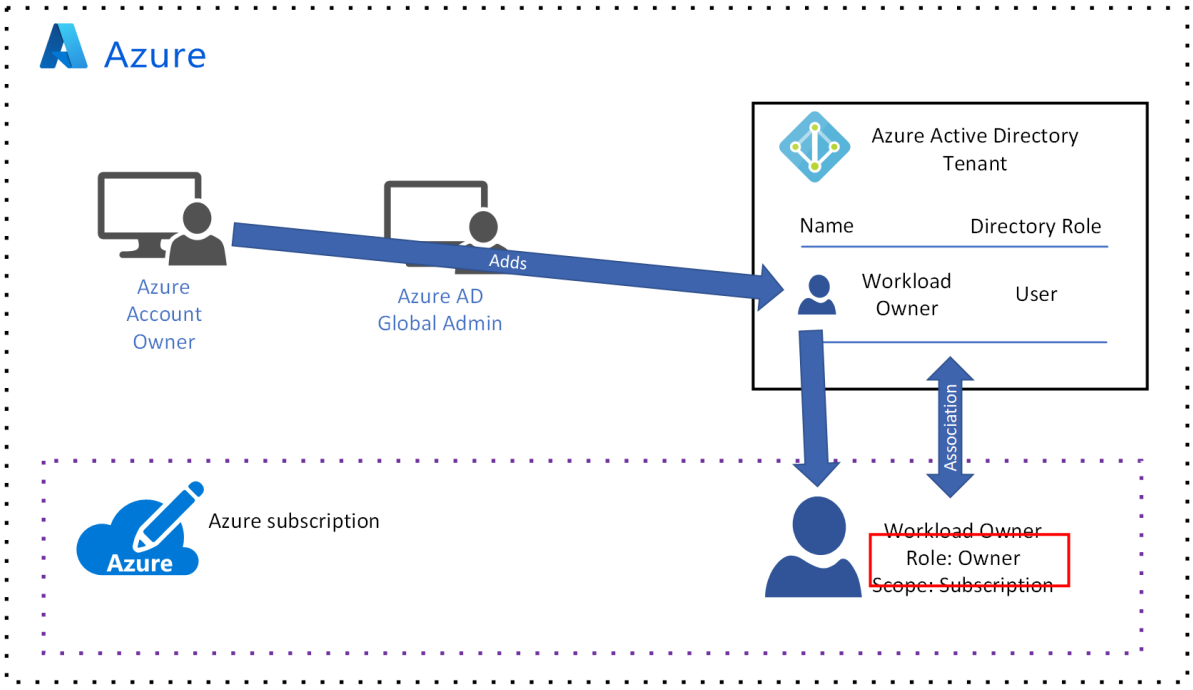


Figure 6: The workload owner was assigned the built-in Owner role.

The built-in **owner** role grants all permissions to the **workload owner** at the subscription scope.

IMPORTANT

The Azure **account owner** is responsible for the financial commitment associated with the subscription, but the **workload owner** has the same permissions. The **account owner** must trust the **workload owner** to deploy resources that are within the subscription budget.

The next level of management scope is the **resource group** level. A resource group is a logical container for resources. Operations applied at the resource group level apply to all resources in a group. Also, it's important to note that permissions for each user are inherited from the next level up unless they're explicitly changed at that scope.

To illustrate this, let's look at what happens when the **workload owner** creates a resource group:

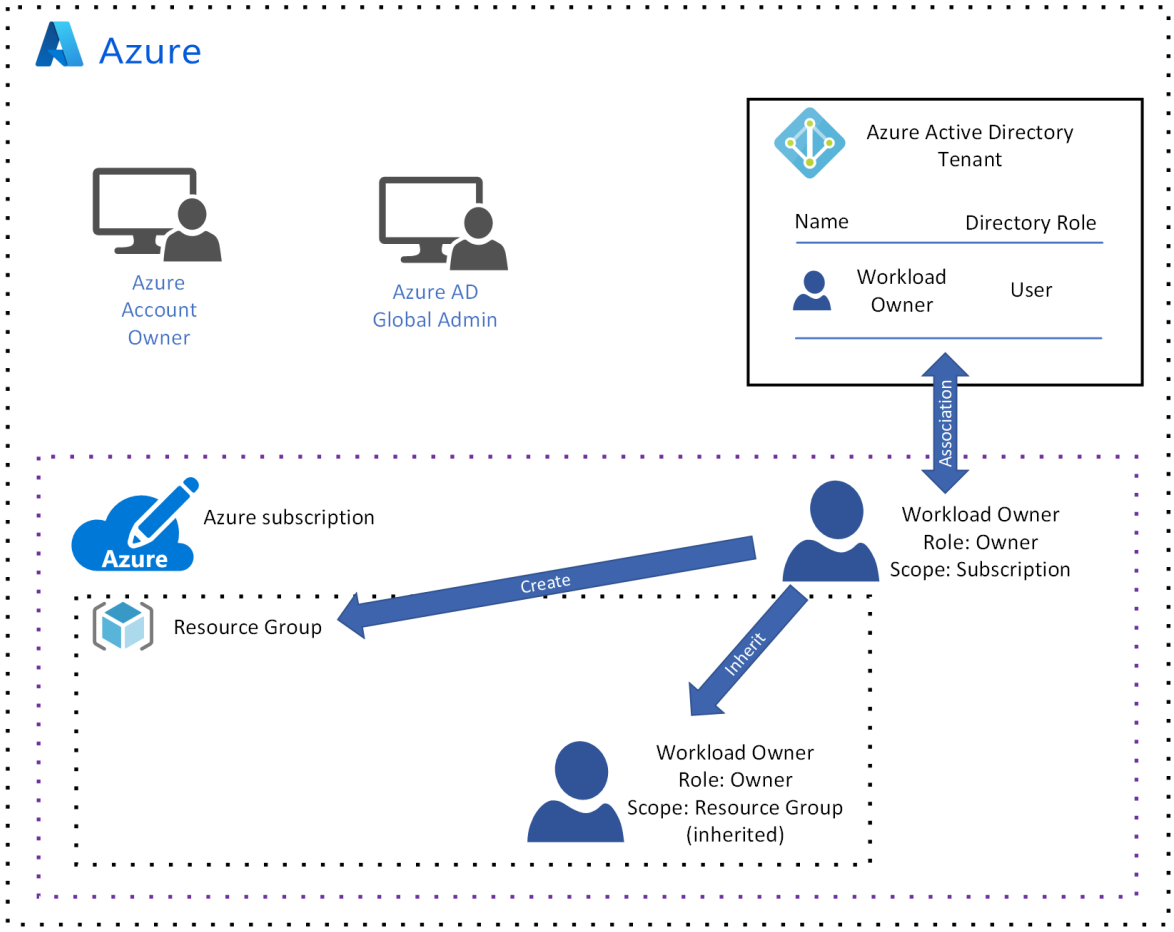


Figure 7: The workload owner creates a resource group and inherits the built-in Owner role at the resource group scope.

Again, the built-in **owner** role grants all permissions to the **workload owner** at the resource group scope. As discussed earlier, this role is inherited from the subscription level. If a different role is assigned to this user at this scope, it applies to this scope only.

The lowest level of management scope is at the **resource** level. Operations applied at the resource level apply only to the resource itself. Again, permissions at the resource level are inherited from resource group scope. For example, let's look at what happens if the **workload owner** deploys a virtual network into the resource group:

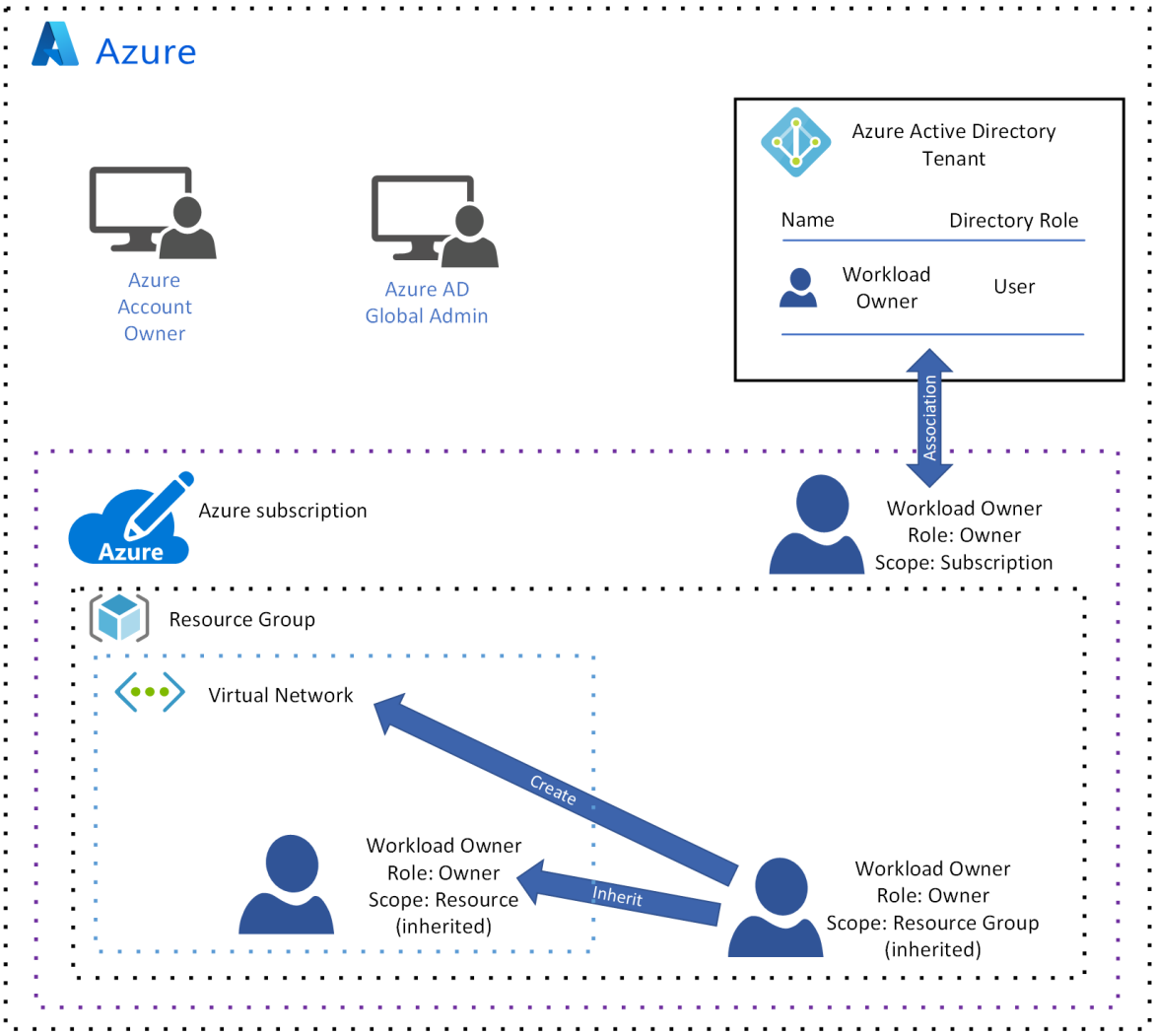


Figure 8: The workload owner creates a resource and inherits the built-in Owner role at the resource scope.

The **workload owner** inherits the Owner role at the resource scope, which means the workload owner has all permissions for the virtual network.

Implement the basic resource access management model

Let's move on to learn how to implement the governance model designed earlier.

To begin, your organization requires an Azure account. If your organization has an existing Microsoft Enterprise Agreement that does not include Azure, Azure can be added by making an upfront monetary commitment. For more information, see Licensing Azure for the enterprise.

When your Azure account is created, you specify a person in your organization to be the Azure **account owner**. An Azure Active Directory (Azure AD) tenant is then created by default. Your Azure **account owner** must create the user account for the person in your organization who is the **workload owner**.

Next, your Azure **account owner** must create a subscription and associate the Azure AD tenant with it.

Finally, now that the subscription is created and your Azure AD tenant is associated with it, you can add the **workload owner** to the subscription with the built-in **owner** role.

Next steps

Learn about resource access for multiple teams

Governance design for multiple teams

5/24/2021 • 24 minutes to read • Edit Online

The goal of this guidance is to help you learn the process for designing a resource governance model in Azure to support multiple teams, multiple workloads, and multiple environments. First you'll look at a set of hypothetical governance requirements, then go through several example implementations that satisfy those requirements.

The requirements are:

- The enterprise plans to transition new cloud roles and responsibilities to a set of users and therefore requires identity management for multiple teams with different resource access needs in Azure. This identity management system is required to store the identity of the following users:
  - The individual in your organization responsible for ownership of **subscriptions**.
  - The individual in your organization responsible for the **shared infrastructure resources** used to connect your on-premises network to a virtual network in Azure.
  - Two individuals in your organization responsible for managing a **workload**.
- Support for multiple **environments**. An environment is a logical grouping of resources, such as virtual machines, virtual networking, and network traffic routing services. These groups of resources have similar management and security requirements and are typically used for a specific purpose such as testing or production. In this example, the requirement is for four environments:
  - A **shared infrastructure environment** that includes resources shared by workloads in other environments. For example, a virtual network with a gateway subnet that provides connectivity to on-premises.
  - A **production environment** with the most restrictive security policies. Could include internal or external facing workloads.
  - A **nonproduction environment** for development and testing work. This environment has security, compliance, and cost policies that differ from those in the production environment. In Azure, this takes the form of an Enterprise Dev/Test subscription.
  - A **sandbox environment** for proof of concept and education purposes. This environment is typically assigned per employee participating in development activities and has strict procedural and operational security controls in place to prevent corporate data from landing here. In Azure, these take the form of Visual Studio subscriptions. These subscriptions should also *not* be tied to the enterprise Azure Active Directory.
- A **permissions model of least privilege** in which users have no permissions by default. The model must support the following:
  - A single trusted user at the subscription scope, treated like a service account and granted permission to assign resource access rights.
  - Each workload owner is denied access to resources by default. Resource access rights are granted explicitly by the single trusted user at the resource group scope.
  - Management access for the shared infrastructure resources, limited to the shared infrastructure owners.
  - Management access for each workload restricted to the workload owner in production, and increasing levels of control as development proceeds through the various deployment environments (development, test, staging, and production).
  - The enterprise does not want to have to manage roles independently in each of the three main environments, and therefore requires the use of only built-in roles available in Azure role-based access control (Azure RBAC). If the enterprise absolutely requires custom roles, additional processes

would be needed to synchronize custom roles across the three environments.

- Cost tracking by workload owner name, environment, or both.

Identity management

Before you can design identity management for your governance model, it's important to understand the four major areas it encompasses:

- **Administration:** The processes and tools for creating, editing, and deleting user identity.
- **Authentication:** Verifying user identity by validating credentials, such as a user name and password.
- **Authorization:** Determining which resources an authenticated user is allowed to access or what operations they have permission to perform.
- **Auditing:** Periodically reviewing logs and other information to discover security issues related to user identity. This includes reviewing suspicious usage patterns, periodically reviewing user permissions to verify they're accurate, and other functions.

There is only one service trusted by Azure for identity, and that is Azure Active Directory (Azure AD). You'll be adding users to Azure AD and using it for all of the functions listed above. Before looking at how to configure Azure AD, it's important to understand the privileged accounts that are used to manage access to these services.

When your organization signed up for an Azure account, at least one Azure **account owner** was assigned. Also, an Azure AD **tenant** was created, unless an existing tenant was already associated with your organization's use of other Microsoft services such as Microsoft 365. A **global administrator** with full permissions on the Azure AD tenant was associated when it was created.

The user identities for both the Azure account owner and the Azure AD Global Administrator are stored in a highly secure identity system that is managed by Microsoft. The Azure account owner is authorized to create, update, and delete subscriptions. The Azure AD Global Administrator is authorized to perform many actions in Azure AD, but for this design guide you'll focus on the creation and deletion of user identity.

> **NOTE**
> Your organization may already have an existing Azure AD tenant if there's an existing Microsoft 365, Intune, or Dynamics 365 license associated with your account.

The Azure account owner has permission to create, update, and delete subscriptions:

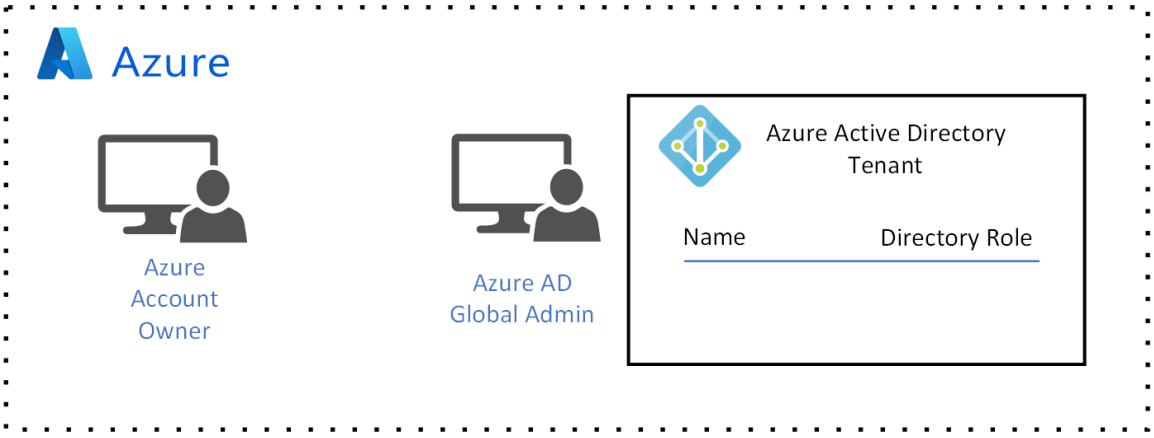


Figure 1: An Azure account with an Azure account owner and Azure AD global administrator.

The Azure AD **global administrator** has permission to create user accounts:

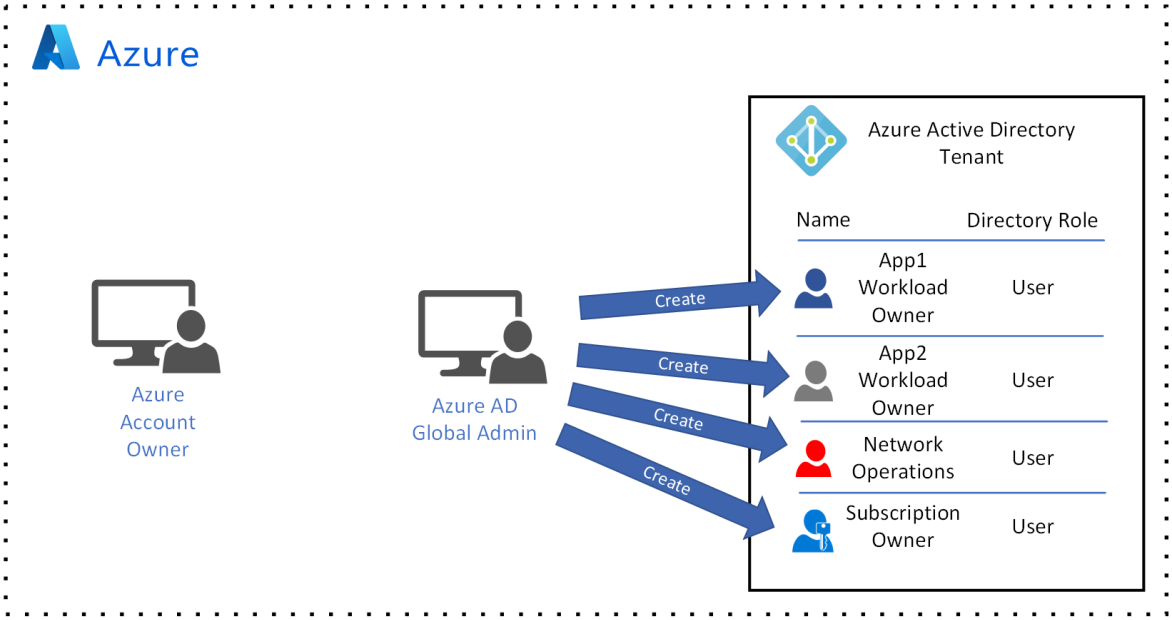


Figure 2: The Azure AD global administrator creates the required user accounts in the tenant.

The first two accounts, **app1 workload owner** and **app2 workload owner**, are each associated with an individual in your organization responsible for managing a workload. The **network operations** account is owned by the individual that is responsible for the shared infrastructure resources. Finally, the **subscription owner** account is associated with the individual responsible for ownership of subscriptions.

Resource access permissions model of least privilege

Now that your identity management system and user accounts have been created, you have to decide how to apply Azure roles to each account to support a permissions model of least privilege.

There's another requirement stating the resources associated with each workload be isolated from one another such that no one workload owner has management access to any other workload they do not own. There's also a requirement to implement this model using only built-in roles for Azure role-based access control.

Each Azure role is applied at one of three scopes in Azure: **subscription**, **resource group**, then an individual **resource**. Roles are inherited at lower scopes. For example, if a user is assigned the built-in Owner role at the subscription level, that role is also assigned to that user at the resource group and individual resource level unless overridden.

Therefore, to create a model of least-privilege access you have to decide the actions a particular type of user is allowed to take at each of these three scopes. For example, the requirement is for a workload owner to have permission to manage access to only the resources associated with their workload and no others. If you were to assign the built-in Owner role at the subscription scope, each workload owner would have management access to all workloads.

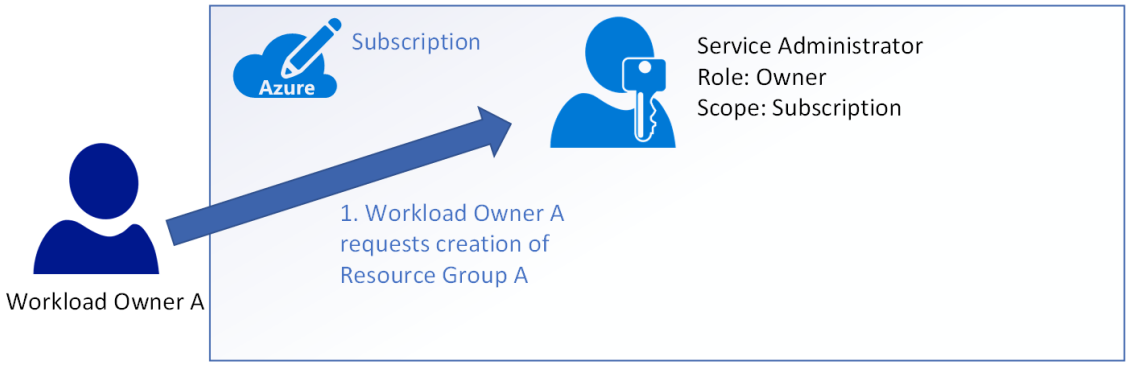
Let's take a look at two example permission models to understand this concept a little better. In the first example, the model trusts only the Service Administrator to create resource groups. In the second example, the model assigns the built-in Owner role to each workload owner at the subscription scope.

In both examples, there is a subscription Service Administrator that is assigned the built-in Owner role at the subscription scope. Recall that the built-in Owner role grants all permissions including the management of access to resources.

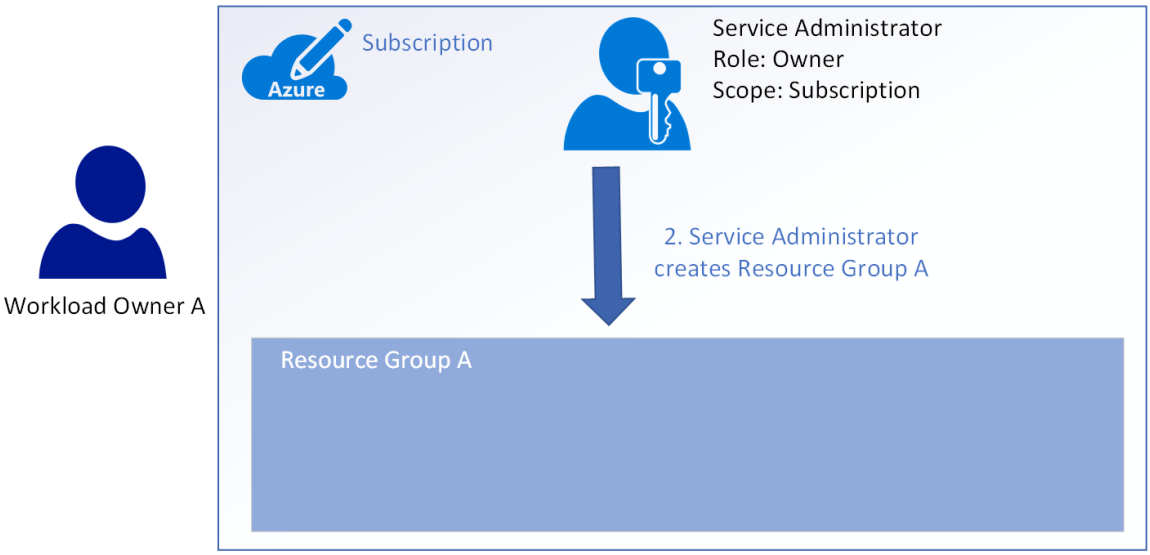


Figure 3: A subscription with a Service Administrator assigned the built-in Owner role.

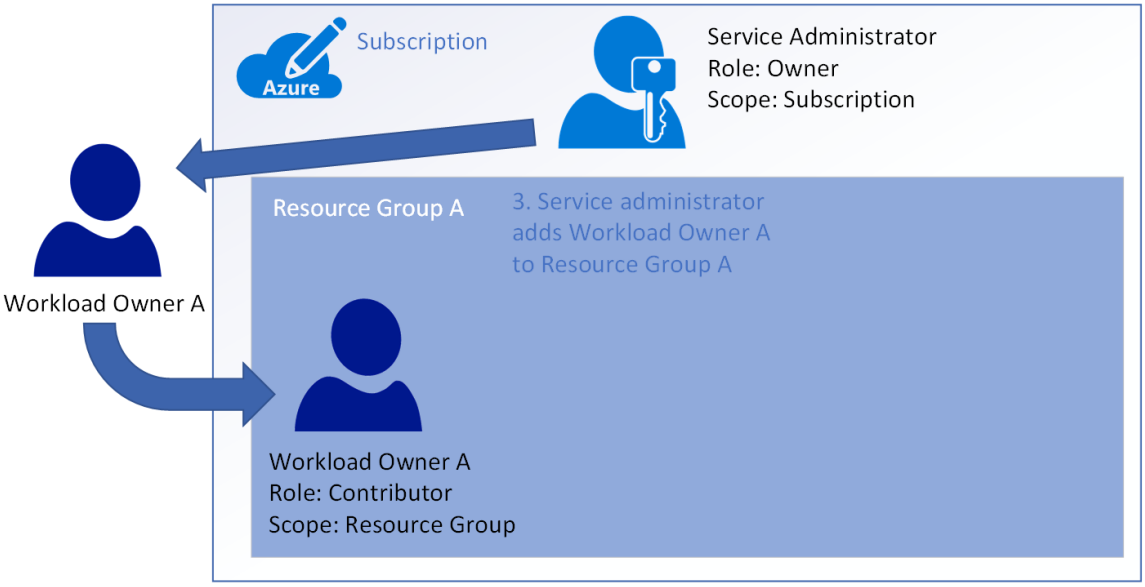
1. In the first example, **workload owner A** has no permissions at the subscription scope and no resource access management rights by default. This user wants to deploy and manage the resources for their workload. They must contact the **Service Administrator** to request creation of a resource group.



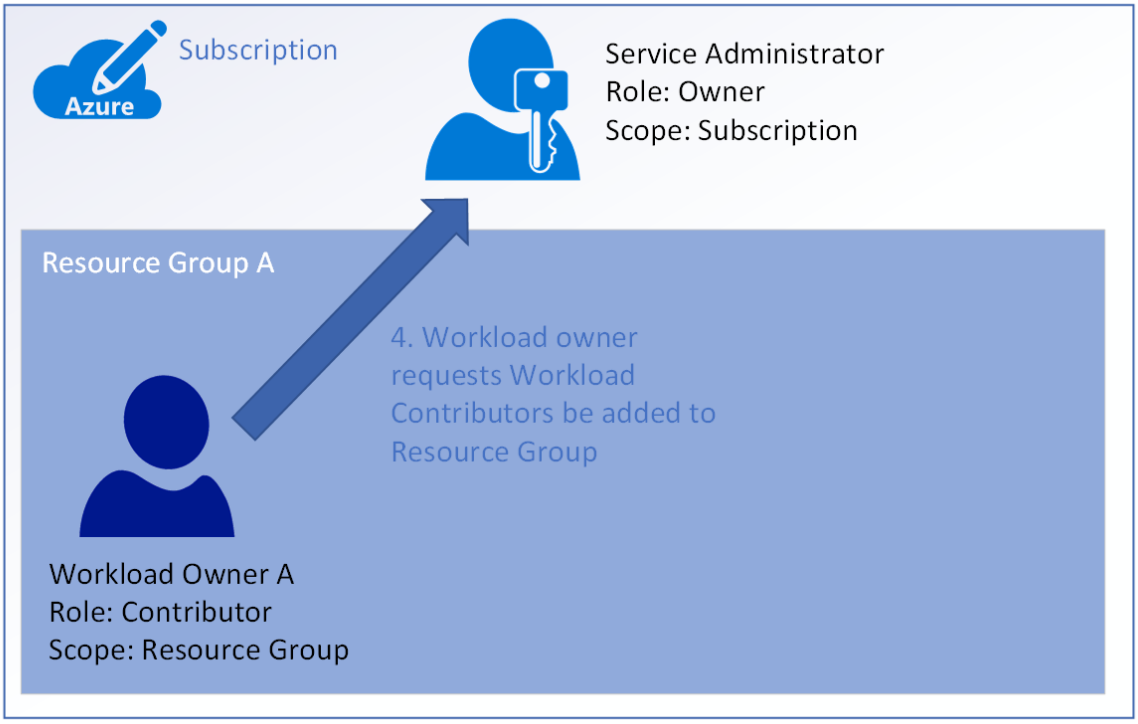
2. The **Service Administrator** reviews their request and creates **resource group A**. At this point, **workload owner A** still doesn't have permission to do anything.



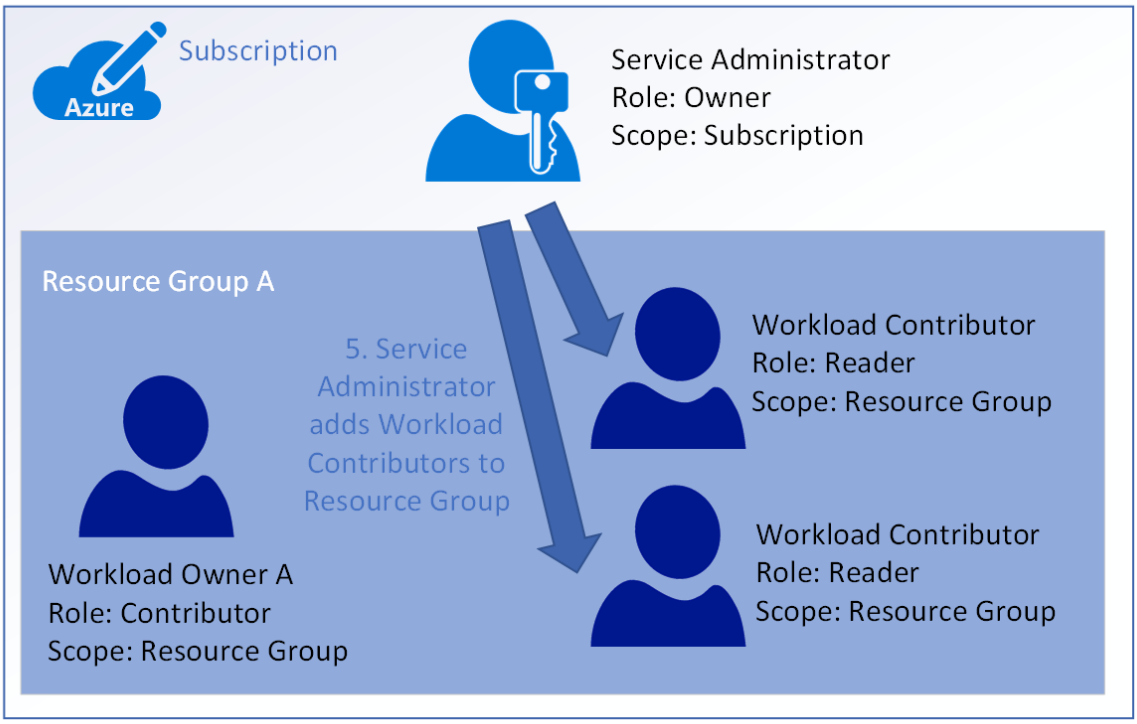
3. The **Service Administrator** adds **workload owner A** to **resource group A** and assigns the built-in Contributor role. The Contributor role grants all permissions on **resource group A** except managing access permission.



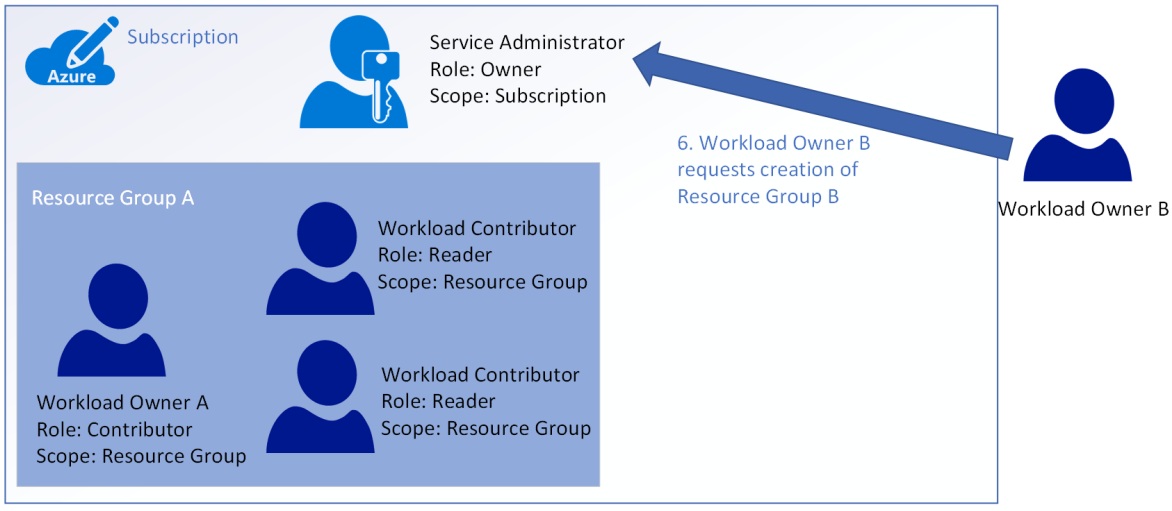
4. Let's assume that **workload owner A** has a requirement for a pair of team members to view the CPU and network traffic monitoring data as part of capacity planning for the workload. Because **workload owner A** is assigned the Contributor role, they do not have permission to add a user to **resource group A**. They must send this request to the **Service Administrator**.



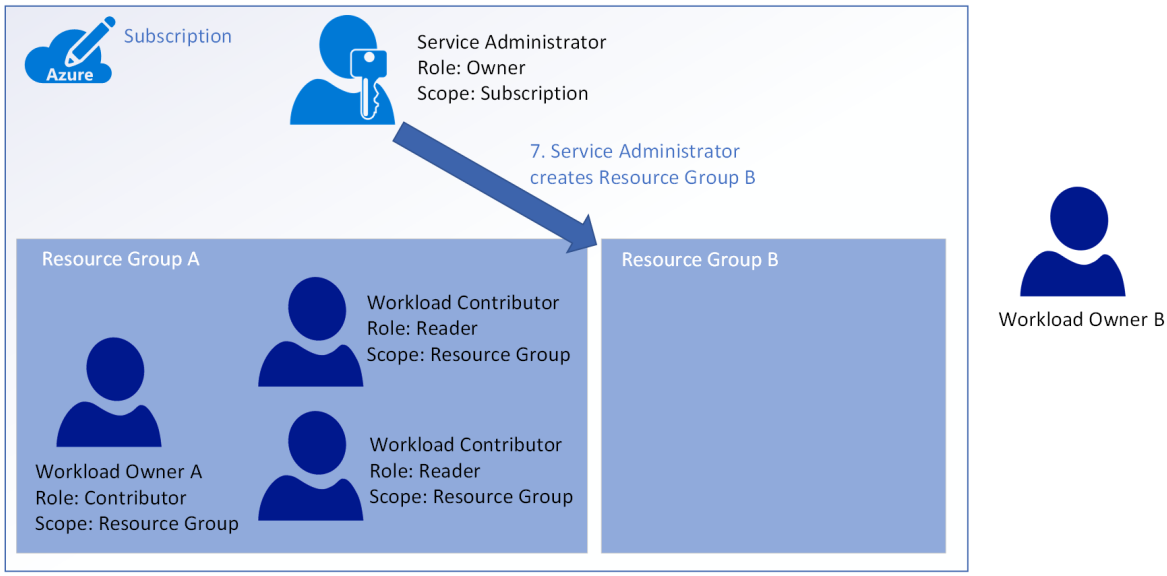
5. The **Service Administrator** reviews the request, and adds the two **workload contributor** users to **resource group A**. Neither of these two users require permission to manage resources, so they're assigned the built-in Reader role.



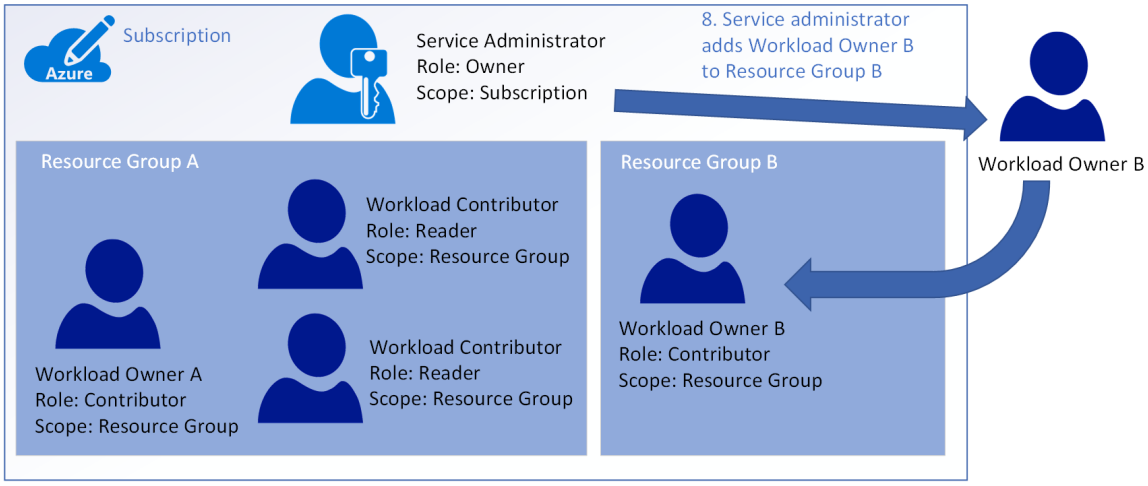
6. Next, **workload owner B** also requires a resource group to contain the resources for their workload. As with **workload owner A**, **workload owner B** initially does not have permission to take any action at the subscription scope so they must send a request to the **Service Administrator**.



7. The **Service Administrator** reviews the request and creates **resource group B**.



8. The **Service Administrator** then adds **workload owner B** to **resource group B** and assigns the built-in Contributor role.



At this point, each of the workload owners is isolated in their own resource group. None of the workload owners or their team members have management access to the resources in any other resource group.

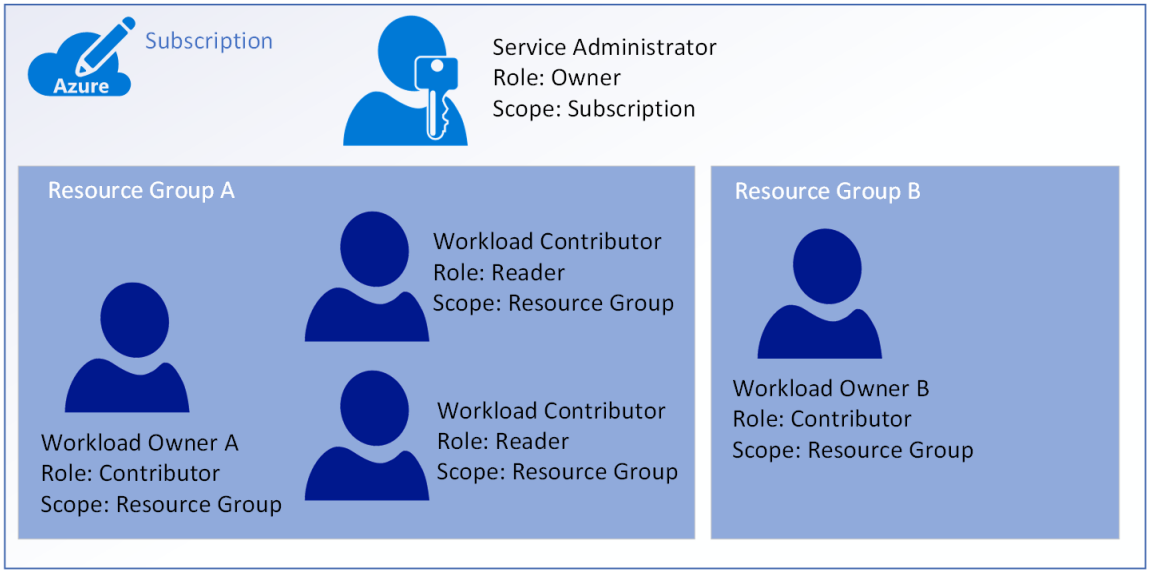


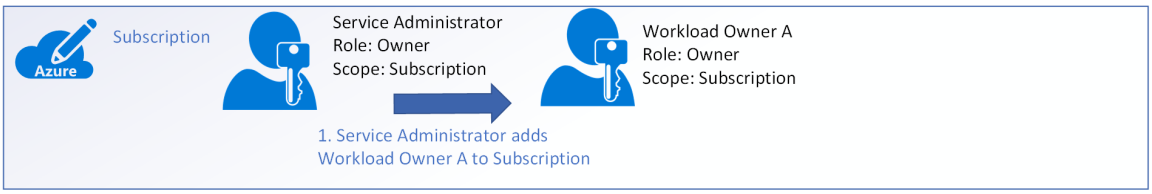
Figure 4: a subscription with two workload owners isolated with their own resource group.

This model is a least-privilege model. Each user is assigned the correct permission at the correct resource management scope.

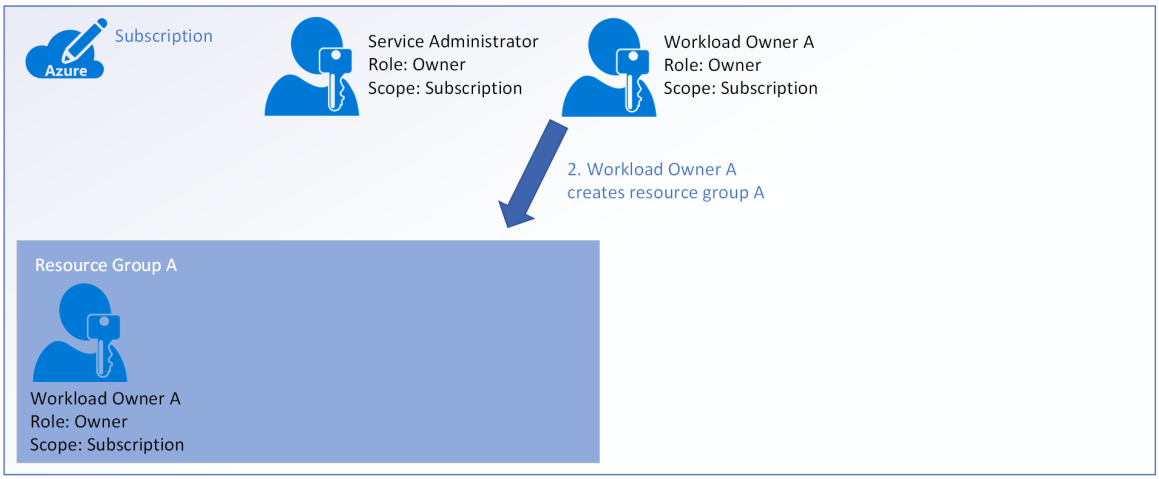
Consider that every task in this example was performed by the **Service Administrator**. While this is a simple example and may not appear to be an issue because there were only two workload owners, it's easy to imagine the types of issues that would result for a large organization. For example, the **Service Administrator** can become a bottleneck with a large backlog of requests that result in delays.

Let's take a look at second example that reduces the number of tasks performed by the **Service Administrator**.

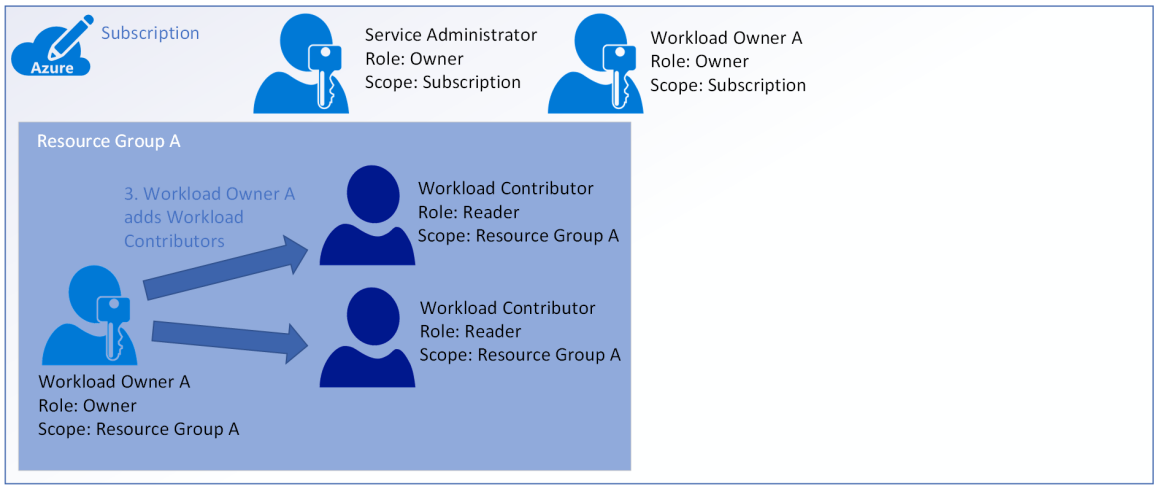
1. In this model, **workload owner A** is assigned the built-in Owner role at the subscription scope, enabling them to create their own resource group: **resource group A**.



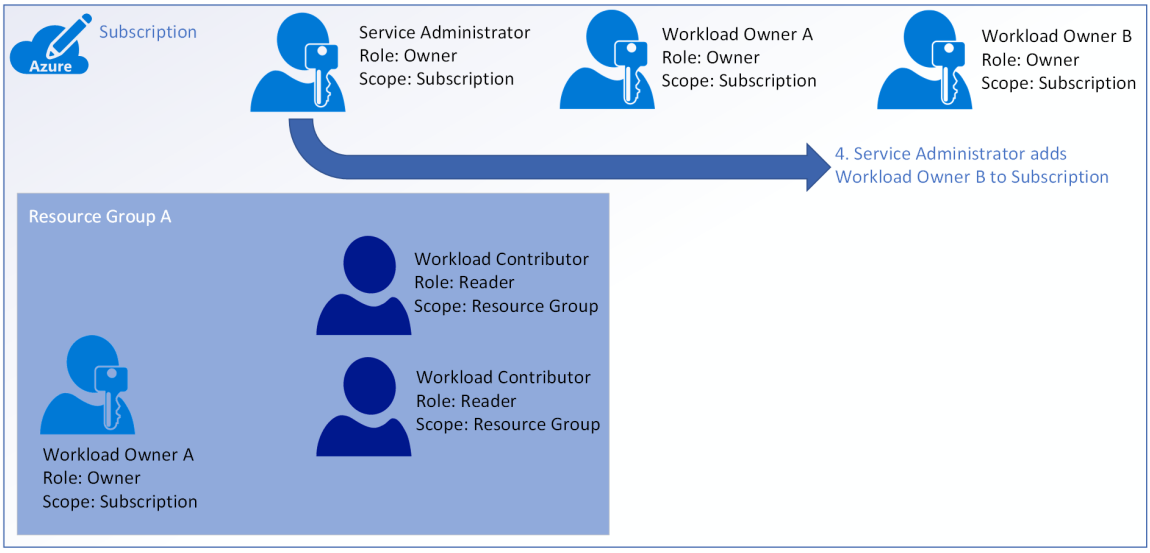
2. When **resource group A** is created, **workload owner A** is added by default and inherits the built-in Owner role from the subscription scope.



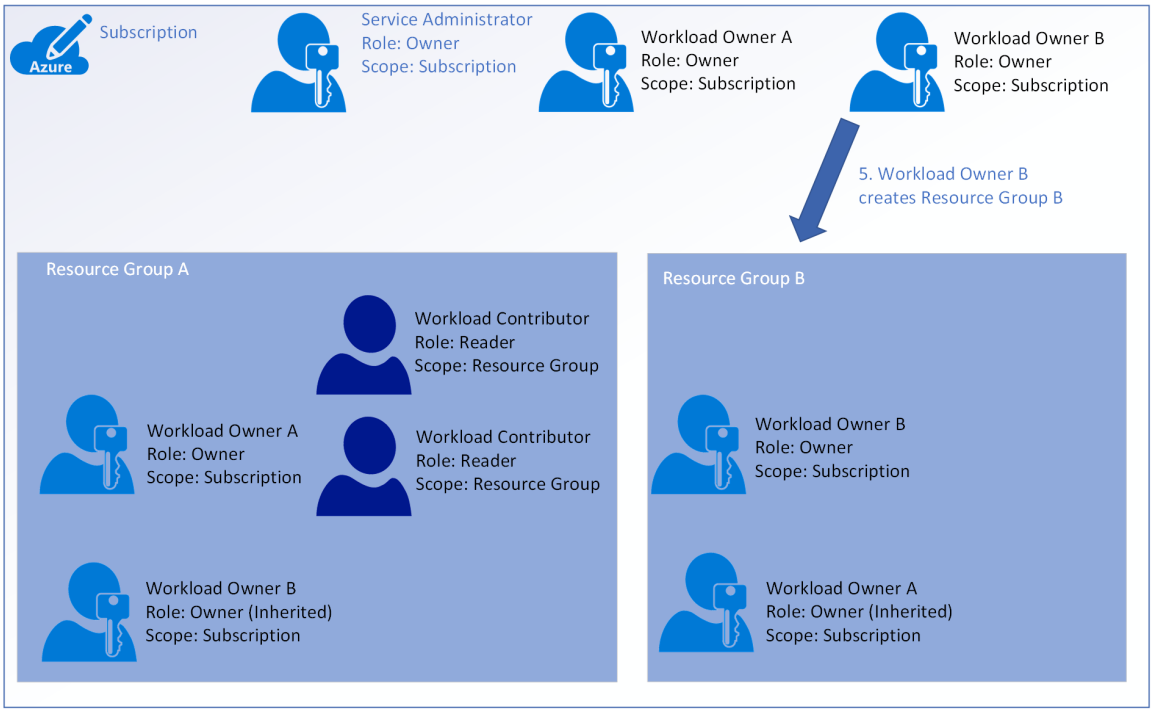
3. The built-in Owner role grants **workload owner A** permission to manage access to the resource group. **Workload owner A** adds two **workload contributors** and assigns the built-in Reader role to each of them.



4. The **Service Administrator** now adds **workload owner B** to the subscription with the built-in Owner role.



5. **Workload owner B** creates **resource group B** and is added by default. Again, **workload owner B** inherits the built-in Owner role from the subscription scope.



Note that in this model, the **Service Administrator** performed fewer actions than they did in the first example due to the delegation of management access to each of the individual workload owners.

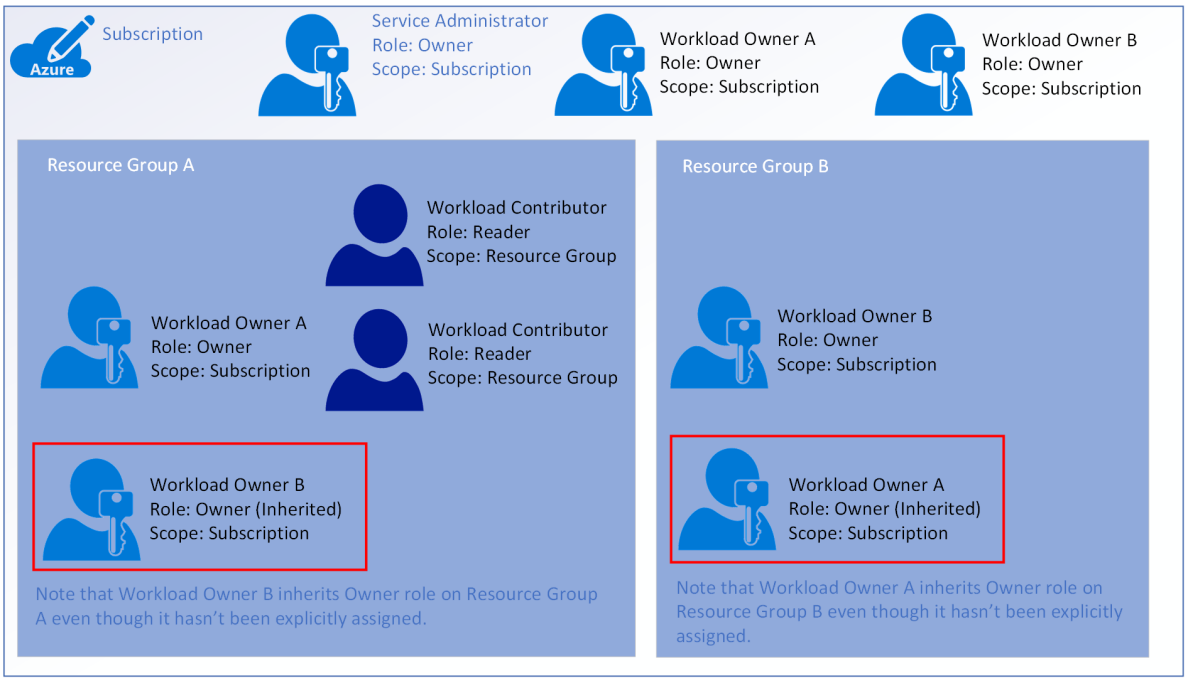


Figure 5: A subscription with a Service Administrator and two workload owners, all assigned the built-in Owner role.

Because both **workload owner A** and **workload owner B** are assigned the built-in Owner role at the subscription scope, they have each inherited the built-in Owner role for each other's resource group. This means that not only do they have full access to each other's resources, they can also delegate management access to each other's resource groups. For example, **workload owner B** has rights to add any other user to **resource group A** and can assign any role to them, including the built-in Owner role.

If you compare each example to the requirements, you'll see that both examples support a single trusted user at the subscription scope with permission to grant resource access rights to the two workload owners. Each of the two workload owners did not have access to resource management by default and required the **Service Administrator** to explicitly assign permissions to them. Only the first example supports the requirement that the resources associated with each workload are isolated from one another such that no workload owner has access to the resources of any other workload.

Resource management model

Now that you've designed a permissions model of least privilege, let's move on to take a look at some practical applications of these governance models. Recall from the requirements that you must support the following three environments:

1. **Shared infrastructure environment:** A group of resources shared by all workloads. These are resources such as network gateways, firewalls, and security services.
2. **Production environment:** Multiple groups of resources representing multiple production workloads. These resources are used to host the private and public-facing application artifacts. These resources typically have the tightest governance and security models to protect the resources, application code, and data from unauthorized access.
3. **Preproduction environment:** Multiple groups of resources representing multiple workloads not yet ready for production. These resources are used for development and testing, and may have a more relaxed governance model to enable increased developer agility. Security within these groups should increase as the application development process moves closer to production.

For each of these three environments, there is a requirement to track cost data by **workload owner**, **environment**, or both. That is, you'll want to know the ongoing cost of the **shared infrastructure**, the costs

incurred by individuals in both the **nonproduction** and **production** environments, and finally the overall cost of **nonproduction** and **production** environments.

You have already learned that resources are scoped to two levels: **subscription** and **resource group**. Therefore, the first decision is how to organize environments by **subscription**. There are only two possibilities: a single subscription or multiple subscriptions.

Before you look at examples of each of these models, let's review the management structure for subscriptions in Azure.

Recall from the requirements that you have an individual in the organization who is responsible for subscriptions, and this user owns the **subscription owner** account in the Azure AD tenant. This account does not have permission to create subscriptions. Only the **Azure account owner** has permission to do this:

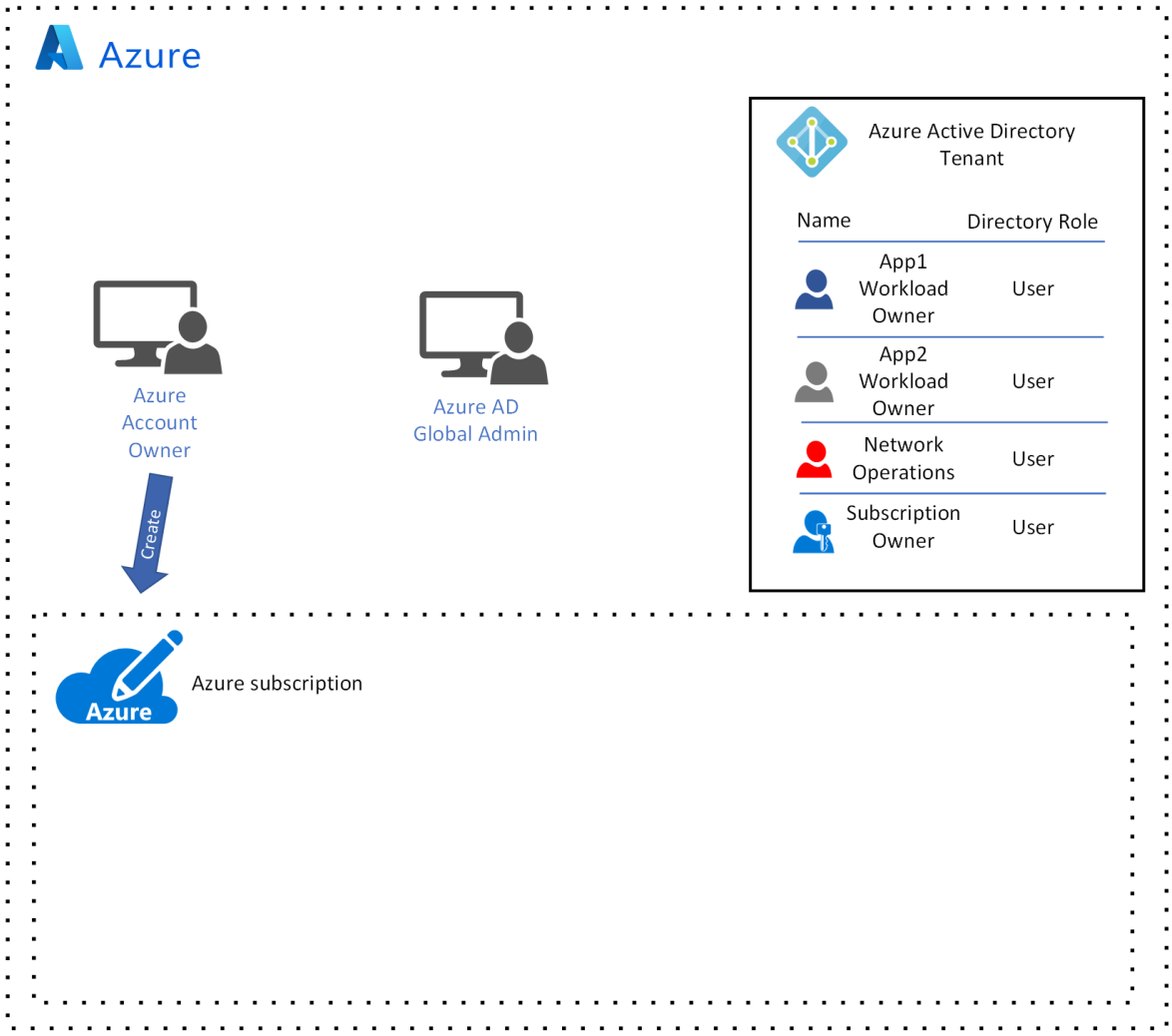
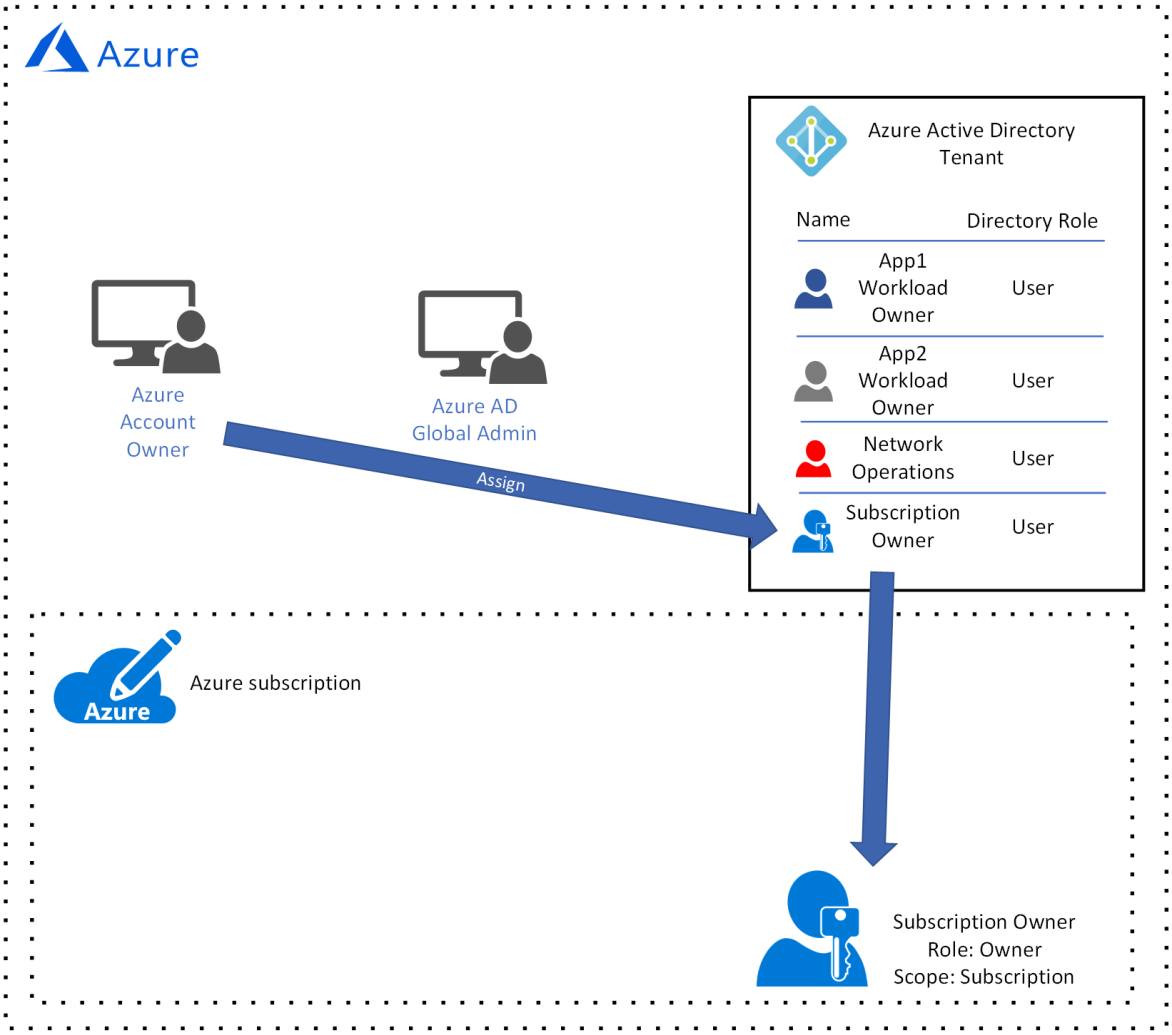


Figure 6: An Azure account owner creates a subscription.

Once the subscription has been created, the **Azure account owner** can add the **subscription owner** account to the subscription with the **owner** role:



. Figure 7: The Azure account owner adds the subscription owner user account to the subscription with the owner role.

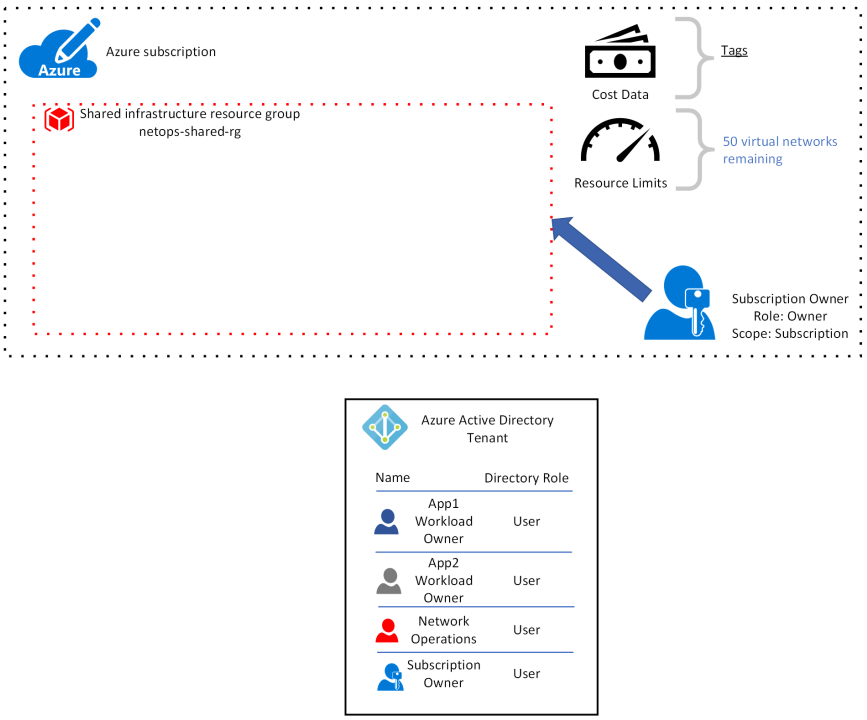
The **subscription owner** account can now create **resource groups** and delegate resource access management.

First let's look at an example resource management model using a single subscription. The first decision is how to align resource groups to the three environments. You have two options:

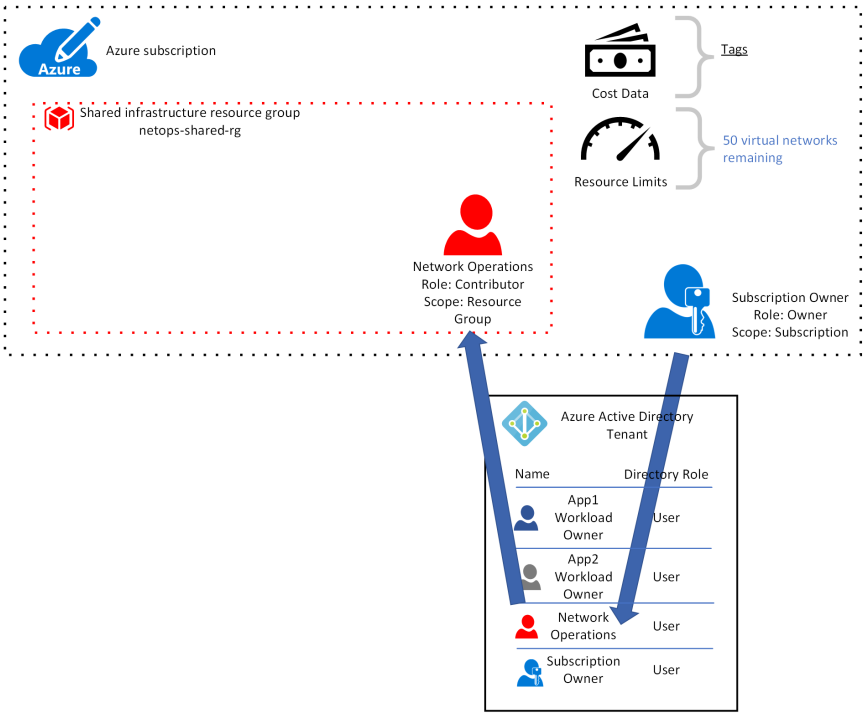
1. Align each environment to a single resource group. All shared infrastructure resources are deployed to a single **shared infrastructure** resource group. All resources associated with development workloads are deployed to a single **development** resource group. All resources associated with production workloads are deployed into a single **production** resource group for the **production** environment.
2. Create separate resource groups for each workload, using a naming convention and tags to align resource groups with each of the three environments.

Let's begin by evaluating the first option. You'll be using the permissions model that was discussed in the previous section, with a single subscription Service Administrator who creates resource groups and adds users to them with either the built-in **Contributor** or **Reader** role.

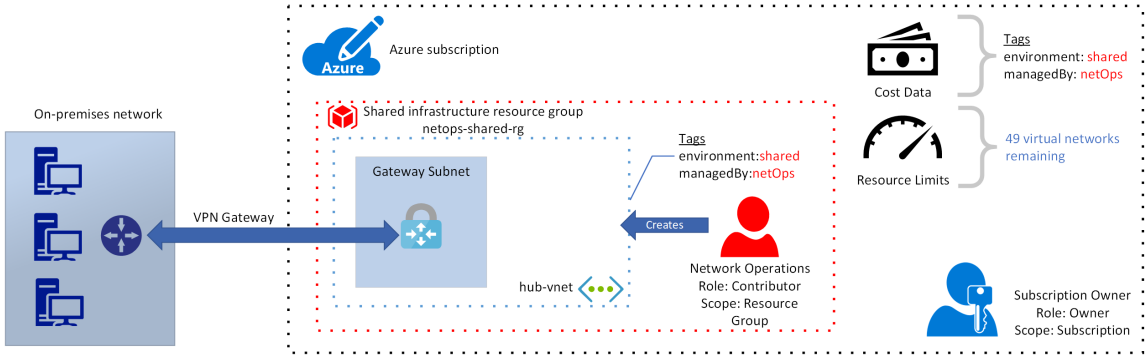
1. The first resource group deployed represents the **shared infrastructure** environment. The **subscription owner** account creates a resource group for the shared infrastructure resources named `netops-shared-rg`.



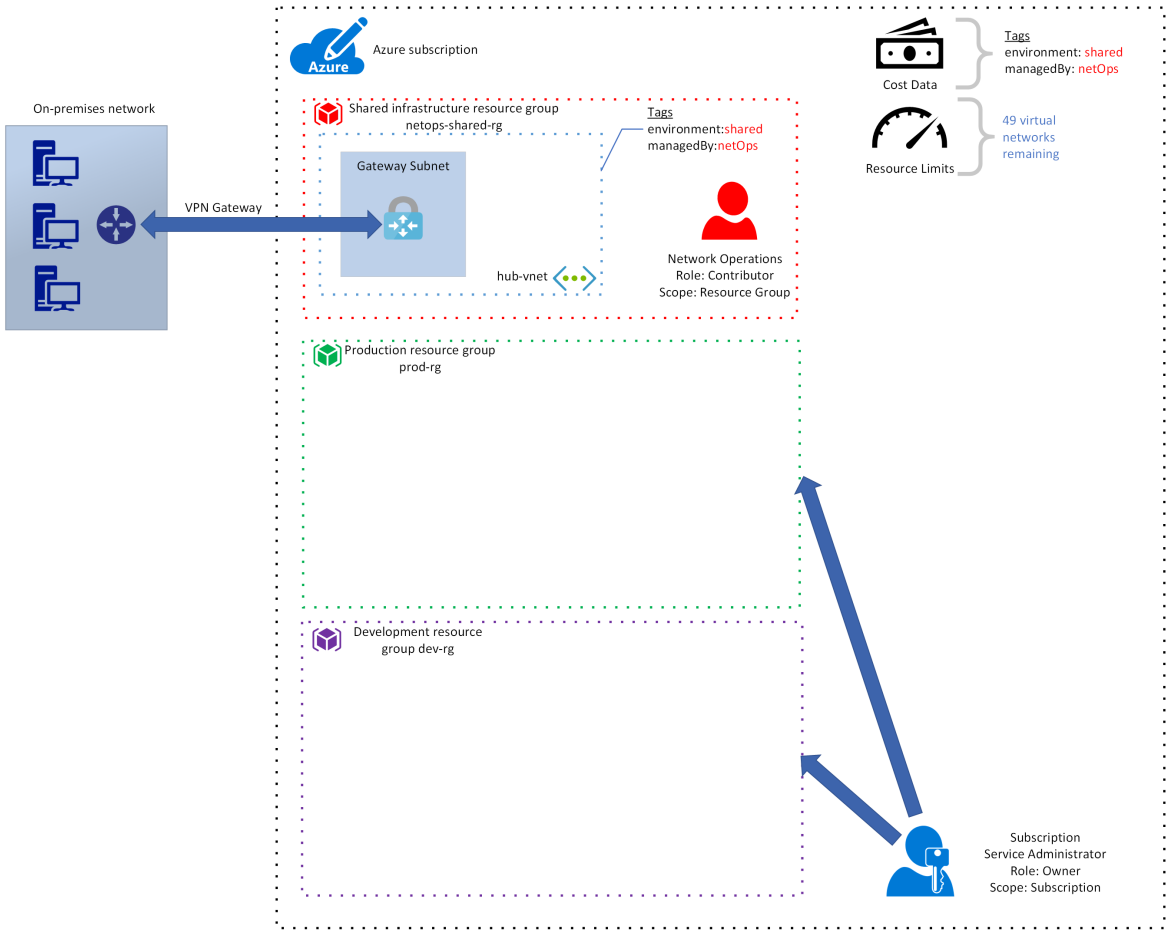
2. The **subscription owner** account adds the **network operations user** account to the resource group and assigns the **Contributor** role.



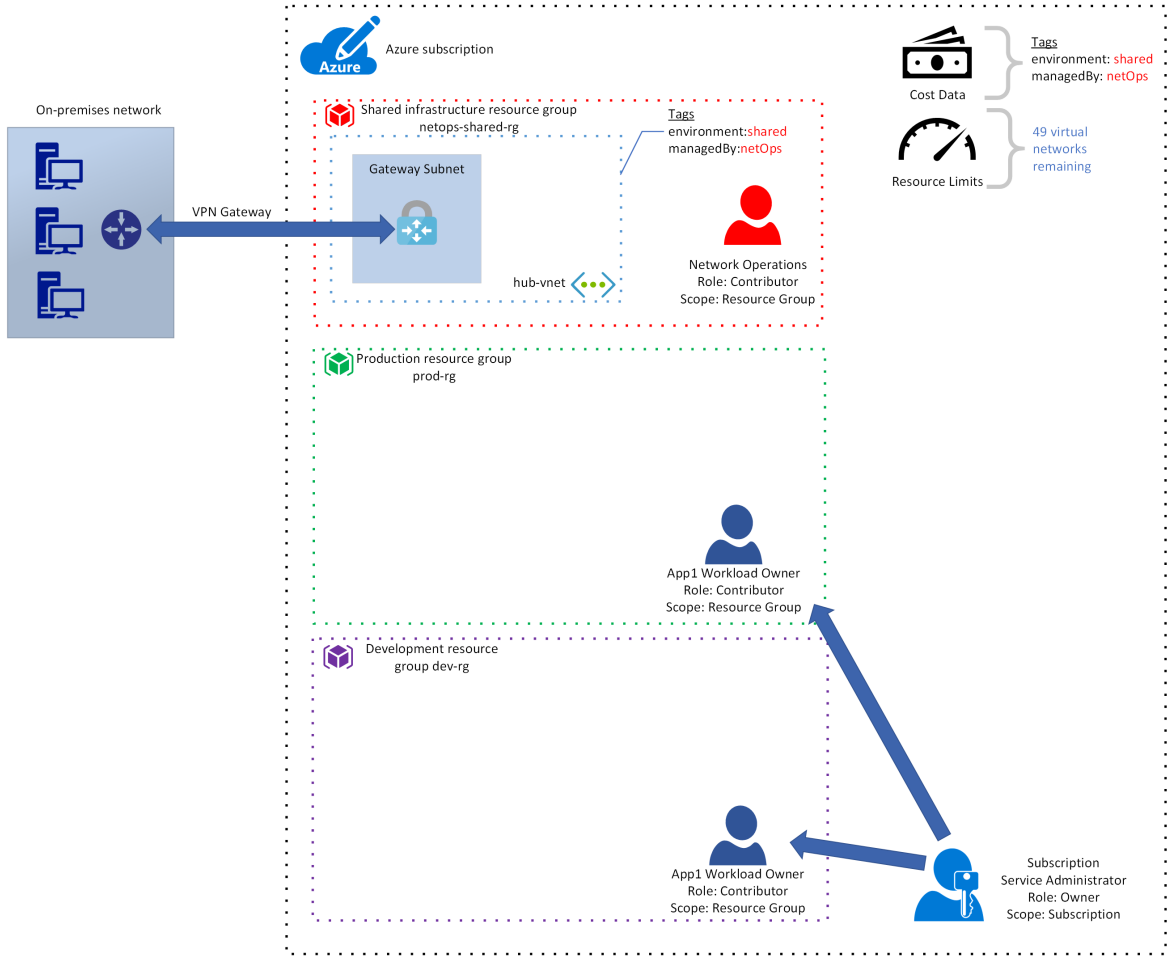
3. The **network operations user** creates a VPN gateway and configures it to connect to the on-premises VPN appliance. The **network operations user** also applies a pair of tags to each of the resources: `environment:shared` and `managedBy:netops`. When the **subscription Service Administrator** exports a cost report, costs will be aligned with each of these tags. This allows the **subscription Service Administrator** to pivot costs using the `environment` tag and the `managedBy` tag. Notice the **resource limits** counter at the top right-hand side of the figure. Each Azure subscription has service limits, and to help you understand the effect of these limits you'll follow the virtual network limit for each subscription. There is a limit of 1,000 virtual networks per subscription, and after the first virtual network is deployed there are now 999 available.



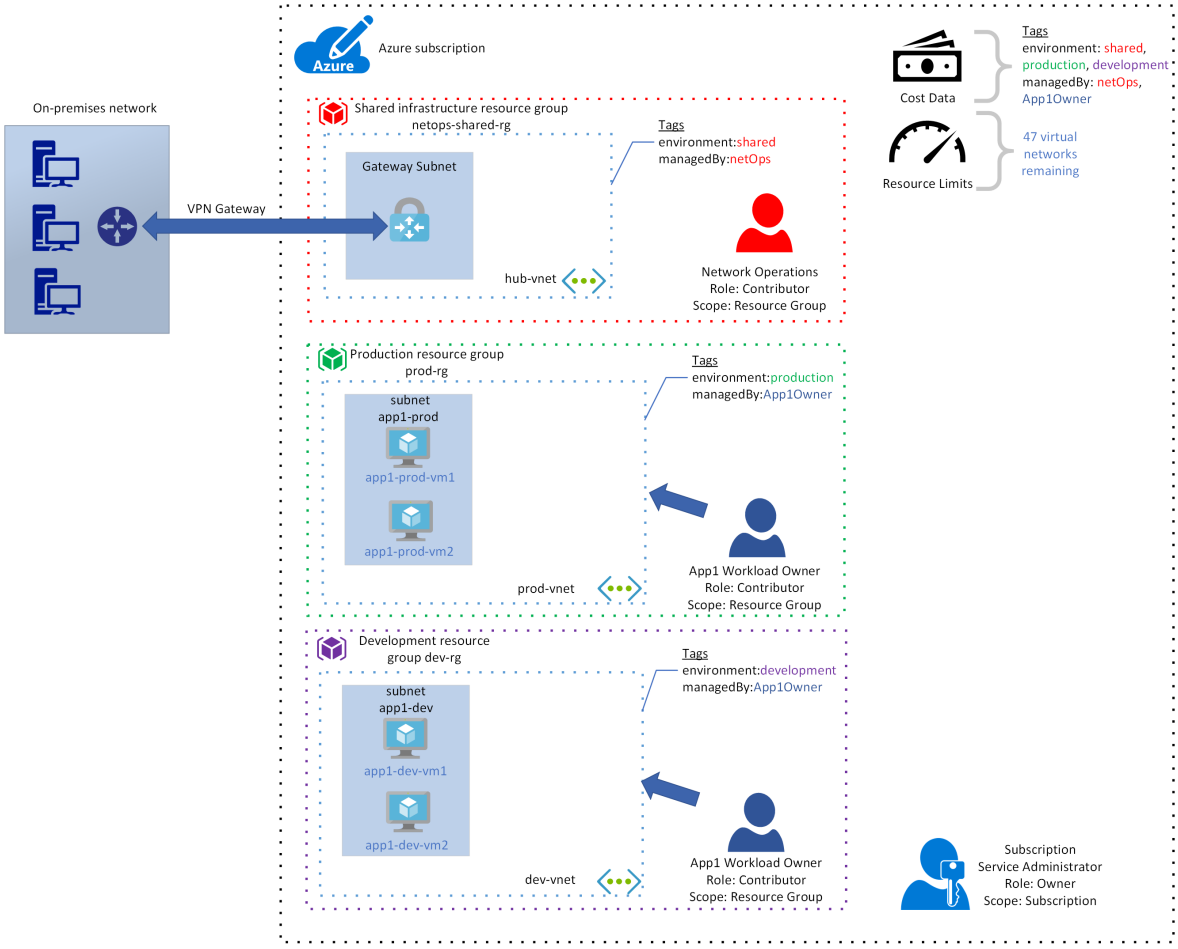
4. Two more resource groups are deployed. The first is named `prod-rg`. This resource group is aligned with the production environment. The second is named `dev-rg` and is aligned with the development environment. All resources associated with production workloads are deployed to the production environment and all resources associated with development workloads are deployed to the development environment. In this example, you'll only deploy two workloads to each of these two environments, so you won't encounter any Azure subscription service limits. Consider that each resource group has a limit of 800 resources per resource group. If you continue to add workloads to each resource group, you'll eventually reach this limit.



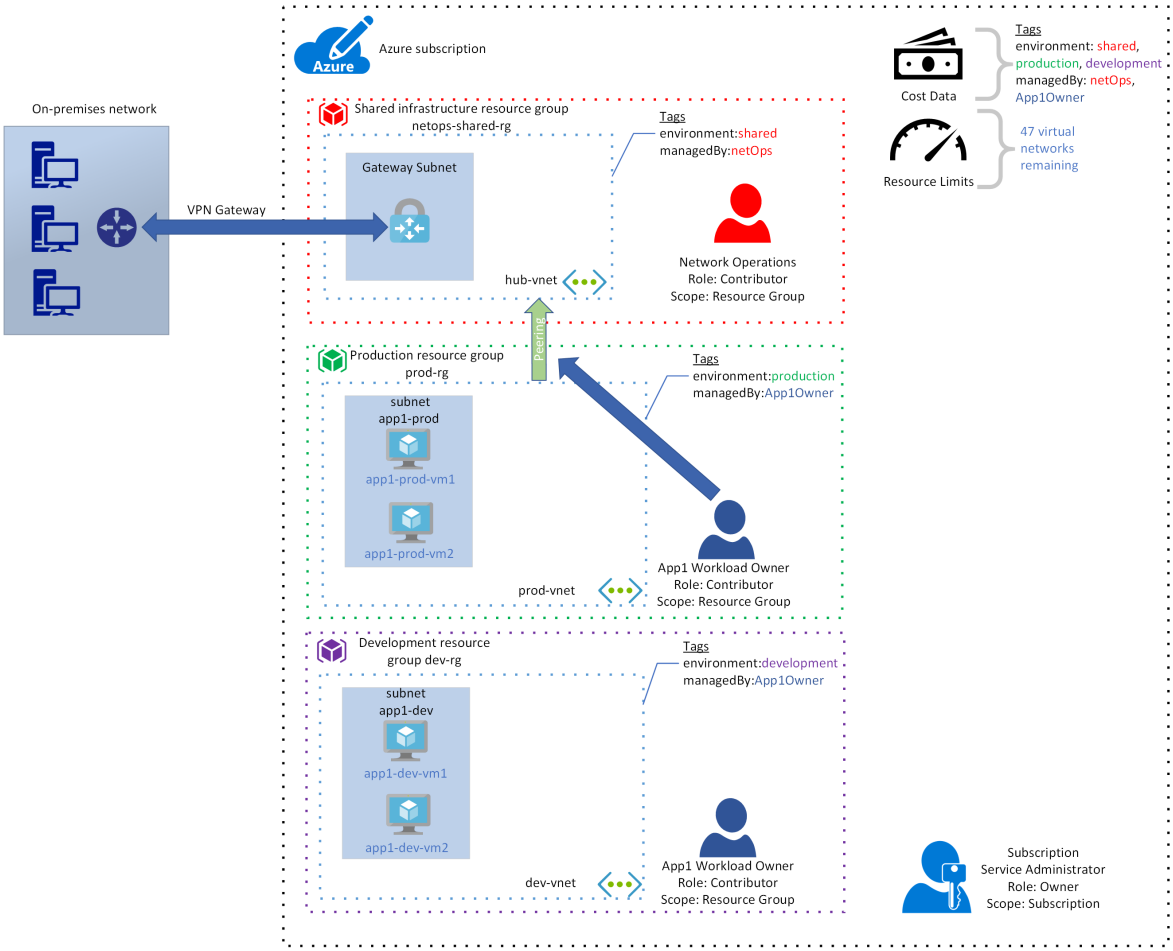
5. The first **workload owner** sends a request to the **subscription Service Administrator** and is added to each of the development and production environment resource groups with the **Contributor** role. As you learned earlier, the **Contributor** role allows the user to perform any operation other than assigning a role to another user. The first **workload owner** can now create the resources associated with their workload.



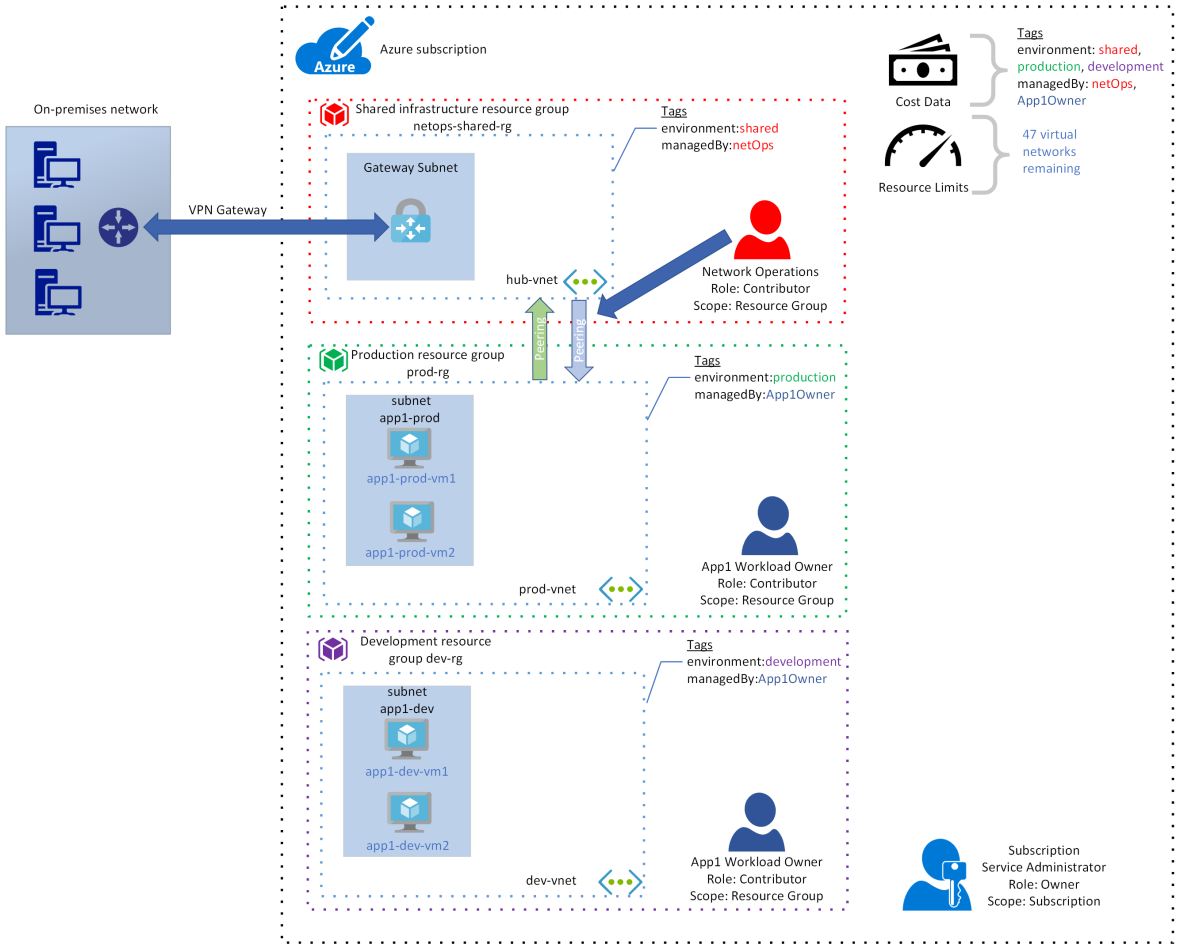
6. The first **workload owner** creates a virtual network in each of the two resource groups with a pair of virtual machines in each. The first **workload owner** applies the `environment` and `managedBy` tags to all resources. Note that the Azure service limit counter is now at 997 virtual networks remaining.



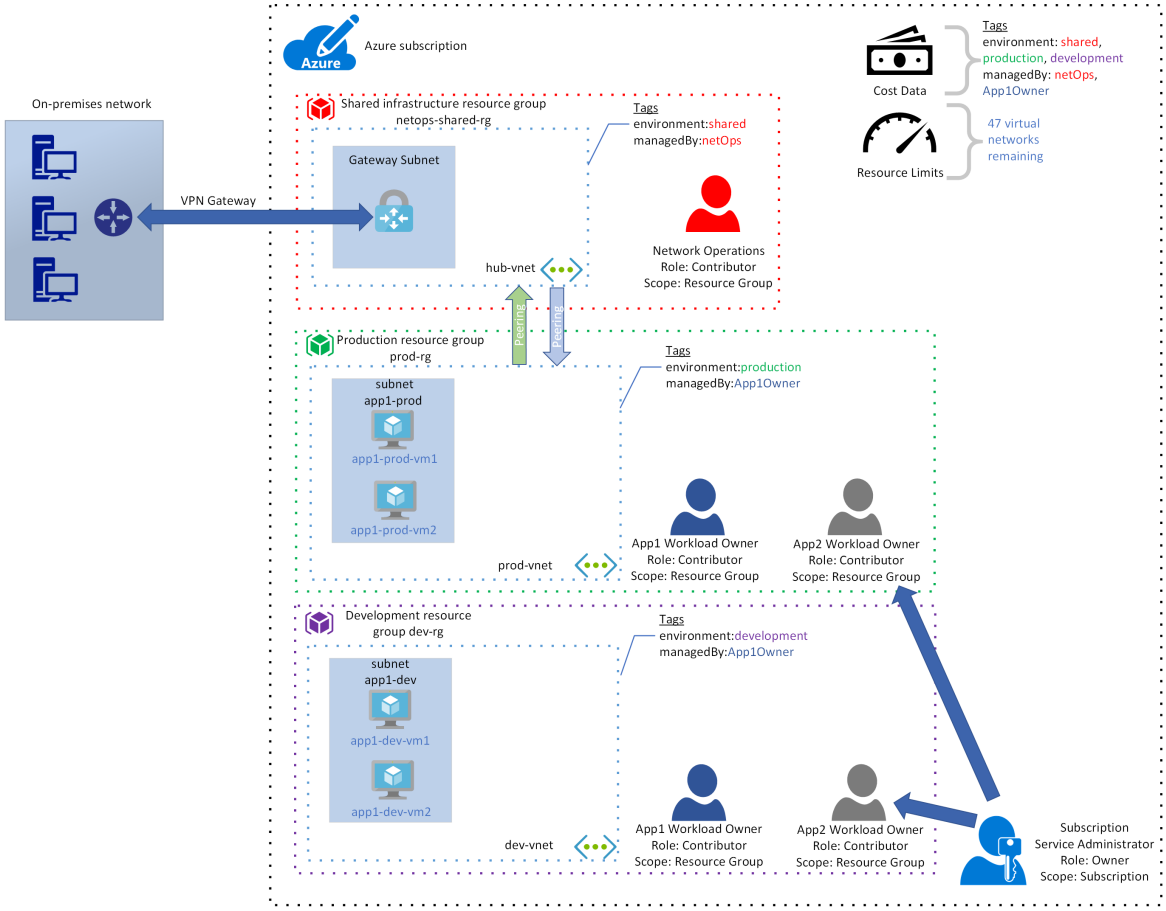
7. None of the virtual networks has connectivity to on-premises when created. In this type of architecture, each virtual network must be peered to the `hub-vnet` in the **shared infrastructure** environment. Virtual network peering creates a connection between two separate virtual networks and allows network traffic to travel between them. Note that virtual network peering is not inherently transitive. A peering must be specified in each of the two virtual networks that are connected, and if only one of the virtual networks specifies a peering, then the connection is incomplete. To illustrate the effect of this, the first **workload owner** specifies a peering between `prod-vnet` and `hub-vnet`. The first peering is created, but no traffic flows because the complementary peering from `hub-vnet` to `prod-vnet` has not yet been specified. The first **workload owner** contacts the **network operations** user and requests this complementary peering connection.



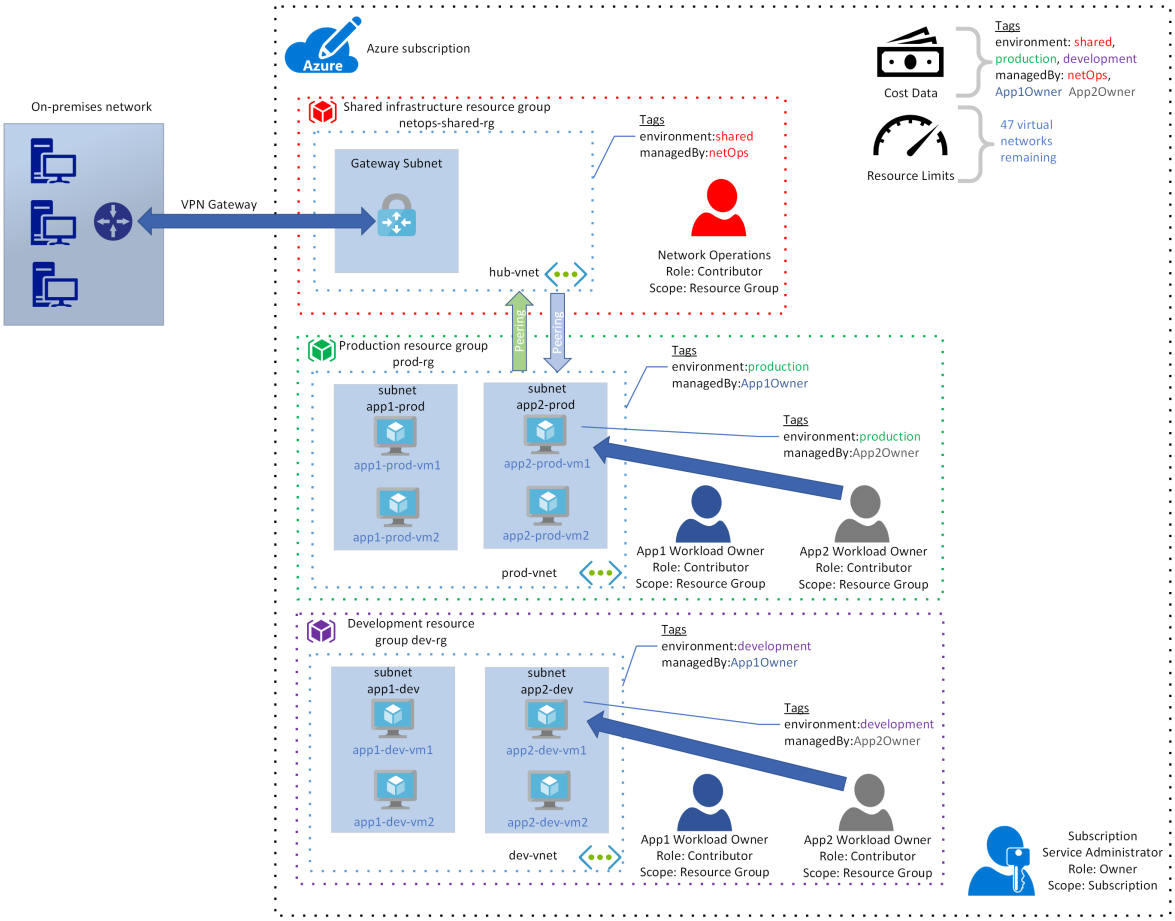
8. The **network operations** user reviews the request, approves it, then specifies the peering in the settings for the `hub-vnet`. The peering connection is now complete, and network traffic flows between the two virtual networks.



9. Now, a second **workload owner** sends a request to the **subscription Service Administrator** and is added to the existing **production** and **development** environment resource groups with the **Contributor** role. The second **workload owner** has the same permissions on all resources as the first **workload owner** in each resource group.



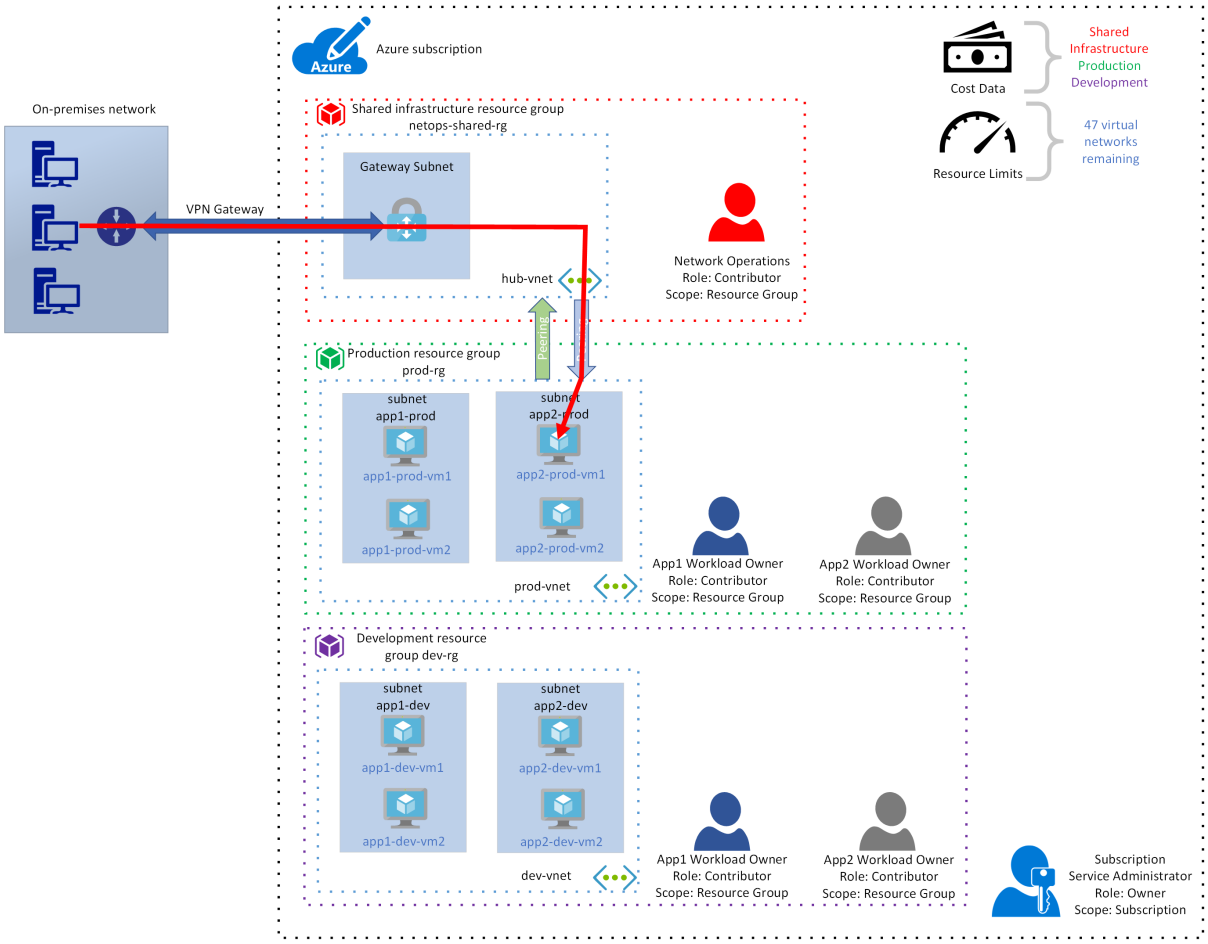
10. The second **workload owner** creates a subnet in the `prod-vnet` virtual network, then adds two virtual machines. The second **workload owner** applies the `environment` and `managedBy` tags to each resource.



This example resource management model enables us to manage resources in the three required environments.

The shared infrastructure resources are protected because only a single user in the subscription has permission to access those resources. Each of the workload owners can use the shared infrastructure resources without having any permissions on the shared resources themselves. This management model fails the requirement for workload isolation, because both **workload owners** can access the resources of each other's workload.

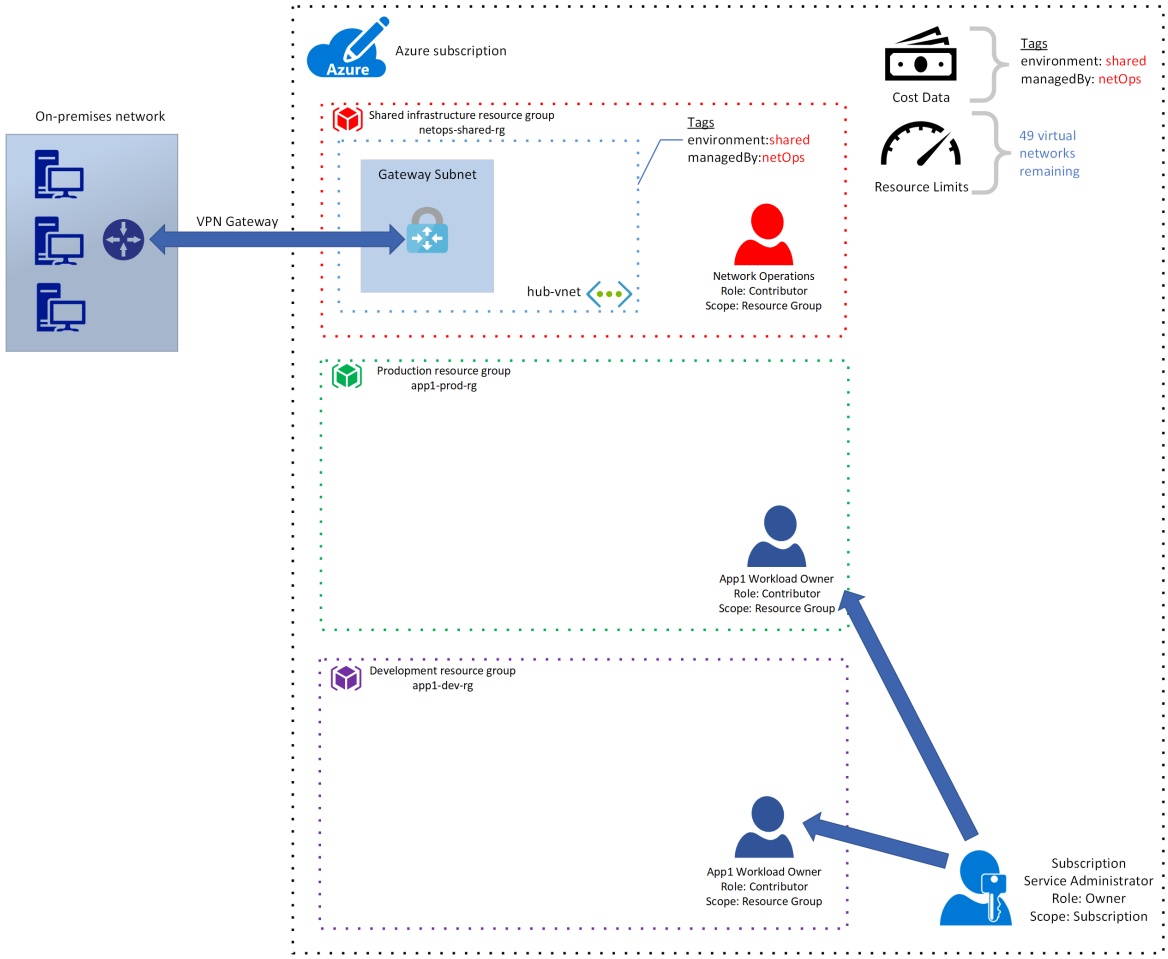
There's another important consideration with this model that may not be immediately obvious. In the example, it was **app1 workload owner** that requested the network peering connection with the `hub-vnet` to provide connectivity to the on-premises network. The **network operations** user evaluated that request based on the resources deployed with that workload. When the **subscription owner** account added **app2 workload owner** with the **Contributor** role, that user had management access rights to all resources in the `prod-rg` resource group.



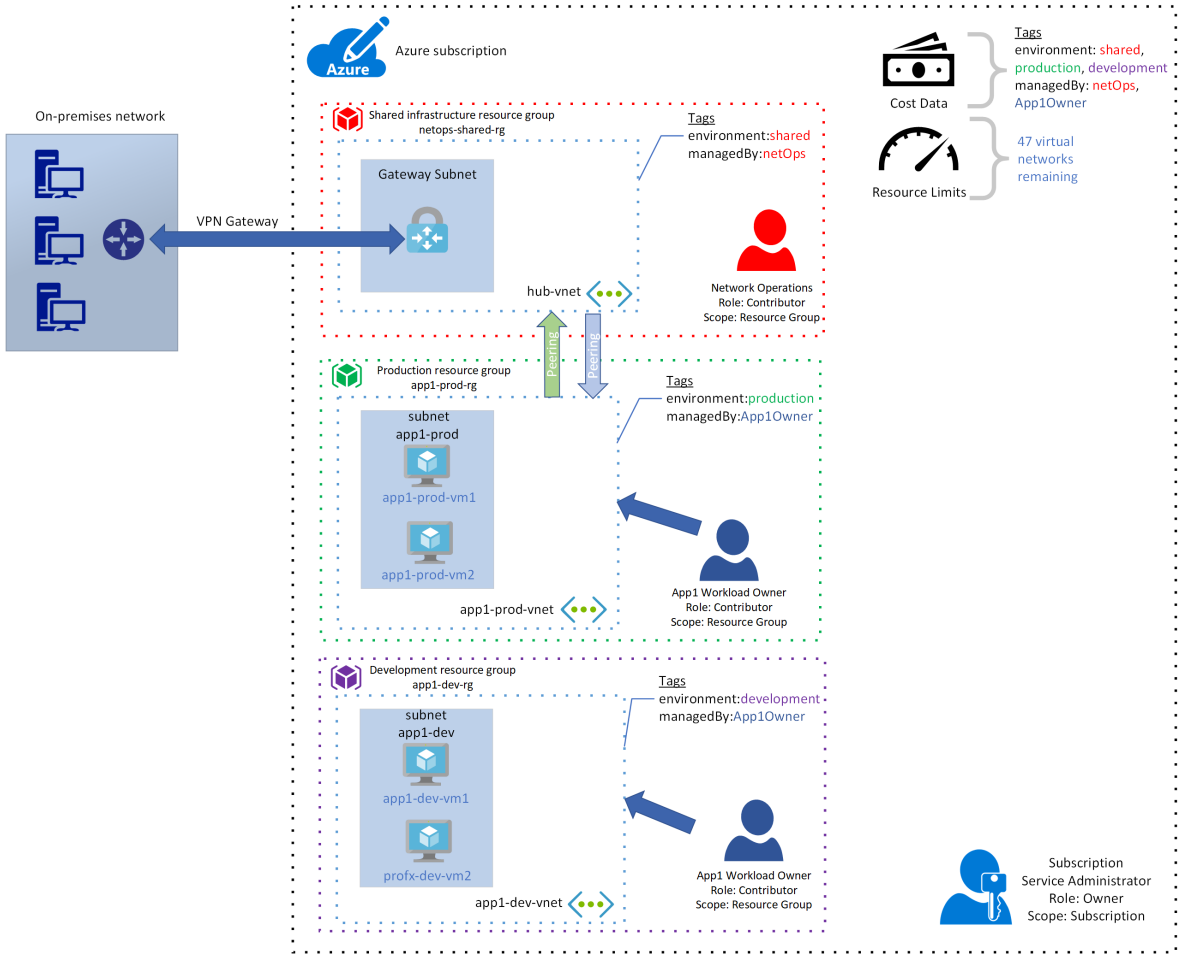
This means **app2 workload owner** had permission to deploy their own subnet with virtual machines in the `prod-vnet` virtual network. By default, those virtual machines have access to the on-premises network. The **network operations** user is not aware of those machines and did not approve their connectivity to on-premises.

Next, let's look at a single subscription with multiple resource groups for different environments and workloads. Note that in the previous example, the resources for each environment were easily identifiable because they were in the same resource group. Now that you no longer have that grouping, you will have to rely on a resource group naming convention to provide that functionality.

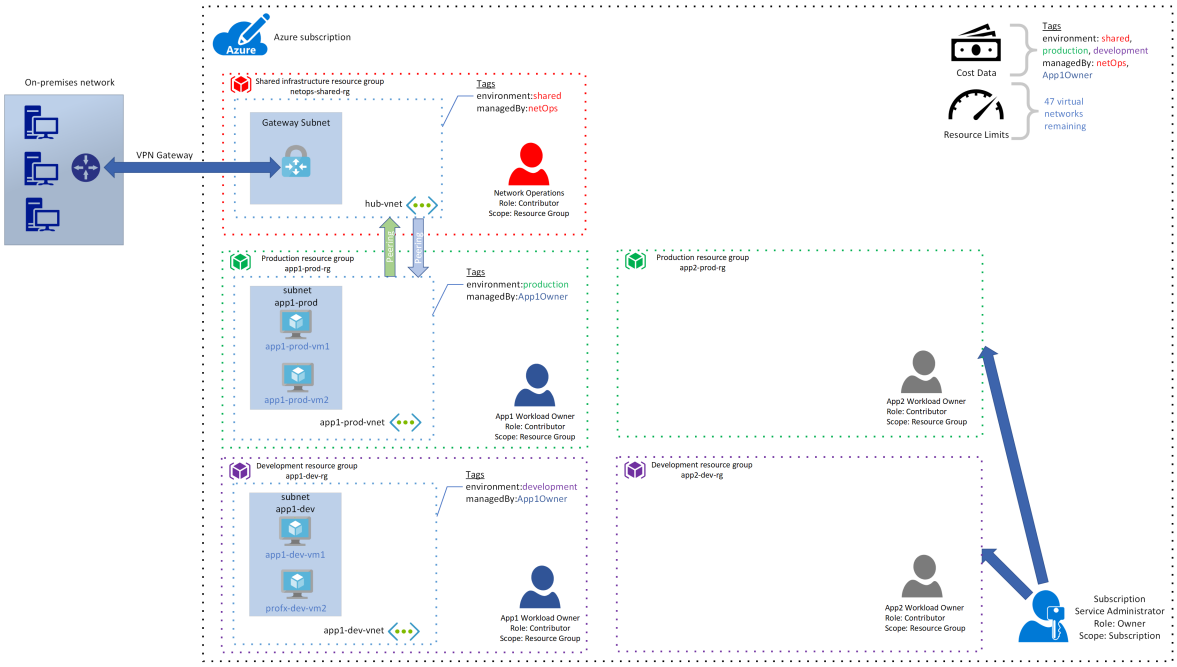
1. The **shared infrastructure** resources will still have a separate resource group in this model, so that remains the same. Each workload requires two resource groups, one for each of the **development** and **production** environments. For the first workload, the **subscription owner** account creates two resource groups. The first is named `app1-prod-rg` and the second is named `app1-dev-rg`. As discussed earlier, this naming convention identifies the resources as being associated with the first workload, `app1`, and either the **development** or **production** environment. Again, the **subscription owner** account adds **app1 workload owner** to the resource group with the **Contributor** role.



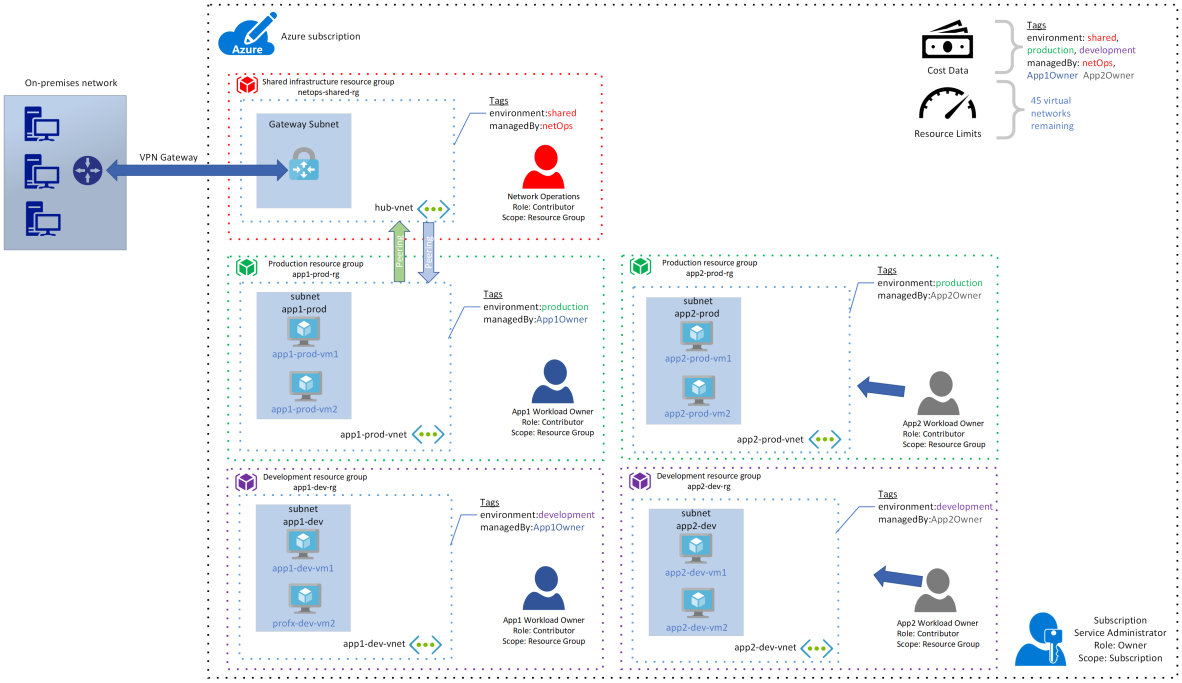
2. Similar to the first example, **app1 workload owner** deploys a virtual network named `app1-prod-vnet` to the **production** environment, and another named `app1-dev-vnet` to the **development** environment. Again, **app1 workload owner** sends a request to the **network operations** user to create a peering connection. Note that **app1 workload owner** adds the same tags as in the first example, and the limit counter has been decremented to 997 virtual networks remaining in the subscription.



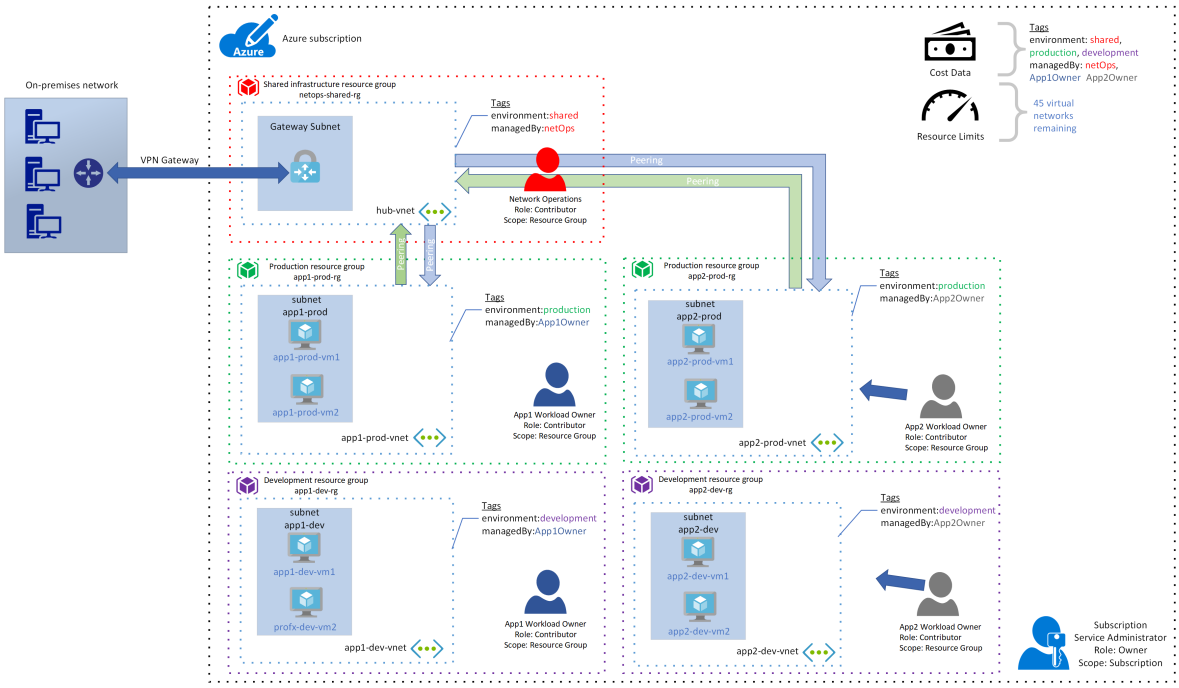
3. The **subscription owner** account now creates two resource groups for **app2 workload owner**. Following the same conventions as for **app1 workload owner**, the resource groups are named `app2-prod-rg` and `app2-dev-rg`. The **subscription owner** account adds **app2 workload owner** to each of the resource groups with the **Contributor** role.



4. The **app2 workload owner** account deploys virtual networks and virtual machines to the resource groups with the same naming conventions. Tags are added and the limit counter has been decremented to 995 virtual networks remaining in the subscription.



5. The **app2 workload owner** account sends a request to the **network operations user** to peer the `app2-prod-vnet` with the `hub-vnet`. The **network operations** user creates the peering connection.



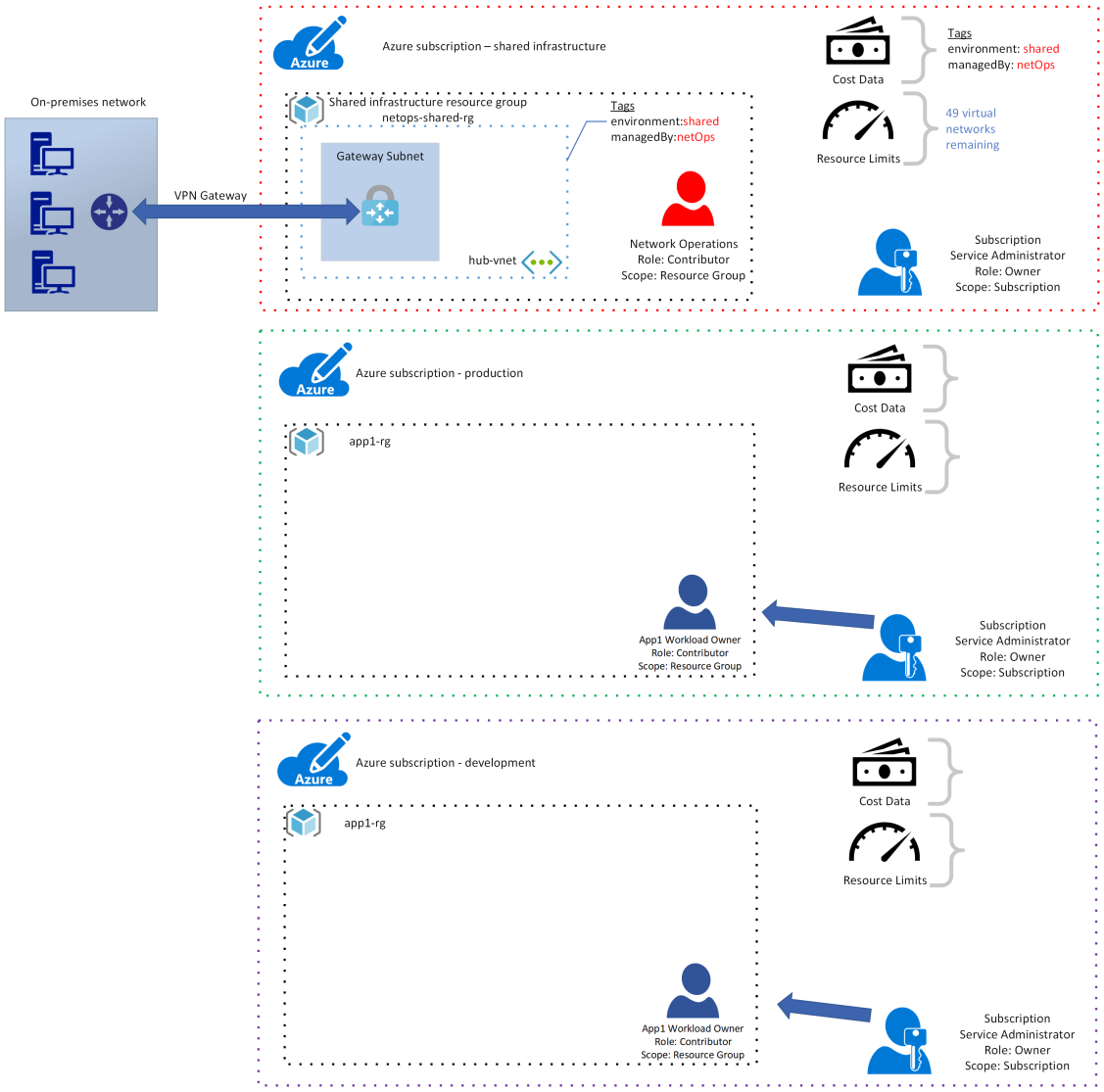
The resulting management model is similar to the first example, with several key differences:

- Each of the two workloads is isolated by workload and by environment.
- This model required two more virtual networks than the first example model. While this is not an important distinction with only two workloads, the theoretical limit on the number of workloads for this model is 24.
- Resources are no longer grouped in a single resource group for each environment. Grouping resources requires an understanding of the naming conventions used for each environment.
- Each of the peered virtual network connections was reviewed and approved by the **network operations user**.

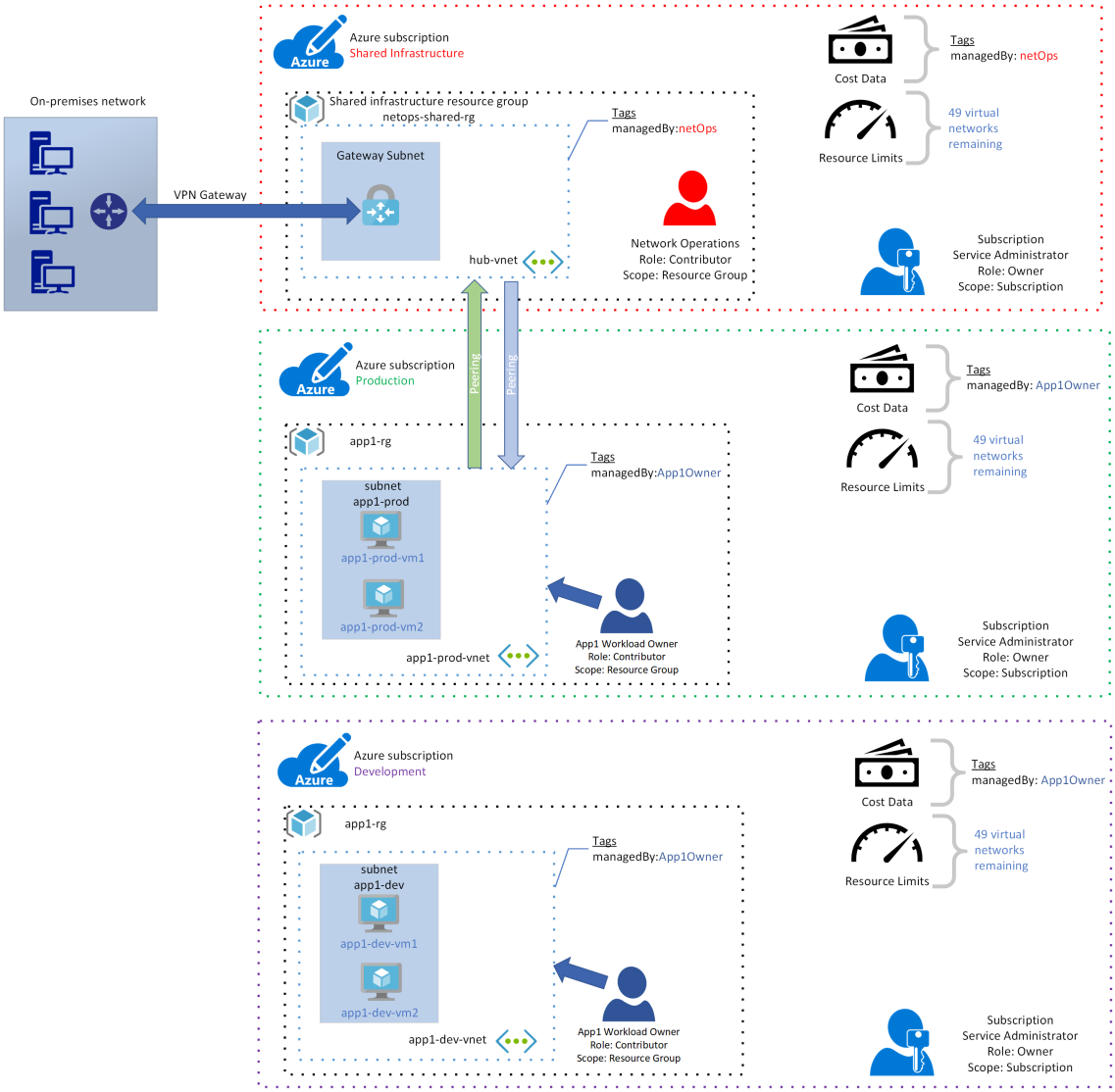
Now let's look at a resource management model using multiple subscriptions. In this model, you'll align each of the three environments to a separate subscription: a **shared services** subscription, **production** subscription, and finally a **development** subscription. The considerations for this model are similar to a model using a single subscription in that you have to decide how to align resource groups to workloads. Already determined is that

creating a resource group for each workload satisfies the workload isolation requirement, so you'll stick with that model in this example.

1. In this model, there are three subscriptions: **shared infrastructure**, **production**, and **development**. Each of these three subscriptions requires a subscription owner, and in the simple example you'll use the same user account for all three. The **shared infrastructure** resources are managed similarly to the first two examples above, and the first workload is associated with the `app1-rg` resource group in the **production** environment and the same-named resource group in the **development** environment. The **app1 workload owner** account is added to each of the resource group with the **Contributor** role.



2. As with the earlier examples, **app1 workload owner** creates the resources and requests the peering connection with the **shared infrastructure** virtual network. The **app1 workload owner** account adds only the `managedBy` tag because there is no longer a need for the `environment` tag. That is, resources are for each environment are now grouped in the same **subscription** and the `environment` tag is redundant. The limit counter is decremented to 999 virtual networks remaining.



3. Finally, the **subscription owner** account repeats the process for the second workload, adding the resource groups with **app2 workload owner** in the **Contributor** role. The limit counter for each of the environment subscriptions is decremented to 998 virtual networks remaining.

This management model has the benefits of the second example above. The key difference is that limits are less of an issue due to the fact that they're spread over two subscriptions. The drawback is that the cost data tracked by tags must be aggregated across all three subscriptions.

Therefore, you can select any of these two examples resource management models depending on the priority of your requirements. If you anticipate that your organization will not reach the service limits for a single subscription, you can use a single subscription with multiple resource groups. Conversely, if your organization anticipates many workloads, multiple subscriptions for each environment may be better.

Implement the resource management model

You've learned about several different models for governing access to Azure resources. Now you'll walk through the steps necessary to implement the resource management model with one subscription for each of the **shared infrastructure**, **production**, and **development** environments from the design guide. You'll have one **subscription owner** account for all three environments. Each workload will be isolated in a **resource group** with a **workload owner** added with the **Contributor** role.

> **NOTE**
> To learn more about the relationship between Azure accounts and subscriptions, see Understanding resource access in Azure.

Follow these steps:

1. Create an Azure account if your organization doesn't already have one. The person who signs up for the Azure account becomes the Azure account administrator, and your organization's leadership must select an individual to assume this role. This individual will be responsible for:
   - Creating subscriptions.
   - Creating and administering Azure Active Directory (Azure AD) tenants that store user identity for those subscriptions.
2. Your organization's leadership team decides who is responsible for:
   - Management of user identity; an Azure AD tenant is created by default when your organization's Azure account is created, and the account administrator is added as the Azure AD Global Administrator by default. Your organization can choose another user to manage user identity by assigning the Azure AD Global Administrator role to that user.
   - Subscriptions, which means these users:
     - Manage costs associated with resource usage in that subscription.
     - Implement and maintain least permission model for resource access.
     - Keep track of service limits.
   - Shared infrastructure services (if your organization decides to use this model), which means this user is responsible for:
     - On-premises to Azure network connectivity.
     - Ownership of network connectivity within Azure through virtual network peering.
   - Workload owners.
3. The Azure AD Global Administrator creates the new user accounts for:
   - The person who will be the subscription owner for each subscription associated with each environment. Note that this is necessary only if the subscription Service Administrator will not be tasked with managing resource access for each subscription/environment.
   - The person who will be the **network operations user**.
   - The people who are **workload owners**.
4. The Azure account administrator creates three Azure subscriptions:
   - A subscription for the **shared infrastructure** environment.
   - A subscription for the **production** environment.
   - A subscription for the **development** environment.
5. The Azure account administrator adds the subscription service owner to each subscription.
6. Create an approval process for **workload owners** to request the creation of resource groups. The approval process can be implemented in many ways, such as over email, or you can using a process management tool such as SharePoint workflows. The approval process can follow these steps:
   - The **workload owner** prepares a bill of materials for required Azure resources in either the **development** environment, **production** environment, or both, and submits it to the **subscription owner**.
   - The **subscription owner** reviews the bill of materials and validates the requested resources to ensure that the requested resources are appropriate for their planned use, such as checking that the requested virtual machine sizes are correct.
   - If the request is not approved, the **workload owner** is notified. If the request is approved, the **subscription owner** creates the requested resource group following your organization's naming

conventions, adds the **workload owner** with the **Contributor** role and sends notification to the **workload owner** that the resource group has been created.

7. Create an approval process for workload owners to request a virtual network peering connection from the shared infrastructure owner. As with the previous step, this approval process can be implemented using email or a process management tool.

Now that you've implemented your governance model, you can deploy your shared infrastructure services.

Related resources

Azure built-in roles

Deployment Acceleration discipline overview

3/6/2021 • 2 minutes to read • Edit Online

Deployment acceleration is one of the Five Disciplines of Cloud Governance within the Cloud Adoption Framework governance model. This discipline focuses on ways of establishing policies to govern asset configuration or deployment. Within the Five Disciplines of Cloud Governance, the Deployment Acceleration discipline includes deployment, configuration alignment, and script reusability. This could be through manual activities or fully automated DevOps activities. In either case, the policies would remain largely the same. As this discipline matures, the cloud governance team can serve as a partner in DevOps and deployment strategies by accelerating deployments and removing barriers to cloud adoption, through the application of reusable assets.

This article outlines the deployment acceleration process that a company experiences during the planning, building, adopting, and operating phases of implementing a cloud solution. It's impossible for any one document to account for all of the requirements of any business. As such, each section of this article outlines suggested minimum and potential activities. The objective of these activities is to help you build a policy MVP, but establish a framework for incremental policy improvement. The cloud governance team should decide how much to invest in these activities to improve the deployment acceleration position.

> **NOTE**
> The Deployment Acceleration discipline does not replace the existing IT teams, processes, and procedures that allow your organization to effectively deploy and configure cloud-based resources. The primary purpose of this discipline is to identify potential business risks and provide risk-mitigation guidance to the IT staff that are responsible for managing your resources in the cloud. As you develop governance policies and processes make sure to involve relevant IT teams in your planning and review processes.

The primary audience for this guidance is your organization's cloud architects and other members of your cloud governance team. The decisions, policies, and processes that emerge from this discipline should involve engagement and discussions with relevant members of your business and IT teams, especially those leaders responsible for deploying and configuring cloud-based workloads.

Policy statements

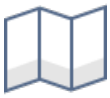
Actionable policy statements and the resulting architecture requirements serve as the foundation of a Deployment Acceleration discipline. Use sample policy statements as a starting point for defining your Deployment Acceleration policies.

Caution

The sample policies come from common customer experiences. To better align these policies to specific cloud governance needs, execute the following steps to create policy statements that meet your unique business needs.

Develop governance policy statements

The following steps help you define governance policies to control deployment and configuration of resources in your cloud environment.

	Deployment Acceleration discipline template: Download the template for documenting a Deployment Acceleration discipline.
	Business risks: Understand the motives and risks commonly associated with the Deployment Acceleration discipline.
	Indicators and metrics: Indicators to understand whether it's the right time to invest in the Deployment Acceleration discipline.
	Policy adherence processes: Suggested processes for supporting policy compliance in the Deployment Acceleration discipline.
	Maturity: Align cloud management maturity with phases of cloud adoption.
	Toolchain: Azure services that can be implemented to support the Deployment Acceleration discipline.

Next steps

Get started by evaluating business risks in a specific environment.

Understand business risks

Deployment acceleration template

3/6/2021 • 2 minutes to read • Edit Online

The first step to implementing change is communicating the desired change. The same is true when changing governance practices. The following template serves as a starting point for documenting and communicating policy statements that govern configuration and deployment issues in the cloud. The template also outlines the business criteria that may have led you to create the documented policy statements.

As your discussions progress, use this template's structure as a model for capturing the business risks, risk tolerances, compliance processes, and tooling needed to define your organization's Deployment Acceleration policy statements.

> **IMPORTANT**
> This template is a limited sample. Before updating this template to reflect your requirements, you should review the subsequent steps for defining an effective Deployment Acceleration discipline within your cloud governance strategy.

Download the Deployment Acceleration discipline template

Next steps

Solid governance practices start with an understanding of business risk. Review the article on business risks and begin to document the business risks that align with your current cloud adoption plan.

Understand business risks

Motivations and business risks in the Deployment Acceleration discipline

3/6/2021 • 2 minutes to read • Edit Online

This article discusses the reasons that customers typically adopt a Deployment Acceleration discipline within a cloud governance strategy. It also provides a few examples of business risks that drive policy statements.

Relevance

On-premises systems are often deployed using baseline images or installation scripts. Additional configuration is usually necessary, which may involve multiple steps or human intervention. These manual processes are error-prone and often result in "configuration drift", requiring time-consuming troubleshooting and remediation tasks.

Most Azure resources can be deployed and configured manually via the Azure portal. This approach may be sufficient for your needs when only have a few resources to manage. As your cloud estate grows, your organization should begin to integrate automation into your deployment processes to ensure your cloud resources avoid configuration drift or other problems introduced by manual processes. Adopting a DevOps or DevSecOps approach is often the best way to manage your deployments as you cloud adoption efforts mature.

A robust deployment acceleration plan ensures that your cloud resources are deployed, updated, and configured correctly and consistently, and remain that way. The maturity of your Deployment Acceleration strategy can also be a significant factor in your Cost Management strategy. Automated provisioning and configuration of your cloud resources allows you to scale down or deallocate resources when demand is low or time-bound, so you only pay for resources as you need them.

Business risk

The Deployment Acceleration discipline attempts to address the following business risks. During cloud adoption, monitor each of the following for relevance:

- **Service disruption:** Lack of predictable repeatable deployment processes or unmanaged changes to system configurations can disrupt normal operations and can result in lost productivity or lost business.
- **Cost overruns:** Unexpected changes in configuration of system resources can make identifying root cause of issues more difficult, raising the costs of development, operations, and maintenance.
- **Organizational inefficiencies:** Barriers between development, operations, and security teams can cause numerous challenges to effective adoption of cloud technologies and the development of a unified cloud governance model.

Next steps

Use the Deployment Acceleration discipline template to document business risks that are likely to be introduced by the current cloud adoption plan.

Once an understanding of realistic business risks is established, the next step is to document the business's tolerance for risk and the indicators and key metrics to monitor that tolerance.

Metrics, indicators, and risk tolerance

Risk tolerance metrics and indicators in the Deployment Acceleration discipline

3/6/2021 • 2 minutes to read • Edit Online

Learn to quantify business risk tolerance associated with the Deployment Acceleration discipline. Defining metrics and indicators helps to create a business case for investing in the maturity of this discipline.

Metrics

Deployment acceleration focuses on risks related to how cloud resources are configured, deployed, updated, and maintained. The following information is useful when adopting the Deployment Acceleration discipline:

- **Deployment failures:** Percentage of deployments that fail or result in misconfigured resources.
- **Time to deployment:** The amount of time needed to deploy updates to an existing system.
- **Assets out-of-compliance:** The number or percentage of resources that are out of compliance with defined policies.

Risk tolerance indicators

Risks related to deployment acceleration are largely related to the number and complexity of cloud-based systems deployed for your enterprise. As your cloud estate grows, the number of systems deployed and the frequency of updating your cloud resources will increase. Dependencies between resources magnify the importance of ensuring proper configuration of resources and designing systems for resiliency if one or more resources experiences unexpected downtime.

Traditional corporate IT organizations often have siloed operations, security, and development teams that often do not collaborate well or are even adversarial or hostile toward one another. Recognizing these challenges early and integrating key stakeholders from each of the teams can help ensure agility in your cloud adoption while remaining secure and well-governed. Therefore, consider adopting a DevOps or DevSecOps organizational culture early in your cloud adoption journey.

Work with your DevSecOps team and business stakeholders to identify business risks related to configuration, then determine an acceptable baseline for configuration risk tolerance. This section of the Cloud Adoption Framework guidance provides examples, but the detailed risks and baselines for your company or deployments will likely differ.

Once you have a baseline, establish minimum benchmarks representing an unacceptable increase in your identified risks. These benchmarks act as triggers for when you need to take action to remediate these risks. The following are a few examples of how configuration-related metrics, such as those discussed above, can justify an increased investment in the Deployment Acceleration discipline.

- **Configuration drift triggers:** A company that is experiencing unexpected changes in the configuration of key system components, or failures in the deployment of or updates to its systems, should invest in the Deployment Acceleration discipline to identify root causes and steps for remediation.
- **Out of compliance triggers:** If the number of out-of-compliance resources exceeds a defined threshold (either as a total number of resources or a percentage of total resources), a company should invest in Deployment Acceleration discipline improvements to ensure each resource's configuration remains in compliance throughout that resource's lifecycle.
- **Project schedule triggers:** If the time to deploy a company's resources and applications often exceed a define threshold, a company should invest in its deployment acceleration processes to introduce or improve

automated deployments for consistency and predictability. Deployment times measured in days or even weeks usually indicate a suboptimal Deployment Acceleration strategy.

Next steps

Use the Deployment Acceleration discipline template to document metrics and tolerance indicators that align to the current cloud adoption plan.

Review sample Deployment Acceleration policies as a starting point to develop your own policies to address specific business risks aligned with your cloud adoption plans.

Review sample policies

Deployment Acceleration sample policy statements

3/6/2021 • 3 minutes to read • Edit Online

Individual cloud policy statements are guidelines for addressing specific risks identified during your risk assessment process. These statements should provide a concise summary of risks and plans to deal with them. Each statement definition should include these pieces of information:

- **Technical risk:** A summary of the risk this policy will address.
- **Policy statement:** A clear summary explanation of the policy requirements.
- **Design options:** Actionable recommendations, specifications, or other guidance that IT teams and developers can use when implementing the policy.

The following sample policy statements address common configuration-related business risks. These statements are examples you can reference when drafting policy statements to address your organization's needs. These examples are not meant to be proscriptive, and there are potentially several policy options for dealing with each identified risk. Work closely with business and IT teams to identify the best policies for your unique set of risks.

Reliance on manual deployment or configuration of systems

Technical risk: Relying on human intervention during deployment or configuration increases the likelihood of human error and reduces the repeatability and predictability of system deployments and configuration. It also typically leads to slower deployment of system resources.

Policy statement: All assets deployed to the cloud should be deployed using templates or automation scripts whenever possible.

Potential design options: Azure Resource Manager templates enable using infrastructure as code to deploy your resources to Azure. You could also use Terraform as a consistent on-premises and cloud-based deployment tool.

Lack of visibility into system issues

Technical risk: Insufficient monitoring and diagnostics for business systems prevent operations personnel from identifying and remediating issues before a system outage occurs, and can significantly increase the time needed to properly resolve an outage.

Policy statement: The following policies will be implemented:

- Key metrics and diagnostics measures will be identified for all production systems and components, and monitoring and diagnostic tools will be applied to these systems and monitored regularly by operations personnel.
- Operations will consider using monitoring and diagnostic tools in nonproduction environments such as staging and QA to identify system issues before they occur in the production environment.

Potential design options: Azure Monitor, including Log Analytics and Application Insights, provides tools for collecting and analyzing telemetry to help you understand how your applications are performing and proactively identify issues affecting them and the resources they depend on. Additionally, Azure activity log reports all changes that are being made at the platform level and should be monitored and audited for noncompliant changes.

Configuration security reviews

Technical risk: Over time, new security threats or concerns can increase the risks of unauthorized access to secure resources.

Policy statement: Cloud governance processes must include monthly review with configuration management teams to identify malicious actors or usage patterns that should be prevented by cloud asset configuration.

Potential design options: Establish a monthly security review meeting that includes both governance team members and IT staff responsible for configuration cloud applications and resources. Review existing security data and metrics to establish gaps in current Deployment Acceleration policy and tooling, and update policy to remediate any new risks.

Next steps

Use the samples mentioned in this article as a starting point to develop policies that address specific business risks that align with your cloud adoption plans.

To begin developing your own custom Identity Baseline policy statements, download the Identity Baseline discipline template.

To accelerate adoption of this discipline, choose the actionable governance guide that most closely aligns with your environment. Then modify the design to incorporate your specific corporate policy decisions.

Building on risks and tolerance, establish a process for governing and communicating Deployment Acceleration policy adherence.

Establish policy compliance processes

Deployment Acceleration policy compliance processes

5/18/2021 • 4 minutes to read • Edit Online

This article discusses an approach to policy-adherence processes that govern the Deployment Acceleration discipline. Effective governance of cloud configuration starts with recurring manual processes designed to detect issues and impose policies to remediate those risks. You can automate these processes and supplement to reduce the overhead of governance and allow for faster response to deviation.

Planning, review, and reporting processes

The best Deployment Acceleration tools in the cloud are only as good as the processes and policies that they support. The following is a set of example processes commonly used as part of a Deployment Acceleration discipline. Use these examples as a starting point when planning the processes that will allow you to continue to update deployment and configuration policy based on business change and feedback from the development and IT teams responsible for turning governance guidance into action.

Initial risk assessment and planning: As part of your initial adoption of the Deployment Acceleration discipline, identify your core business risks and tolerances related to deployment of your business applications. Use this information to discuss specific technical risks with members of the IT operations team, and develop a baseline set of deployment and configuration policies for remediating these risks to establish your initial governance strategy.

Deployment planning: Before deploying any asset, perform a security and operations review to identify any new risks and ensure all deployment related policy requirements are met.

Deployment testing: As part of the deployment process for any asset, the cloud governance team, in cooperation with your IT operations teams, is responsible for reviewing the deployment policy compliance.

Annual planning: Conduct an annual high-level review of Deployment Acceleration strategy. Explore future corporate priorities and updated cloud adoption strategies to identify potential risk increase and other emerging configuration needs and opportunities. Also use this time to review the latest DevOps best practices and integrate these into your policies and review processes.

Quarterly review and planning: Conduct a quarterly review of operational audit data and incident reports to identify any changes required in Deployment Acceleration policy. As part of this process, review current DevOps and TechOps best practices, and update policy as appropriate. After the review is complete, align application and systems design guidance with updated policy.

This planning process is also a good time to evaluate the current membership of your cloud governance team for knowledge gaps related to new or changing policy and risks related to DevOps and deployment acceleration. Invite relevant IT staff to participate in reviews and planning as either temporary technical advisors or permanent members of your team.

Education and training: On a bimonthly basis, offer training sessions to make sure IT staff and developers are up-to-date on the latest Deployment Acceleration strategy and requirements. As part of this process review and update any documentation, guidance, or other training assets to ensure they're in sync with the latest corporate policy statements.

Monthly audit and reporting reviews: Perform a monthly audit on all cloud deployments to assure their continued alignment with configuration policy. Review deployment-related activities with IT staff and identify

any compliance issues not already handled as part of the ongoing monitoring and enforcement process. The result of this review is a report for the cloud strategy team and each cloud adoption team to communicate overall adherence to policy. The report is also stored for auditing and legal purposes.

Processes for ongoing monitoring

A successful Deployment Acceleration strategy depends on visibility into the current and past state of your cloud infrastructure. Without the ability to analyze the relevant metrics and data of your cloud resources operational health and activity, you cannot identify changes in your risks or detect violations of your risk tolerances. The ongoing governance processes discussed above requires quality data to ensure policy can be modified to protect your infrastructure against changing threats and risks from misconfigured resources.

Ensure that your IT operations teams have implemented automated monitoring systems for your cloud infrastructure that capture the relevant logs data you need to evaluate risk. Be proactive in monitoring these systems to ensure prompt detection and mitigation of potential policy violation, and ensure your monitoring strategy is in line with deployment and configuration needs.

Violation triggers and enforcement actions

Because noncompliance with configuration policies can lead to critical service disruption risks, the cloud governance team should have visibility into serious policy violations. Ensure IT staff have clear escalation paths for reporting configuration compliance issues to the governance team members best suited to identify and verify that policy issues are mitigated when detected.

When violations are detected, you should take actions to realign with policy as soon as possible. Your IT team can automate most violation triggers using the tools outlined in the Deployment Acceleration toolchain for Azure.

The following triggers and enforcement actions provide examples you can use when discussing how to use monitoring data to resolve policy violations:

- **Unexpected changes in configuration detected.** If the configuration of a resource changes unexpectedly, work with IT staff and workload owners to identify root cause and develop a remediation plan.
- **Configuration of new resources does not adhere to policy.** Work with DevOps teams and workload owners to review Deployment Acceleration policies during project startup so everyone involved understands the relevant policy requirements.
- **Deployment failures or configuration issues cause delays in project schedules.** Work with development teams and workload owners to ensure the team understands how to automate the deployment of cloud-based resources for consistency and repeatability. Fully automated deployments should be required early in the development cycle. Trying to accomplish this later usually leads to unexpected issues and delays.

Next steps

Use the Deployment Acceleration discipline template to document the processes and triggers that align to the current cloud adoption plan.

For guidance on executing cloud management policies in alignment with adoption plans, see Deployment Acceleration discipline improvement.

Deployment Acceleration discipline improvement

Deployment Acceleration discipline improvement

3/6/2021 • 4 minutes to read • Edit Online

The Deployment Acceleration discipline focuses on establishing policies that ensure that resources are deployed and configured consistently and repeatably, and remain in compliance throughout their lifecycle. Within the Five Disciplines of Cloud Governance, the Deployment Acceleration discipline includes decisions regarding automating deployments, source-controlling deployment artifacts, monitoring deployed resources to maintain desired state, and auditing any compliance issues.

This article outlines some potential tasks your company can engage in to better develop and mature the Deployment Acceleration discipline. These tasks can be broken down into planning, building, adopting, and operating phases of implementing a cloud solution, which are then iterated on allowing the development of an incremental approach to cloud governance.

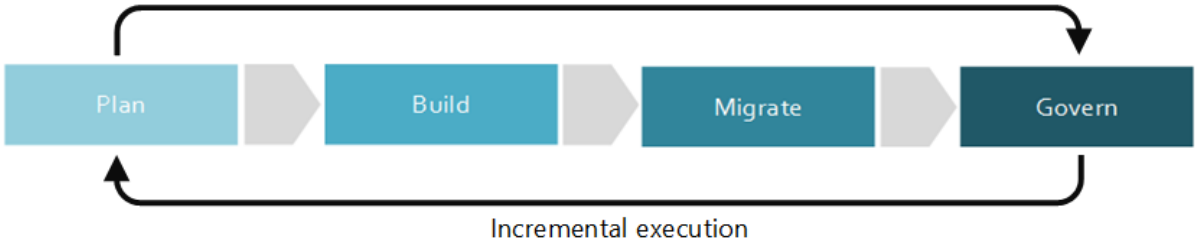


Figure 1: Phases of an incremental approach to cloud governance.

It's impossible for any one document to account for the requirements of all businesses. As such, this article outlines suggested minimum and potential example activities for each phase of the governance maturation process. The initial objective of these activities is to help you build a policy MVP and establish a framework for incremental policy improvement. Your cloud governance team will need to decide how much to invest in these activities to improve your Identity Baseline discipline.

Caution

Neither the minimum or potential activities outlined in this article are aligned to specific corporate policies or third-party compliance requirements. This guidance is designed to help facilitate the conversations that will lead to alignment of both requirements with a cloud governance model.

Planning and readiness

This phase of governance maturity bridges the divide between business outcomes and actionable strategies. During this process, the leadership team defines specific metrics, maps those metrics to the digital estate, and begins planning the overall migration effort.

Minimum suggested activities:

- Evaluate your Deployment Acceleration toolchain options and implement a hybrid strategy that is appropriate to your organization.
- Develop a draft architecture guidelines document and distribute to key stakeholders.
- Educate and involve the people and teams affected by the development of architecture guidelines.
- Train development teams and IT staff to understand DevSecOps principles and strategies and the importance of fully automated deployments in the Deployment Acceleration discipline.

Potential activities:

- Define roles and assignments that will govern deployment acceleration in the cloud.

Build and predeployment

Minimum suggested activities:

- For new cloud-based applications, introduce fully automated deployments early in the development process. This investment will improve the reliability of your testing processes and ensure consistency across your development, QA, and production environments.
- Store all deployment artifacts such as deployment templates or configuration scripts using a source-control platform such as GitHub or Azure DevOps.
- Store all secrets, passwords, certificates, and connection strings in Azure Key Vault.
- Consider a pilot test before implementing your Deployment Acceleration toolchain, making sure it streamlines your deployments as much as possible. Apply feedback from pilot tests during the predeployment phase, repeating as needed.
- Evaluate the logical and physical architecture of your applications, and identify opportunities to automate the deployment of application resources or improve portions of the architecture using other cloud-based resources.
- Update the architecture guidelines document to include deployment and user adoption plans, and distribute to key stakeholders.
- Continue to educate the people and teams most affected by the architecture guidelines.

Potential activities:

- Define a continuous integration and continuous deployment (CI/CD) pipeline to fully manage releasing updates to your application through your development, QA, and production environments.

Adopt and migrate

Migration is an incremental process that focuses on the movement, testing, and adoption of applications or workloads in an existing digital estate.

Minimum suggested activities:

- Migrate your Deployment Acceleration toolchain from development to production.
- Update the architecture guidelines document and distribute to key stakeholders.
- Develop educational materials and documentation, awareness communications, incentives, and other programs to help drive developer and IT adoption.

Potential activities:

- Validate that the best practices defined during the build and predeployment phases are properly executed.
- Ensure that each application or workload aligns with the Deployment Acceleration strategy before release.

Operate and post-implementation

Once the transformation is complete, governance and operations must live on for the natural lifecycle of an application or workload. This phase of governance maturity focuses on the activities that commonly come after the solution is implemented and the transformation cycle begins to stabilize.

Minimum suggested activities:

- Customize your Deployment Acceleration toolchain based on your organization's changing needs.
- Automate notifications and reports to alert you of potential configuration issues or malicious threats.
- Monitor and report on application and resource usage.
- Report on post-deployment metrics and distribute to stakeholders.
- Revise the architecture guidelines to guide future adoption processes.

- Continue to communicate with and train the affected people and teams on a regular basis to ensure ongoing adherence to architecture guidelines.

Potential activities:

- Configure a desired state configuration monitoring and reporting tool.
- Regularly review configuration tools and scripts to improve processes and identify common issues.
- Work with development, operations, and security teams to help mature DevSecOps practices and break down organizational silos that lead to inefficiencies.

Next steps

Now that you understand the concept of cloud identity governance, examine the Identity Baseline toolchain to identify Azure tools and features that you'll need when developing your Identity Baseline discipline on the Azure platform.

Identity Baseline toolchain for Azure

Deployment Acceleration tools in Azure

3/6/2021 • 2 minutes to read • Edit Online

The Deployment Acceleration discipline is one of the Five Disciplines of Cloud Governance. This discipline focuses on ways of establishing policies to govern asset configuration or deployment. Within the Five Disciplines of Cloud Governance, the Deployment Acceleration discipline involves deployment and configuration alignment. This could be through manual activities or fully automated DevOps activities. In either case, the policies involved would remain largely the same.

Cloud custodians, cloud guardians, and cloud architects with an interest in governance are each likely to invest a lot of time in the Deployment Acceleration discipline, which codifies policies and requirements across multiple cloud adoption efforts. The tools in this toolchain are important to the cloud governance team and should be a high priority on the learning path for the team.

The following is a list of Azure tools that can help mature the policies and processes that support this discipline.

	AZURE POLICY	AZURE MANAGEMENT GROUPS	AZURE RESOURCE MANAGER	AZURE BLUEPRINTS	AZURE RESOURCE GRAPH	AZURE COST MANAGEMENT + BILLING
Implement corporate policies	Yes	No	No	No	No	No
Apply policies across subscriptions	Required	Yes	No	No	No	No
Deploy defined resources	No	No	Yes	No	No	No
Create fully compliant environments	Required	Required	Required	Yes	No	No
Audit policies	Yes	No	No	No	No	No
Query Azure resources	No	No	No	No	Yes	No
Report on cost of resources	No	No	No	No	No	Yes

The following are additional tools that may be required to accomplish specific deployment acceleration objectives. Often these tools are used outside of the governance team, but are still considered an aspect of the Deployment Acceleration discipline.

	AZURE PORTAL	AZURE RESOURCE MANAGER	AZURE POLICY	AZURE DEVOPS	AZURE BACKUP	AZURE SITE RECOVERY
Manual deployment (single asset)	Yes	Yes	No	Not efficiently	No	Yes
Manual deployment (full environment)	Not efficiently	Yes	No	Not efficiently	No	Yes
Automated deployment (full environment)	No	Yes	No	Yes	No	Yes
Update configuration of a single asset	Yes	Yes	Not efficiently	Not efficiently	No	Yes, during replication
Update configuration of a full environment	Not efficiently	Yes	Yes	Yes	No	Yes, during replication
Manage configuration drift	Not efficiently	Not efficiently	Yes	Yes	No	Yes, during replication
Create an automated pipeline to deploy code and configure assets (DevOps)	No	No	No	Yes	No	No

Aside from the Azure native tools mentioned above, it's common for customers to use third-party tools to facilitate deployment acceleration and DevOps deployments.

Govern antipatterns

4/22/2021 • 3 minutes to read • Edit Online

Customers often experience antipatterns during the Govern phase of cloud adoption. Taking the time to understand shared responsibilities helps you avoid these antipatterns, as does building your security strategy on existing frameworks instead of creating your own.

Antipattern: Misunderstand shared responsibilities

When you adopt the cloud, it's not always clear where your responsibility ends and the cloud provider's responsibility starts regarding the different service models. Cloud skills and knowledge are required to build processes and practices around work items that use service models.

Example: Assume the cloud provider manages updates

Members of a company's human resources (HR) department set up many Windows servers in the cloud using infrastructure as a service (IaaS). They assume that the cloud provider manages updates, because on-site IT usually handles update installation. The HR department doesn't configure the updates because they're not aware that Azure doesn't deploy and install operating system updates by default. As a result, the servers are noncompliant and pose a security risk.

Preferred outcome: Create a readiness plan

Understand shared responsibility in the cloud. Build and create a readiness plan. A readiness plan can create ongoing momentum for learning and developing expertise.

Antipattern: Assume out-of-the-box solutions provide security

Companies often perceive security as a given in the cloud. Although this assumption is usually correct, most environments also need to adhere to compliance framework requirements, which can differ from security requirements. Azure provides basic security. And, through the Azure Security Center, the Azure portal provides help with improving security. But, enforcing a compliance and security standard isn't an out-of-the-box experience when you create a subscription.

Example: Neglect cloud security

A company develops a new application in the cloud. It chooses an architecture based on many platform as a service (PaaS) services, plus some IaaS components for debugging purposes. After releasing the application to production, the company realizes that one of its jump servers has been compromised and was extracting data to an unknown IP address. The company discovers that the problem is the jump server's public IP address and a password that's easy to guess. The company could have avoided this situation if it had focused more on cloud security.

Preferred outcome: Define a cloud security strategy

Define a proper cloud security strategy. See CISO cloud readiness guide for more information, and refer your chief information security office (CISO) to this guide. It discusses topics such as security platform resources, privacy and controls, compliance, and transparency.

Read about secure cloud workloads in the Azure Security Benchmark. Build upon the CIS Controls v7.1 from the Center for Internet Security, along with NIST SP800-53 from the National Institute of Standards and Technology, which address most security risks and measures.

Use the Azure Security Center to identify risks, adapt best practices, and improve your company's security posture.

Implement or support company-specific, automated compliance and security requirements by using Azure Policy and Azure Blueprints.

Antipattern: Use a custom compliance or governance framework

Introducing a custom compliance and governance framework that isn't based on industry standards can substantially increase cloud adoption time, because it can be difficult to translate the custom framework to cloud settings. This scenario can increase the effort that's needed to translate custom measures and requirements into implementable security controls. Most companies need to comply with similar sets of security and compliance requirements. As a result, most custom compliance and security frameworks differ only slightly from current compliance frameworks. Companies with additional security requirements can consider building new frameworks.

Example: Use a custom security framework

The CISO of a company assigns IT security employees the task of coming up with a cloud security strategy and framework. Instead of building upon industry standards, the IT security department creates a new framework that builds on the current on-premises security policy. After completing the cloud security policy, the AppOps and DevOps teams have difficulty implementing the cloud security policy.

Azure offers a more comprehensive security and compliance structure that differs from the company's own framework. The CISO team thinks that the Azure controls are incompatible with its own compliance and security rules. If it had based its framework on standardized controls, it wouldn't have come to that conclusion.

Preferred outcome: Rely on existing frameworks

Use or build on existing frameworks, such as CIS Controls v7.1 or NIST SP800-53, before you establish or introduce a custom company compliance framework. Existing frameworks make the transition to cloud security settings easier and more measurable. Find more framework implementations on the Azure Blueprints samples page. Blueprints for common compliance frameworks are also available for Azure.

Next steps

- Adapt existing roles, skills, and processes for the cloud
- Define a security strategy

Cloud management in the Cloud Adoption Framework

5/4/2021 • 2 minutes to read • Edit Online

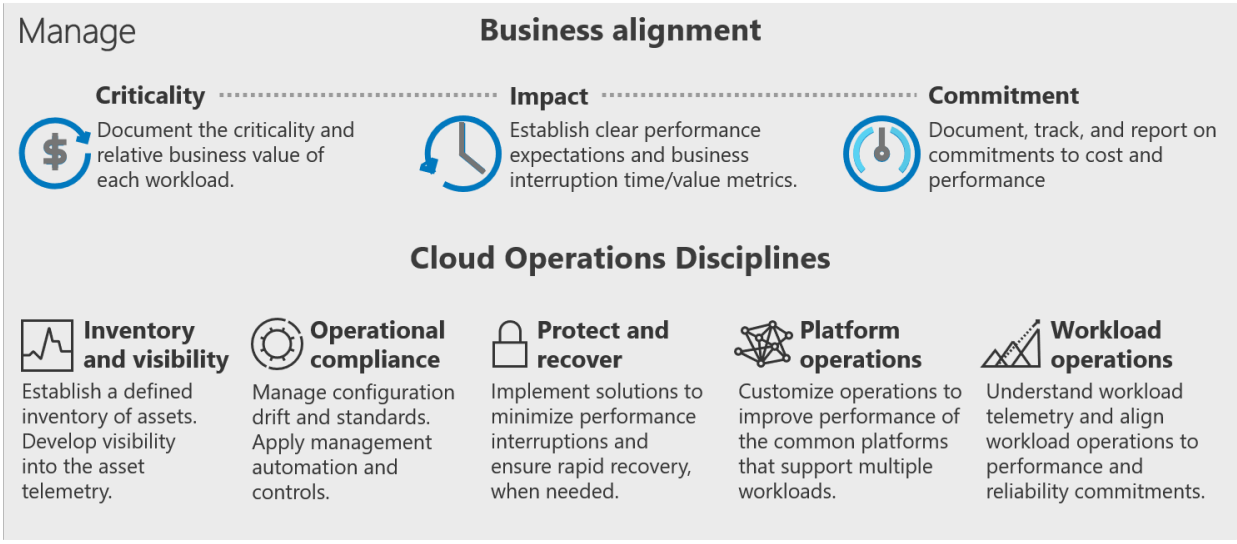
Delivering on a cloud strategy requires solid planning, readiness, and adoption. But it's the ongoing operation of the digital assets that delivers tangible business outcomes. Without a plan for reliable, well-managed operations of the cloud solutions, those efforts will yield little value. The following exercises help develop the business and technical approaches needed to provide cloud management that powers ongoing operations.

Get started

To prepare you for this phase of the cloud adoption lifecycle, the framework suggests the following exercises:

	Define business commitments: Document supported workloads to establish operational commitments with the business and agree on cloud management investments for each workload.
	Establish a management baseline: Define the criticality classifications, cloud management tools, and processes required to deliver your minimum commitment to operations management.
	Expand the management baseline: Based on business commitments and operations decisions, make use of the included best practices to implement the required cloud management tooling.
	Advanced operations and design principles: Platforms or workloads that require a higher level of business commitment might require a deeper architecture review to deliver on resiliency and reliability commitments.

The preceding steps create actionable approaches to deliver on the Cloud Adoption Framework's Manage methodology.



As discussed in the business alignment article, not all workloads are mission critical. Within any portfolio are various degrees of operational management needs. Business alignment efforts aid in capturing the business impact and negotiating management costs with the business, to ensure the most appropriate operational management processes and tools.

The guidance in the manage section of the Cloud Adoption Framework serves two purposes:

- Provides examples of actionable operations management approaches that represent common experiences often encountered by customers.
- Helps you create personalized management solutions based on business commitments.

This content is intended for use by the cloud operations team. It's also relevant to cloud architects who need to develop a strong foundation in cloud operations or cloud design principles.

The content in the Cloud Adoption Framework affects the business, technology, and culture of enterprises. This section of the Cloud Adoption Framework interacts heavily with IT operations, IT governance, finance, line-of-business leaders, networking, identity, and cloud adoption teams. Various dependencies on these personnel require a facilitative approach by the cloud architects who are using this guidance. Facilitation with these teams is seldom a one-time effort.

The cloud architect serves as the thought leader and facilitator to bring these audiences together. The content in this collection of guides is designed to help the cloud architect facilitate the right conversation, with the right audience, to drive necessary decisions. Business transformation that's empowered by the cloud depends on the cloud architect to help guide decisions throughout the business and IT.

Each section of the Cloud Adoption Framework represents a different specialization or variant of the cloud architect role. This section of the Cloud Adoption Framework is designed for cloud architects with a passion for operations and management of deployment solutions. Within this framework, these specialists are referred to frequently as *cloud operations*, or collectively as the *cloud operations team*.

If you want to follow this guide from beginning to end, this content aids in developing a robust cloud operations strategy. The guidance walks you through the theory and implementation of such a strategy.

You can also apply the methodology to establish clear business commitments.

Azure Management Guide: Before you start

5/4/2021 • 2 minutes to read • Edit Online

Before you start

The Azure Management Guide helps Azure customers create a management baseline to establish resource consistency across Azure. This guide outlines the basic tools needed for any Azure production environments, especially environments that host sensitive data. For more information, best practices, and considerations related to preparing your cloud environment, see the readiness section of the Cloud Adoption Framework.

Scope of this guide

This guide teaches you how to establish tooling for a management baseline. It also outlines ways to extend the baseline or build resiliency beyond the baseline.

- **Inventory and visibility:** Create an inventory of assets across multiple clouds. Develop visibility into the run state of each asset.
- **Operational compliance:** Establish controls and processes to ensure each state is properly configured and running in a well-governed environment.
- **Protect and recover:** Ensure all managed assets are protected and can be recovered using baseline management tooling.
- **Enhanced baseline options:** Evaluate common additions to the baseline that might meet business needs.
- **Platform operations:** Extend the management baseline with a well-defined service catalog and centrally managed platforms.
- **Workload operations:** Extend the management baseline to include a focus on mission-critical workloads.

Management baseline

A management baseline is the minimum set of tools and processes that should be applied to every asset in an environment. Several additional options can be included in the management baseline. The next few articles accelerate cloud management capabilities by focusing on the minimum options necessary instead of on all of the available options.

The next step is Inventory and visibility.

This guide provides interactive steps that let you try features as they're introduced. To come back to where you left off, use the breadcrumb for navigation.

Inventory and visibility in Azure

5/26/2021 • 3 minutes to read • Edit Online

Inventory and visibility is the first of three disciplines in a cloud management baseline.



Basic level of cloud management for non-critical, production workloads

This discipline comes first because collecting proper operational data is vital when you make decisions about operations. Cloud management teams must understand what is managed and how well those assets are operated. This article describes the different tools that provide both an inventory and visibility into the inventory's run state.

For any enterprise-grade environment, the following table outlines the suggested minimum for a management baseline.

PROCESS	TOOL	PURPOSE
Monitor health of Azure services	Azure Service Health	Health, performance, and diagnostics for services running in Azure
Log centralization	Log Analytics	Central logging for all visibility purposes
Monitoring centralization	Azure Monitor	Central monitoring of operational data and trends
Virtual machine inventory and change tracking	Change Tracking and Inventory	Inventory VMs and monitor changes for guest OS level
Subscription monitoring	Azure activity log	Monitoring change at the subscription level
Guest OS monitoring	Azure Monitor for VMs	Monitoring changes and performance of VMs
Network monitoring	Azure Network Watcher	Monitoring network changes and performance

PROCESS	TOOL	PURPOSE
DNS monitoring	DNS Analytics	Security, performance, and operations of DNS

Azure Service Health

- Azure Service Health

Azure Service Health provides a personalized view of the health of your Azure services and regions. Information about active issues is posted to Azure Service Health to help you understand the effect on your resources. Regular updates keep you informed as issues are resolved.

We also publish planned maintenance events to Azure Service Health so you'll know about changes that can affect resource availability. Set up Service Health alerts to notify you when service issues, planned maintenance, or other changes might affect your Azure services and regions.

Azure Service Health includes:

- **Azure status:** A global view of the health of Azure services.
- **Service health:** A personalized view of the health of your Azure services.
- **Resource health:** A deeper view of the health of your individual resources.

Action

To set up a Service Health alert:

1. Go to **Service Health**.
2. Select **Health alerts**.
3. Create a Service Health alert.

GO TO SERVICE HEALTH

To set up Service Health alerts, go to the Azure portal.

Learn more

For more information, see Azure Service Health.

Log Analytics

- Log Analytics

A Log Analytics workspace is a unique environment for storing Azure Monitor log data. Each workspace has its own data repository and configuration. Data sources and solutions are configured to store their data in particular workspaces. Azure monitoring solutions require all servers to be connected to a workspace, so that their log data can be stored and accessed.

Action

EXPLORE AZURE MONITOR

Learn more

To learn more, see the Log Analytics workspace creation documentation.

Azure Monitor

- Azure Monitor

Azure Monitor provides a single unified hub for all monitoring and diagnostics data in Azure and gives you visibility across your resources. With Azure Monitor, you can find and fix problems and optimize performance. You can also understand customer behavior.

- **Monitor and visualize metrics.** Metrics are numerical values available from Azure resources. They help you understand the health of your systems. Customize charts for your dashboards, and use workbooks for reporting.
- **Query and analyze logs.** Logs include activity logs and diagnostic logs from Azure. Collect additional logs from other monitoring and management solutions for your cloud or on-premises resources. Log Analytics provides a central repository to aggregate all of this data. From there, you can run queries to help troubleshoot issues or to visualize data.
- **Set up alerts and actions.** Alerts notify you of critical conditions. Corrective actions can be taken based on triggers from metrics, logs, or service-health issues. You can set up different notifications and actions and can also send data to your IT service management tools.

Action

EXPLORE AZURE MONITOR

Start monitoring your:

- Applications
- Containers
- Virtual machines
- Networks

To monitor other resources, find additional solutions in Azure Marketplace.

To explore Azure Monitor, go to the Azure portal.

Learn more

To learn more, see Azure Monitor documentation.

Onboard solutions

- Onboard solutions

To enable solutions, you need to configure the Log Analytics workspace. Onboarded Azure VMs and on-premises servers get the solutions from the Log Analytics workspaces they're connected to.

There are two approaches to onboarding:

- Single VM
- Entire subscription

Each article guides you through a series of steps to onboard these solutions:

- Update Management solution
- Change Tracking and Inventory solution
- Azure activity log
- Azure Log Analytics Agent Health
- Antimalware assessment
- Azure Monitor for VMs

- Azure Security Center

Each of the previous steps helps establish inventory and visibility.

Operational compliance in Azure

5/26/2021 • 3 minutes to read • Edit Online

Operational compliance is the second discipline in any cloud management baseline.



Basic level of cloud management for non-critical, production workloads

Improving operational compliance reduces the likelihood of an outage related to configuration drift or vulnerabilities related to systems being improperly patched.

For any enterprise-grade environment, this table outlines the suggested minimum for a management baseline.

PROCESS	TOOL	PURPOSE
Patch management	Azure Automation Update Management	Management and scheduling of updates
Policy enforcement	Azure Policy	Policy enforcement to ensure environment and guest compliance
Environment configuration	Azure Blueprints	Automated compliance for core services
Resource configuration	Desired State Configuration	Automated configuration on guest OS and some aspects of the environment

Update Management

- Update Management

Computers that are managed by the Update Management solution for Azure Automation use the following configurations to do assessment and update deployments:

- Microsoft Monitoring Agent (MMA) for Windows or Linux.
- PowerShell Desired State Configuration (DSC) for Linux.
- Azure Automation Hybrid Runbook Worker.
- Microsoft Update or Windows Server Update Services (WSUS) for Windows computers.

For more information, see Update Management solution for Azure Automation.

> **WARNING**
>
> Before using Update Management, you must onboard virtual machines or an entire subscription into Log Analytics and Azure Automation.
>
> There are two approaches to onboarding:
>
> - Single VM
> - Entire subscription
>
> You should follow one before proceeding with Update Management.

Manage updates

To apply a policy to a resource group:

1. Go to Azure Automation.
2. Select **Automation accounts**, and choose one of the listed accounts.
3. Go to **Configuration Management**.
4. **Inventory**, **Change Management**, and **State Configuration** can be used to control the state and operational compliance of the managed VMs.

ASSIGN POLICY

Azure Policy

- Azure Policy

Azure Policy is used throughout governance processes. It's also highly valuable within cloud management processes. Azure Policy can audit and remediate Azure resources and can also audit settings inside a machine. The validation is performed by the Guest Configuration extension and client. The extension, through the client, validates settings like:

- Operating system configuration.
- Application configuration or presence.
- Environment settings.

Azure Policy Guest Configuration currently only audits settings inside the machine. It doesn't apply configurations.

Action

Assign a built-in policy to a management group, subscription, or resource group.

ASSIGN POLICY

Apply a policy

To apply a policy to a resource group:

1. Go to Azure Policy.
2. Select **Assign a policy**.

Learn more

To learn more, see:

- Azure Policy
- Azure Policy: Guest Configuration
- Cloud Adoption Framework policy enforcement decision guide

Azure Blueprints

- Azure Blueprints

With Azure Blueprints, cloud architects and central information-technology groups can define a repeatable set of Azure resources. These resources implement and adhere to an organization's standards, patterns, and requirements.

With Azure Blueprints, development teams can rapidly build and stand up new environments. Teams can also trust they're building within organizational compliance. They do so by using a set of built-in components like networking to speed up development and delivery.

Blueprints are a declarative way to orchestrate the deployment of different resource templates and other artifacts like:

- Role assignments.
- Policy assignments.
- Azure Resource Manager templates.
- Resource groups.

Applying a blueprint can enforce operational compliance in an environment if this enforcement isn't done by the cloud governance team.

Create a blueprint

To create a blueprint:

1. Go to **Blueprints: Getting started**.
2. On the **Create a Blueprint** pane, select **Create**.
3. Filter the list of blueprints to select the appropriate blueprint.
4. In the **Blueprint name** box, enter the blueprint name.
5. Select **Definition location**, and choose the appropriate location.
6. Select **Next : Artifacts**, and review the artifacts included in the blueprint.
7. Select **Save draft**.

CREATE A BLUEPRINT

1. Go to Blueprints: Getting started.
2. On the **Create a Blueprint** pane, select **Create**.
3. Filter the list of blueprints to select the appropriate blueprint.
4. In the **Blueprint name** box, enter the blueprint name.
5. Select **Definition location**, and choose the appropriate location.
6. Select **Next : Artifacts**, and review the artifacts included in the blueprint.
7. Select **Save draft**.

Publish a blueprint

To publish blueprint artifacts to your subscription:

1. Go to **Blueprints - Blueprint definitions**.
2. Select the blueprint you created in the previous steps.

3. Review the blueprint definition, then select **Publish blueprint**.
4. In the **Version** box, enter a version like "1.0".
5. In the **Change notes** box, enter your notes.
6. Select **Publish**.

BLUEPRINT DEFINITIONS

1. In the Azure portal, go to Blueprints: Blueprint definitions.
2. Select the blueprint you created in the previous steps.
3. Review the blueprint definition, then select **Publish blueprint**.
4. In the **Version** box, enter a version like "1.0".
5. In the **Change notes** box, enter your notes.
6. Select **Publish**.

Learn more

To learn more, see:

- Azure Blueprints
- Cloud Adoption Framework resource consistency decision guide
- Standards-based blueprints samples

Protect and recover in Azure

5/4/2021 • 2 minutes to read • Edit Online

Protect and recover is the third and final discipline in any cloud-management baseline.



In Operational compliance in Azure the objective is to reduce the likelihood of a business interruption. The current article aims to reduce the duration and impact of outages that can't be prevented.

For any enterprise-grade environment, this table outlines the suggested minimum for any management baseline:

PROCESS	TOOL	PURPOSE
Protect data	Azure Backup	Back up data and virtual machines in the cloud.
Protect the environment	Azure Security Center	Strengthen security and provide advanced threat protection across your hybrid workloads.

Azure Backup

- Azure Backup

With Azure Backup, you can back up, protect, and recover your data in the Microsoft cloud. Azure Backup replaces your existing on-premises or offsite backup solution with a cloud-based solution. This new solution is reliable, secure, and cost competitive. Azure Backup can also help protect and recover on-premises assets through one consistent solution.

For data present in Azure, Azure Backup offer varied levels of protection. For example, when backing up key cloud infrastructure pieces such as Azure Virtual Machines and Azure Files, it offers Azure Virtual Machines backup and Azure Files backup. For more critical components such as databases running in Azure Virtual Machines, it offers dedicated database backup solutions for SQL Server and SAP HANA with far lower RPO.

Review the following section to see how easily you can enable backup for Azure Virtual Machines.

Enable backup for an Azure VM

1. In the Azure portal, select **Virtual machines**, then select the VM you want to backup.
2. On the **Operations** pane, select **Backup**.
3. Create or select an existing Azure Recovery Services vault.
4. Select **Create (or edit) a new policy**.
5. Configure the schedule and retention period.
6. Select **OK**.
7. Select **Enable backup**.

GO TO VIRTUAL MACHINES

For more details about Azure Backup, see Overview of Azure Backup.

Azure Site Recovery

- Azure Site Recovery

Azure Site Recovery is a critical component in your disaster recovery strategy.

Site Recovery replicates VMs and workloads that are hosted in a primary Azure region. It replicates them to a copy that is hosted in a secondary region. When an outage occurs in your primary region, you fail over to the copy running in the secondary region. You then continue to access your applications and services from there. This proactive approach to recovery can significantly reduce recovery times. When the recovery environment is no longer needed, production traffic can fall back to the original environment.

Replicate an Azure VM to another region with Site Recovery

The following steps outline the process to use Site Recovery for Azure-to-Azure replication, which is replication of an Azure VM to another region.

> **TIP**
> Depending on your scenario, the exact steps might differ slightly.

Enable replication for the Azure VM

1. In the Azure portal, select **Virtual machines**, then select the VM you want to replicate.
2. On the **Operations** pane, select **Disaster recovery**.
3. Select **Configure disaster recovery** > **Target region**, and choose the target region to which you'll replicate.
4. For this quickstart, accept the default values for all other options.
5. Select **Enable replication**, which starts a job to enable replication for the VM.

GO TO VIRTUAL MACHINES

Verify settings

After the replication job has finished, you can check the replication status, verify replication health, and test the deployment.

1. In the VM menu, select **Disaster recovery**.
2. Verify replication health, the recovery points that have been created, and source and target regions on the map.

GO TO VIRTUAL MACHINES

Learn more

- Azure Site Recovery overview
- Replicate an Azure VM to another region

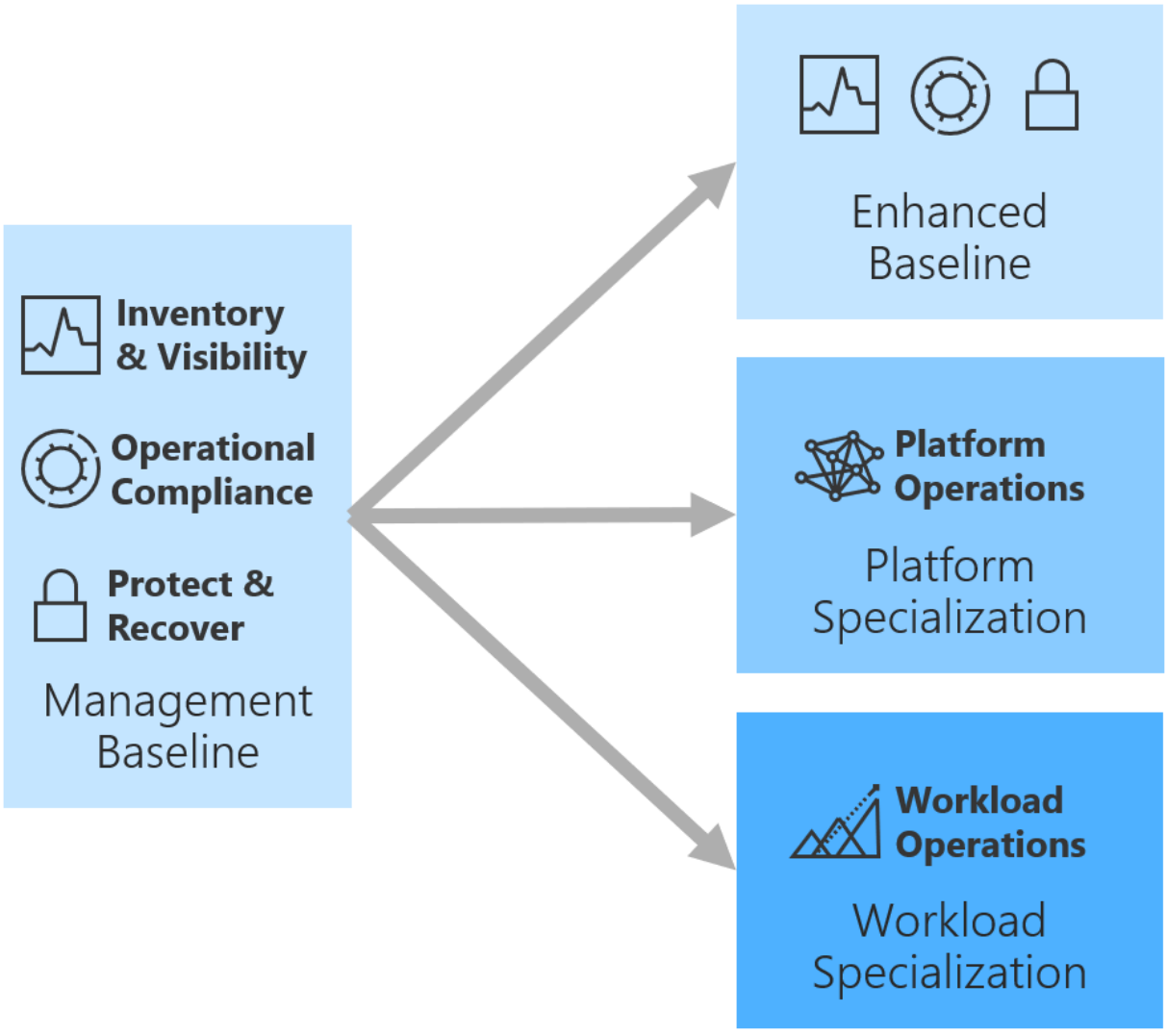
Enhanced management baseline in Azure

5/18/2021 • 4 minutes to read • Edit Online

The first three cloud management disciplines describe a management baseline. The preceding articles in this guide outline a minimum viable product (MVP) for cloud management services, which is referred to as a management baseline. This article outlines a few common improvements to the baseline.

The purpose of a management baseline is to create a consistent offering that provides a minimum level of business commitment for **all** supported workloads. With this baseline of common, repeatable management offerings, the team can deliver highly optimized operational management with minimal deviation.

However, you might need a greater commitment to the business beyond the standard offering. The following image and list show three ways to go beyond the management baseline.



- **Enhanced management baseline:**
  - Add enhancements to the management baseline, when the majority of workloads in the portfolio have a shared requirement.
  - Improved business commitments using additional cloud-native operations tools and processes.
  - Baseline enhancements should have no impact on the architecture of specific workloads.
- **Workload operations:**

- Largest per-workload operations investment.
- Highest degree of resiliency.
- Suggested for the approximately 20 percent of workloads that drive business value.
- Typically reserved for high-criticality or mission-critical workloads.

- **Platform operations:**
  - Operations investment is spread across many workloads.
  - Resiliency improvements affect all workloads that use the defined platform.
  - Suggested for the approximately 20 percent of platforms that have highest criticality.
  - Typically reserved for medium-criticality to high-criticality workloads.

Both workload operations and platform operations require changes to design and architecture principles. Those changes can take time and might result in increased operating expenses. To reduce the number of workloads that require such investments, an enhanced management baseline can provide enough of an improvement to the business commitment.

This table outlines a few processes, tools, and potential effects common in customers' enhanced management baselines:

DISCIPLINE	PROCESS	TOOL	POTENTIAL IMPACT	LEARN MORE
Inventory and visibility	Service change tracking	Azure Resource Graph	Greater visibility into changes to Azure services might help detect negative effects sooner or remediate faster.	Overview of Azure Resource Graph
Inventory and visibility	IT Service Management (ITSM) integration	IT Service Management Connector	Automated ITSM connection creates awareness sooner.	IT Service Management Connector (ITSMC)
Operational compliance	Operations automation	Azure Automation	Automate operational compliance for faster and more accurate response to change.	See the following sections
Operational compliance	Performance automation	Azure Automation	Automate operational compliance with performance expectations to resolve common resource specific scaling or sizing issues.	See the following sections
Operational compliance	Multicloud operations	Azure Automation Hybrid Runbook Worker	Automate operations across multiple clouds.	Hybrid Runbook Worker overview
Operational compliance	Guest automation	Desired State Configuration (DSC)	Code-based configuration of guest operating systems to reduce errors and configuration drift.	DSC overview

DISCIPLINE	PROCESS	TOOL	POTENTIAL IMPACT	LEARN MORE
Protect and recover	Breach notification	Azure Security Center	Extend protection to include security-breach recovery triggers.	See the following sections

Azure Automation

- Azure Automation

Azure Automation provides a centralized system for the management of automated controls. In Azure Automation, you can run simple remediation, scale, and optimization processes in response to environmental metrics. These processes reduce the overhead associated with manual incident processing.

Most importantly, automated remediation can be delivered in near-real-time, significantly reducing interruptions to business processes. A study of the most common business interruptions identifies activities within your environment that could be automated.

Runbooks

The basic unit of code for delivering automated remediation is a runbook. Runbooks contain the instructions for remediating or recovering from an incident.

To create or manage runbooks:

1. Go to **Azure Automation**.
2. Select **Automation accounts** and choose one of the listed accounts.
3. Go to **Process automation**.
4. With the options presented, you can create or manage runbooks, schedules, and other automated remediation functionality.

GO TO AZURE AUTOMATION

Azure Security Center

- Azure Security Center

Azure Security Center also plays an important part in your protect-and-recover strategy. It can help you monitor the security of your machines, networks, storage, data services, and applications.

Azure Security Center provides advanced threat detection by using machine learning and behavioral analytics to help identify active threats targeting your Azure resources. It also provides threat protection that blocks malware and other unwanted code, and it reduces the surface area exposed to brute force and other network attacks.

When Azure Security Center identifies a threat, it triggers a security alert with steps you need for responding to an attack. It also provides a report with information about the detected threat.

Azure Security Center is offered in two tiers: Free and Standard. Features like security recommendations are available in the Free tier. The Standard tier provides additional protection like advanced threat detection and protection across hybrid cloud workloads.

Action

Try Standard tier for free for your first 30 days

After you enable and configure security policies for a subscription's resources, you can view the security state of

your resources and any issues on the **Prevention** pane. You can also view a list of those issues on the **Recommendations** tile.

EXPLORE AZURE SECURITY CENTER

To explore Azure Security Center, go to the Azure portal.

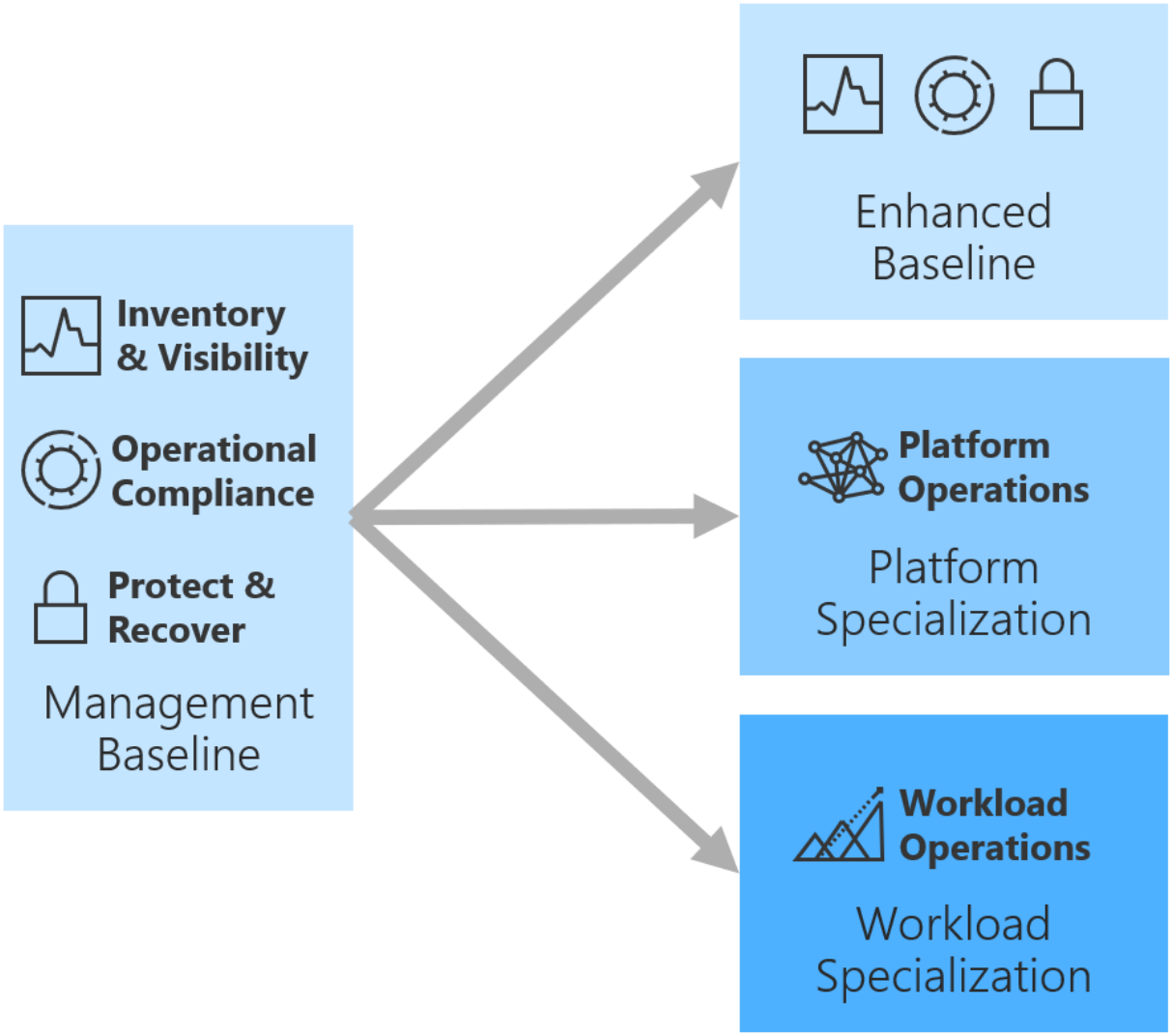
Learn more

To learn more, see Azure Security Center documentation.

Platform specialization for cloud management

5/4/2021 • 5 minutes to read • Edit Online

Much like the enhanced management baseline, platform specialization is extension beyond the standard management baseline. See the following image and list that show the ways to expand the management baseline. This article addresses the platform specialization options.



- **Workload operations:** The largest per-workload operations investment and the highest degree of resiliency. We suggest workload operations for the approximately 20% of workloads that drive business value. This specialization is usually reserved for high criticality or mission-critical workloads.
- **Platform operations:** Operations investment is spread across many workloads. Resiliency improvements affect all workloads that use the defined platform. We suggest platform operations for the approximately 20% of platforms that have the highest criticality. This specialization is usually reserved for medium to high criticality workloads.
- **Enhanced management baseline:** The relatively lowest operations investment. This specialization slightly improves business commitments by using additional cloud-native operations tools and processes.

Both workload and platform operations require changes to design and architecture principles. Those changes can take time and might result in increased operating expenses. To reduce the number of workloads requiring such investments, an enhanced management baseline might provide enough of an improvement to the

business commitment.

This table outlines a few common processes, tools, and potential effects common in customers' enhanced management baselines:

PROCESS	TOOL	PURPOSE	SUGGESTED MANAGEMENT LEVEL
Improve system design	Microsoft Azure Well-Architected Framework	Improving the architectural design of the platform to improve operations	N/A
Automate remediation	Azure Automation	Responding to advanced platform data with platform-specific automation	Platform operations
Service catalog	Managed applications center	Providing a self-service catalog of approved solutions that meet organizational standards	Platform operations
Container performance	Azure Monitor for containers	Monitoring and diagnostics of containers	Platform operations
Platform as a service (PaaS) data performance	Azure SQL Analytics	Monitoring and diagnostics for PaaS databases	Platform operations
Infrastructure as a service (IaaS) data performance	SQL Server Health Check	Monitoring and diagnostics for IaaS databases	Platform operations

High-level process

Platform specialization consists of a disciplined execution of the following four processes in an iterative approach. Each process is explained in more detail in later sections of this article.

- **Improve system design:** Improve the design of common systems or platforms to effectively minimize interruptions.
- **Automate remediation:** Some improvements aren't cost effective. In such cases, it might make more sense to automate remediation and reduce the effect of interruptions.
- **Scale the solution:** As systems design and automated remediation are improved, those changes can be scaled across the environment through the service catalog.
- **Continuous improvement:** Different monitoring tools can be used to discover incremental improvements. These improvements can be addressed in the next pass of system design, automation, and scale.

Improve system design

- Improve system design

Improving system design is the most effective approach to improving operations of any common platform. Through system-design improvements, stability can increase and business interruptions can decrease. Design of individual systems is beyond the scope of the environment view that's taken throughout the Cloud Adoption Framework.

As a complement to this framework, the Microsoft Azure Well-Architected Framework provides guiding tenets for improving the quality of a platform or a specific workload. The framework focuses on improvement across

five pillars of architecture excellence:

- **Cost optimization:** Manage costs to maximize the value delivered.
- **Operational excellence:** Follow operational processes that keep a system running in production.
- **Performance efficiency:** Scale systems to adapt to changes in load.
- **Reliability:** Design systems to recover from failures and continue to function.
- **Security:** Protect applications and data from threats.

Technical debt and architectural flaws cause most business interruptions. For existing deployments, you can view system-design improvements as payments against existing technical debt. For new deployments, you can view those improvements as avoidance of technical debt.

The following **Automated remediation** tab shows ways to remediate technical debt that can't or shouldn't be addressed.

Learn more about the Microsoft Azure Well-Architected Framework to improve system design.

As system design improves, return to this article to find new opportunities to improve and scale those improvements across your environment.

Automated remediation

- Automated remediation

Some technical debt can't be addressed. Resolution might be too expensive to correct or might be planned but have a long project duration. The business interruption might not have a significant business effect. Or the business priority might be to recover quickly instead of investing in resiliency.

When resolution of technical debt isn't the desired approach, automated remediation is commonly the next step. Using Azure Automation and Azure Monitor to detect trends and provide automated remediation is the most common approach to automated remediation.

For guidance on automated remediation, see Azure Automation and alerts.

Scale the solution with a service catalog

- Scale the solution with a service catalog

A well-managed service catalog is the cornerstone of platform specialization and platform operations. Use of a catalog is how improvements to systems design and remediation are scaled across an environment.

The cloud platform team and cloud automation team align to create repeatable solutions to the most common platforms in any environment. But if those solutions aren't consistently used, cloud management can provide little more than a baseline offering.

To maximize adoption and minimize maintenance overhead of any optimized platform, you should add the platform to an Azure service catalog. You can deploy each application in the catalog for internal consumption via the service catalog or as a marketplace offering for external consumers.

For instructions on publishing to a service catalog, see the article series on publishing to a service catalog.

Deploy applications from the service catalog

1. In the Azure portal, go to **Managed applications center (preview)**.
2. On the **Browse** pane, select **Service Catalog applications**.
3. Select **+ Add** to choose an application definition from your company's service catalog.

Any managed applications you're servicing are displayed.

GO TO VIRTUAL MACHINES

Manage service catalog applications

1. In the Azure portal, go to **Managed applications center (preview)**.
2. On the **Service** pane, select **Service Catalog applications**.

Any managed applications you're servicing are displayed.

GO TO VIRTUAL MACHINES

Continuous improvement

- Continuous improvement

Platform specialization and platform operations both depend on strong feedback loops among adoption, platform, automation, and management teams. Grounding those feedback loops in data helps each team make wise decisions. For platform operations to achieve long-term business commitments, it's important to use insights specific to the centralized platform.

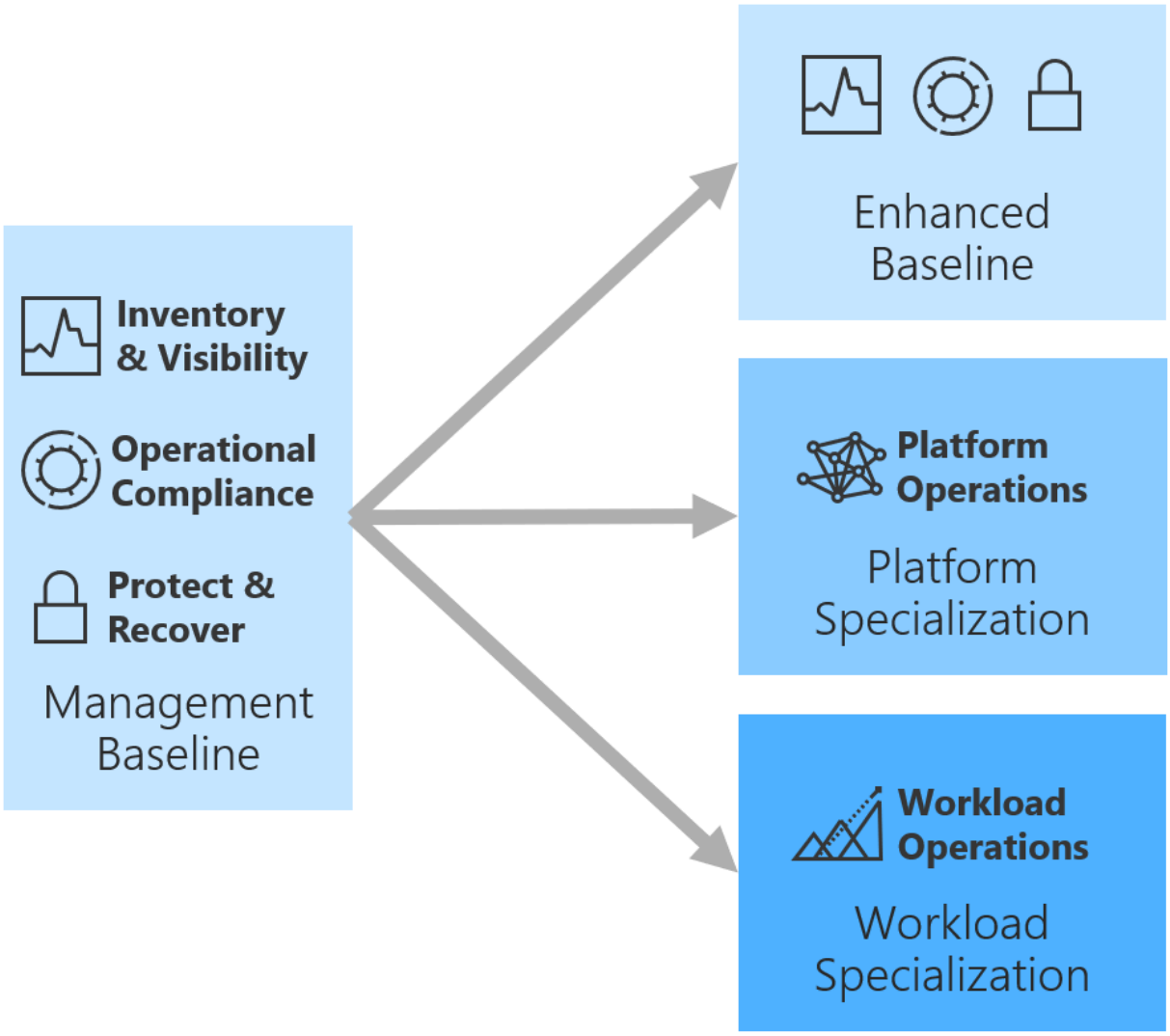
Containers and SQL Server are the two most common centrally managed platforms. These articles can help you get started with continuous-improvement data collection on those platforms:

- Container performance
- PaaS database performance
- IaaS database performance

Workload specialization for cloud management

5/4/2021 • 2 minutes to read • Edit Online

Workload specialization builds on the concepts outlined in Platform specialization.



- **Workload operations:** The largest per-workload operations investment and highest degree of resiliency. We suggest workload operations for the approximately 20% of workloads that drive business value. This specialization is usually reserved for high criticality or mission-critical workloads.
- **Platform operations:** Operations investment is spread across many workloads. Resiliency improvements affect all workloads that use the defined platform. We suggest platform operations for the approximately 20% of platforms that have the highest criticality. This specialization is usually reserved for medium to high criticality workloads.
- **Enhanced management baseline:** The relatively lowest operations investment. This specialization slightly improves business commitments by using additional cloud-native operations tools and processes.

High-level process

Workload specialization consists of a disciplined execution of the following four processes in an iterative approach. Each process is explained in more detail in Platform specialization.

- **Improve system design:** Improve the design of a specific workload to effectively minimize interruptions.
- **Automate remediation:** Some improvements aren't cost effective. In such cases, it might make more sense to automate remediation and reduce the effect of interruptions.
- **Scale the solution:** As you improve systems design and automated remediation, you can scale those changes across the environment through the service catalog.
- **Continuous improvement:** You can use different monitoring tools to discover incremental improvements. These improvements can be addressed in the next pass of system design, automation, and scale.

Cultural change

Workload specialization often triggers a cultural change in traditional IT build processes that focus on delivering a management baseline, enhanced baselines, and platform operations. Those types of offerings can be scaled across the environment. Workload specialization is similar in execution to platform specialization. But unlike common platforms, the specialization required by individual workloads often doesn't scale.

When workload specialization is required, operational management commonly evolves beyond a centralized IT perspective. The approach suggested in Cloud Adoption Framework is a distribution of cloud management functionality.

In this model, operational tasks like monitoring, deployment, DevOps, and other innovation-focused functions shift to an application-development or business-unit organization. The cloud platform team and the core cloud monitoring team still delivers on the management baseline across the environment.

Those centralized teams also guide and instruct workload-specialized teams on operations of their workloads. But the day-to-day operational responsibility falls on a cloud management team that is managed outside of IT. This type of distributed control is one of the primary indicators of maturity in a cloud center of excellence.

Beyond platform specialization: Application Insights

Greater detail on the specific workload is required to provide clear workload operations. During the continuous improvement phase, Application Insights will be a necessary addition to the cloud management toolchain.

REQUIREMENT	TOOL	PURPOSE
Application monitoring	Application Insights	Monitoring and diagnostics for applications
Performance, availability, and usage	Application Insights	Advanced application monitoring with the application dashboard, composite maps, usage, and tracing

Deploy Application Insights

1. In the Azure portal, go to **Application Insights**.
2. Select **+ Add** to create an Application Insights resource to monitor your live web application.
3. Follow the on-screen prompts.

See the Azure Monitor Application Insights hub for guidance on configuring your application for monitoring.

CREATE APPLICATION INSIGHT RESOURCES

Monitor performance, availability, and usage

1. In the Azure portal, search for **Application Insights**.
2. Choose one of the Application Insights resources from the list.

Application Insights contains different kinds of options for monitoring performance, availability, usage, and dependencies. Each of these views of the application data provides clarity into the continuous-improvement feedback loop.

MONITOR APPLICATIONS

Establish operational management practices in the cloud

5/4/2021 • 2 minutes to read • Edit Online

Cloud adoption is a catalyst for enabling business value. However, real business value is realized through ongoing, stable operations of the technology assets deployed to the cloud. This section of the Cloud Adoption Framework guides you through various transitions into operational management in the cloud.

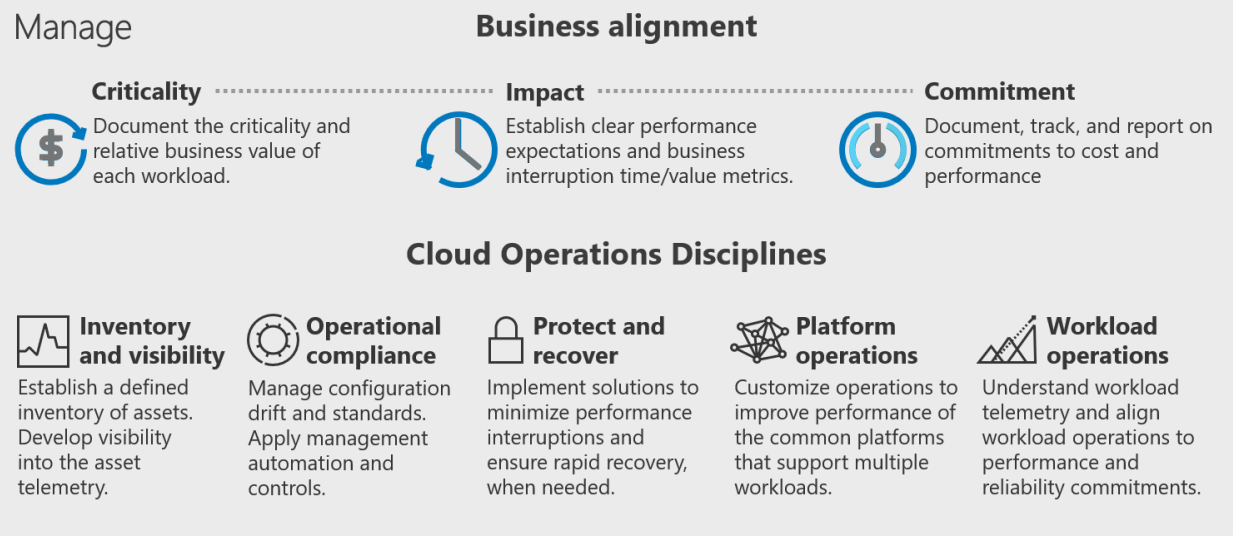
Actionable best practices

Modern operations management solutions create a multicloud view of operations. Assets managed through the following best practices may live in the cloud, in an existing datacenter, or even in a competing cloud provider. Currently, the framework includes two best-practices references to guide operations management maturity in the cloud:

- Azure server management: An onboarding guide to incorporating the cloud-native tools and services needed to manage operations.
- Hybrid monitoring: Many customers have already made a substantial investment in System Center Operations Manager. For those customers, this guide to hybrid monitoring can help them compare and contrast the cloud-native reporting tools with Operations Manager tooling. This comparison makes it easier to decide which tools to use for operational management.

Cloud operations

Both of these best practices build toward a future-state methodology for operations management, as illustrated in the following diagram:



Business alignment: In the Manage methodology, all workloads are classified by criticality and business value. That classification can then be measured through an impact analysis, which calculates the lost value associated with performance degradation or business interruptions. Using that tangible revenue impact, cloud operations teams can work with the business to establish a commitment that balances cost and performance.

Cloud operations disciplines: After the business is aligned, it's much easier to track and report on the proper disciplines of cloud operations for each workload. Making decisions along each discipline can then be converted to commitment terms that can be easily understood by the business. This collaborative approach makes the

business stakeholder a partner in finding the right balance between cost and performance.

- **Inventory and visibility:** At a minimum, operations management requires a means of inventorying assets and creating visibility into the run state of each asset.
- **Operational compliance:** Regular management of configuration, sizing, cost, and performance of assets is key to maintaining performance expectations.
- **Protect and recover:** Minimizing operational interruptions and expediting recovery help the business avoid performance losses and adverse revenue impacts. Detection and recovery are essential aspects of this discipline.
- **Platform operations:** All IT environments contain a set of commonly used platforms. Those platforms could include data stores such as SQL Server or Azure HDInsight. Other common platforms could include container solutions such as Azure Kubernetes Service (AKS). Regardless of the platform, platform operations maturity focuses on customizing operations based on how the common platforms are deployed, configured, and used by workloads.
- **Workload operations:** At the highest level of operational maturity, cloud operations teams can tune operations for critical workloads. For those workloads, available data can assist in automating the remediation, sizing, or protection of workloads based on their utilization.

Additional guidance, such as the Microsoft Azure Well-Architected Framework, can help you make detailed architectural decisions about each workload, within the previously described disciplines.

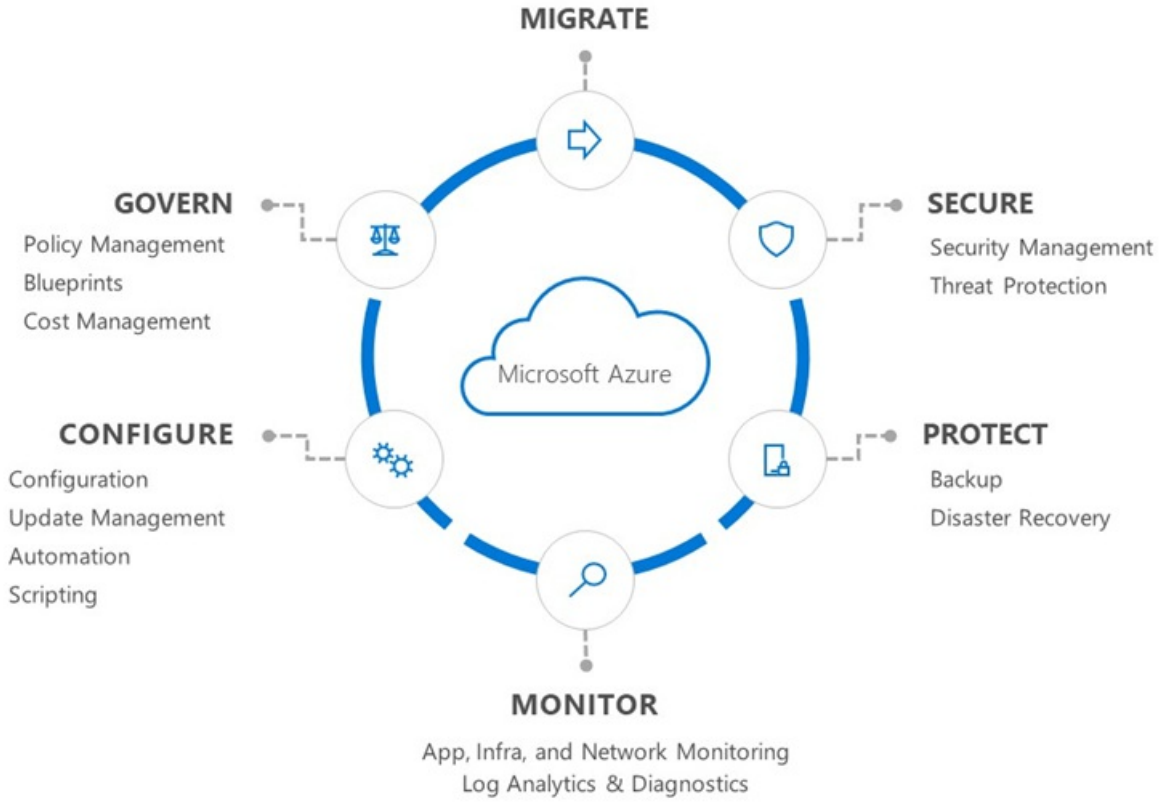
This section of the Cloud Adoption Framework will build on each of the preceding topics to help promote mature cloud operations within your organization.

Overview of Azure server management services

5/4/2021 • 2 minutes to read • Edit Online

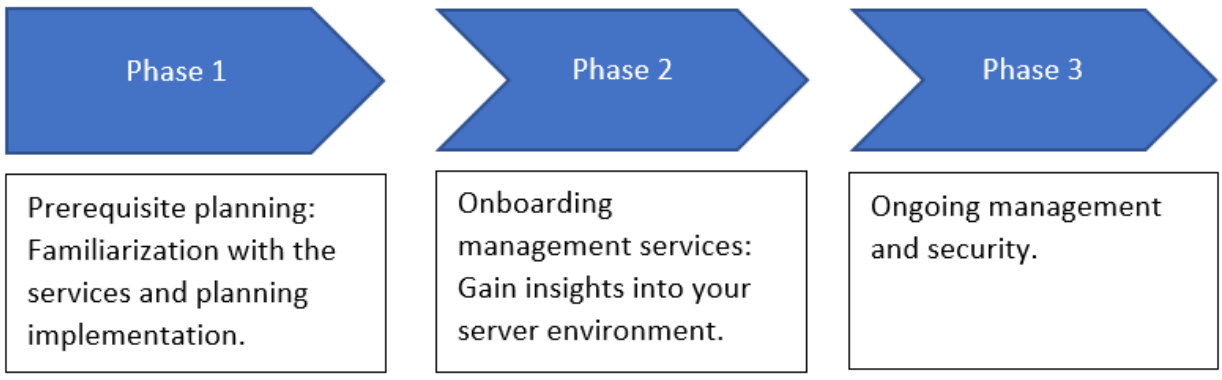
Azure server management services provide a consistent experience for managing servers at scale. These services cover both Linux and Windows operating systems. They can be used in production, development, and test environments. The server management services can support Azure IaaS virtual machines, physical servers, and virtual machines that are hosted on-premises or in other hosting environments.

The Azure server management services suite includes the services in the following diagram:



This section of the Microsoft Cloud Adoption Framework provides an actionable and prescriptive plan for deploying server management services in your environment. This plan helps orient you quickly to these services, guiding you through an incremental set of management stages for all environment sizes.

For simplicity, we've categorized this guidance into three stages:



Why use Azure server management services?

Azure server management services offer the following benefits:

- **Native to Azure:** Server management services are built into and natively integrated with Azure Resource Manager. These services are continuously improved to provide new features and capabilities.
- **Windows and Linux:** Windows and Linux machines get the same consistent management experience.
- **Hybrid:** The server management services cover Azure IaaS virtual machines as well as physical and virtual servers that are hosted on-premises or in other hosting environments.
- **Security:** Microsoft devotes substantial resources to all forms of security. This investment not only protects the Azure infrastructure but also extends the resulting technologies and expertise to protect customers' resources wherever they reside.

Next steps

Familiarize yourself with the tools, services, and planning involved with adopting the Azure server management suite.

Prerequisite tools and planning

Phase 1: Prerequisite planning for Azure server management services

5/4/2021 • 6 minutes to read • Edit Online

In this phase, you'll become familiar with the Azure server management suite of services, and plan how to deploy the resources needed to implement these management solutions.

Understand the tools and services

Review Azure server management tools and services for a detailed overview of:

- The management areas that are involved in ongoing Azure operations.
- The Azure services and tools that help support you in these areas.

You'll use several of these services together to meet your management requirements. These tools are referenced often throughout this guidance.

The following sections discuss the planning and preparation required to use these tools and services.

Log Analytics workspace and Automation account planning

Many of the services you'll use to onboard Azure management services require a Log Analytics workspace and a linked Azure Automation account.

A Log Analytics workspace is a unique environment for storing Azure Monitor log data. Each workspace has its own data repository and configuration. Data sources and solutions are configured to store their data in particular workspaces. Azure monitoring solutions require all servers to be connected to a workspace, so that their log data can be stored and accessed.

Some of the management services require an Azure Automation account. You use this account, and the capabilities of Azure Automation, to integrate Azure services and other public systems to deploy, configure, and manage your server management processes.

The following Azure server management services require a linked Log Analytics workspace and Automation account:

- Update Management
- Change Tracking and Inventory
- Hybrid Runbook Worker
- Desired State Configuration

The second phase of this guidance focuses on deploying services and automation scripts. It shows you how to create a Log Analytics workspace and an Automation account. This guidance also shows you how to use Azure Policy to ensure that new virtual machines are connected to the correct workspace.

The examples in this guidance assume a deployment that doesn't already have servers deployed to the cloud. To learn more about the principles and considerations involved in planning your workspaces, see Manage log data and workspaces in Azure Monitor.

Planning considerations

When preparing the workspaces and accounts that you need for onboarding management services, consider the

following issues:

- **Azure geographies and regulatory compliance:** Azure regions are organized into *geographies*. An Azure geography ensures that data residency, sovereignty, compliance, and resiliency requirements are honored within geographical boundaries. If your workloads are subject to data-sovereignty or other compliance requirements, workspace and Automation accounts must be deployed to regions within the same Azure geography as the workload resources they support.
- **Number of workspaces:** As a guiding principle, create the minimum number of workspaces required per Azure geography. We recommend at least one workspace for each Azure geography where your compute or storage resources are located. This initial alignment helps avoid future regulatory issues when you migrate data to different geographies.
- **Data retention and capping:** You may also need to take data retention policies or data capping requirements into consideration when creating workspaces or Automation accounts. For more information about these principles, and for additional considerations when planning your workspaces, see Manage log data and workspaces in Azure Monitor.
- **Region mapping:** Linking a Log Analytics workspace and an Azure Automation account is supported only between certain Azure regions. For example, if the Log Analytics workspace is hosted in the `East US` region, the linked Automation account must be created in the `East US 2` region to be used with management services. If you have an Automation account that was created in another region, it can't link to a workspace in `East US`. The choice of deployment region can significantly affect Azure geography requirements. Consult the region mapping table to decide which region should host your workspaces and Automation accounts.
- **Workspace multihoming:** The Azure Log Analytics agent supports multihoming in some scenarios, but the agent faces several limitations and challenges when running in this configuration. Unless Microsoft has recommended it for your specific scenario, don't configure multihoming on the Log Analytics agent.

Resource placement examples

There are several different models for choosing the subscription in which you place the Log Analytics workspace and Automation account. In short, place the workspace and Automation accounts in a subscription owned by the team that's responsible for implementing the Update Management solution and the Change Tracking and Inventory service.

The following are examples of some ways to deploy workspaces and Automation accounts.

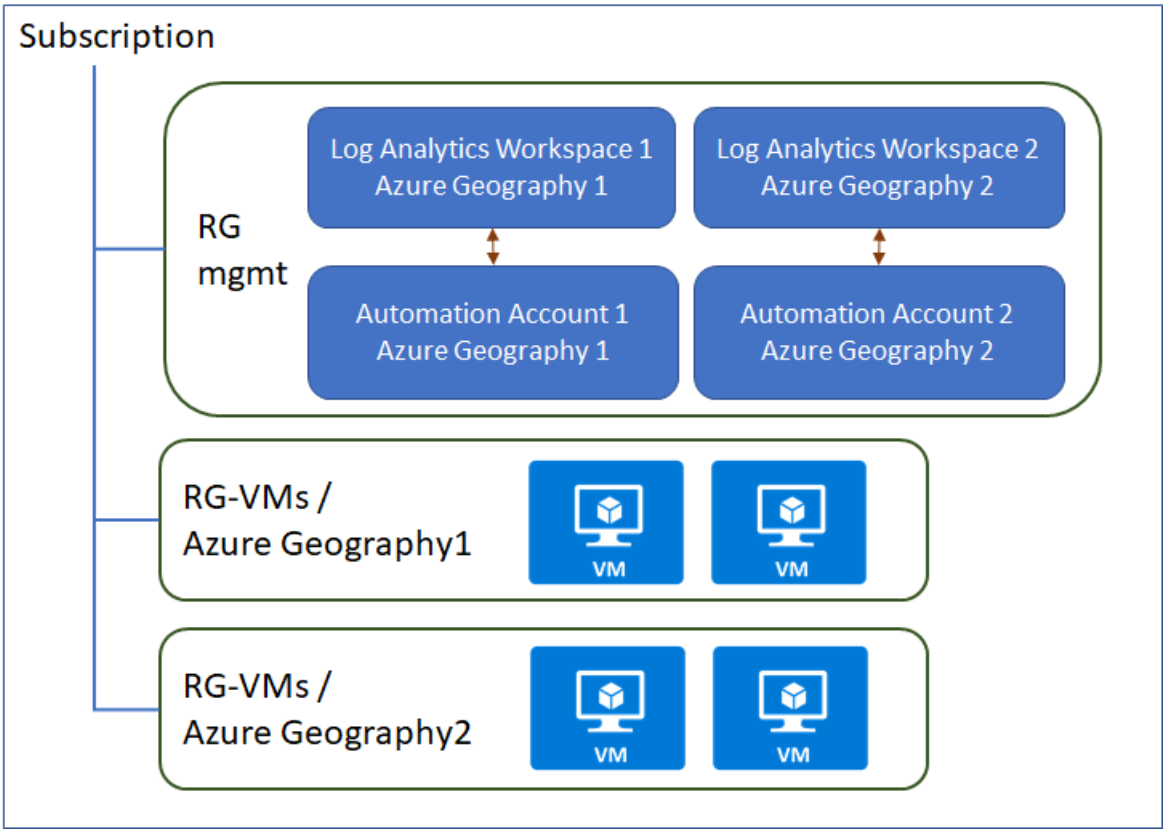
Placement by geography

Small and midsize environments have a single subscription and several hundred resources that span multiple Azure geographies. For these environments, create one Log Analytics workspace and one Azure Automation account in each geography.

You can create a workspace and an Azure Automation account, as one pair, in each resource group. Then, deploy the pair in the corresponding geography to the virtual machines.

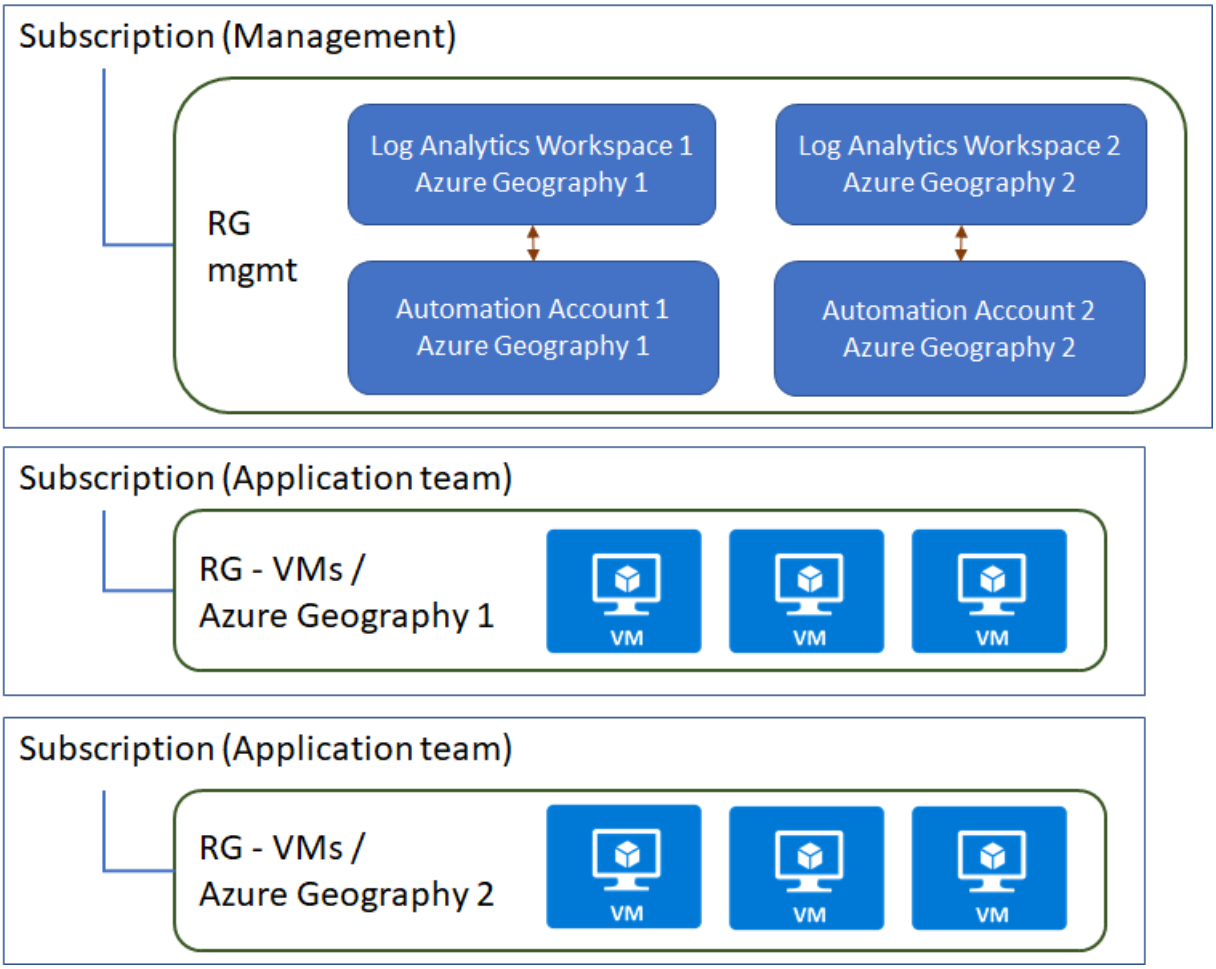
Alternatively, if your data-compliance policies don't dictate that resources reside in specific regions, you can create one pair to manage all the virtual machines. We also recommend that you place the workspace and Automation account pairs in separate resource groups to provide more granular Azure role-based access control (Azure RBAC).

The example in the following diagram has one subscription with two resource groups, each located in a different geography:



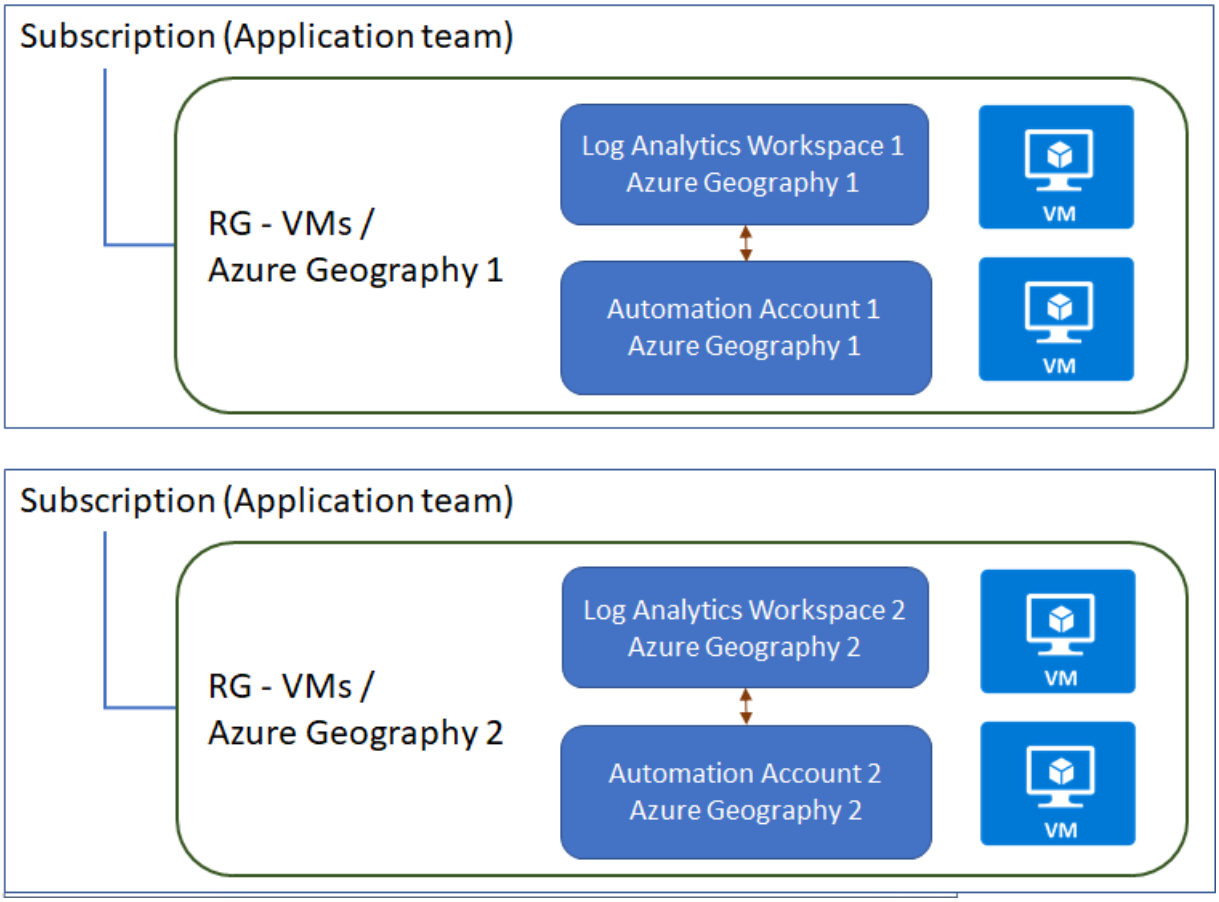
Placement in a management subscription

Larger environments span multiple subscriptions and have a central IT team that owns monitoring and compliance. For these environments, create pairs of workspaces and Automation accounts in an IT management subscription. In this model, virtual-machine resources in a geography store their data in the corresponding geography workspace in the IT management subscription. If application teams need to run automation tasks but don't require linked workspace and Automation accounts, they can create separate Automation accounts in their own application subscriptions.



Decentralized placement

In an alternative model for large environments, the application development team can be responsible for patching and management. In this case, place the workspace and Automation account pairs in the application team subscriptions alongside their other resources.



Create a workspace and Automation account

After you've chosen the best way to place and organize workspace and account pairs, make sure that you've created these resources before starting the onboarding process. The automation examples later in this guidance create a workspace and Automation account pair for you. However, if you want to onboard by using the Azure portal and you don't have an existing workspace and Automation account pair, you'll need to create one.

To create a Log Analytics workspace by using the Azure portal, see Create a workspace. Next, create a matching Automation account for each workspace by following the steps in Create an Azure Automation account.

> **NOTE**
>
> When you create an Automation account by using the Azure portal, the portal attempts by default to create Run As accounts for both Azure Resource Manager and the classic deployment model resources. If you don't have classic virtual machines in your environment and you're not the Co-Administrator on the subscription, the portal creates a Run As account for Resource Manager, but it generates an error when deploying the classic Run As account. If you don't intend to support classic resources, you can ignore this error.
>
> You can also create Run As accounts by using PowerShell.

Next steps

Learn how to onboard your servers to Azure server management services.

Onboard to Azure server management services

Phase 2: Onboarding Azure server management services

5/4/2021 • 2 minutes to read • Edit Online

After you're familiar with the tools and planning involved in Azure management services, you're ready for the second phase. Phase 2 provides step-by-step guidance for onboarding these services for use with your Azure resources. Start by evaluating this onboarding process before adopting it broadly in your environment.

> **NOTE**
> The automation approaches discussed in later sections of this guidance are meant for deployments that don't already have servers deployed to the cloud. They require that you have the Owner role on a subscription to create all the required resources and policies. If you've already created Log Analytics workspaces and Automation accounts, we recommend that you pass these resources in the appropriate parameters when you start the example automation scripts.

Onboarding processes

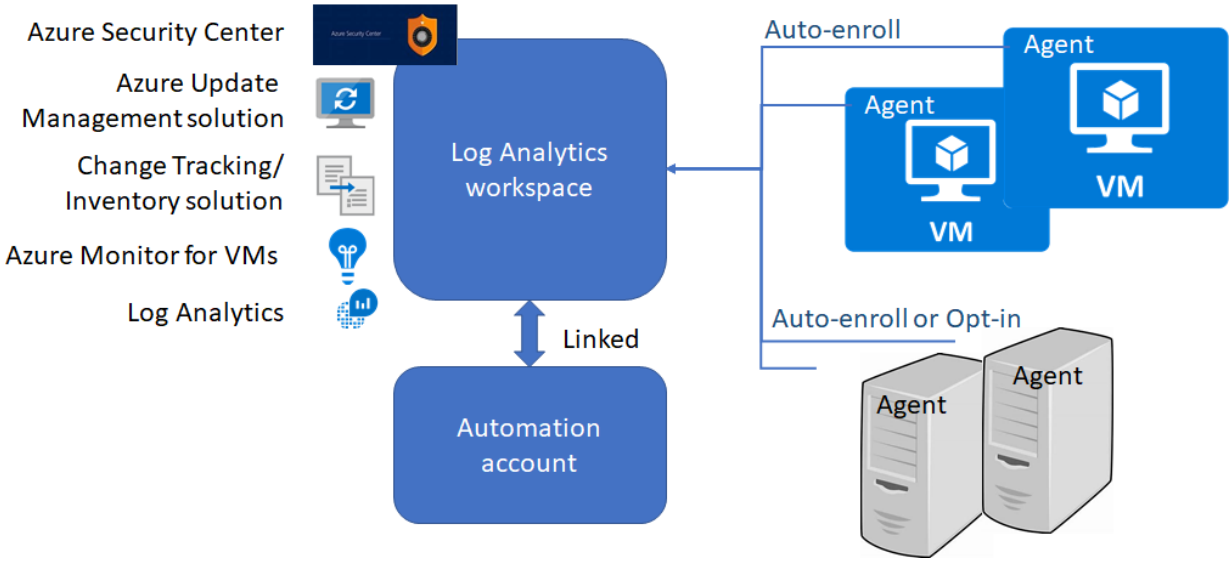
This section of the guidance covers the following onboarding processes for both virtual machines in Azure and on-premises servers:

- **Enable management services on a single VM for evaluation by using the portal.** Use this process to familiarize yourself with the Azure server management services.
- **Configure management services for a subscription by using the portal.** This process helps you configure the Azure environment so that any new VMs that are provisioned will automatically use management services. Use this approach if you prefer the Azure portal experience to scripts and command lines.
- **Configure management services for a subscription by using Azure Automation.** This process is fully automated. Just create a subscription, and the scripts will configure the environment to use management services for any newly provisioned VM. Use this approach if you're familiar with PowerShell scripts and Azure Resource Manager templates, or if you want to learn to use them.

The procedures for each of these approaches are different.

> **NOTE**
> When you use the Azure portal, the sequence of onboarding steps differs from the automated onboarding steps. The portal offers a simpler onboarding experience.

The following diagram shows the recommended deployment model for management services:



As shown in the preceding diagram, the Log Analytics agent has two configurations for on-premises servers:

- **Auto-enroll:** When the Log Analytics agent is installed on a server and configured to connect to a workspace, the solutions that are enabled on that workspace are applied to the server automatically.
- **Opt-in:** Even if the agent is installed and connected to the workspace, the solution isn't applied unless it's added to the server's scope configuration in the workspace.

Next steps

Learn how to onboard a single VM by using the portal to evaluate the onboarding process.

Onboard a single Azure VM for evaluation

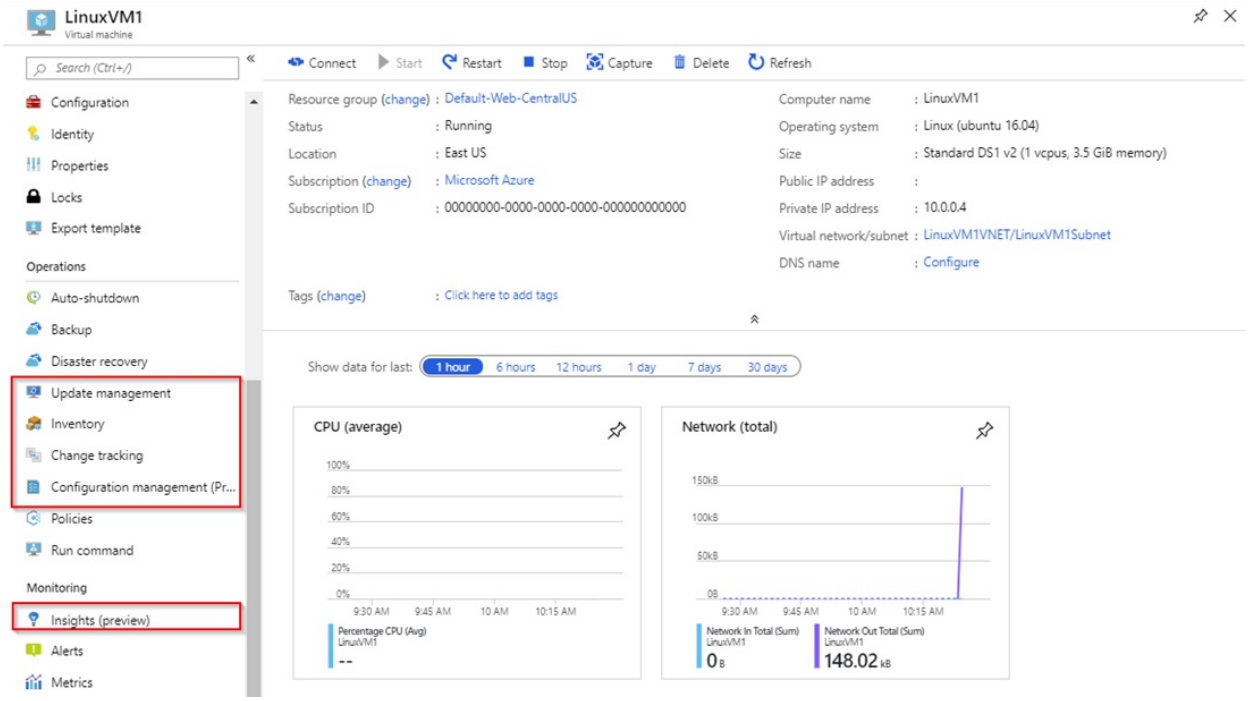
Enable server management services on a single VM for evaluation

5/4/2021 • 2 minutes to read • Edit Online

Learn how to enable server management services on a single VM for evaluation.

> **NOTE**
>
> Create the required Log Analytics workspace and Azure Automation account before you implement Azure management services on a VM.

It's simple to onboard Azure server management services to individual virtual machines in the Azure portal. You can familiarize yourself with these services before you onboard them. When you select a VM instance, all the solutions on the list of management tools and services appear on the **Operations** or **Monitoring** menu. You select a solution and follow the wizard to onboard it.



Related resources

For more information about how to onboard these solutions to individual VMs, see:

- Onboard the Update Management solution and the Change Tracking and Inventory solution for a VM in Azure
- Onboard Azure Monitor for VMs

Next steps

Learn how to use Azure Policy to onboard Azure VMs at scale.

Configure Azure management services for a subscription

Configure Azure server management services at scale

5/4/2021 • 7 minutes to read • Edit Online

You must complete these two tasks to onboard Azure server management services to your servers:

- Deploy service agents to your servers.
- Enable the management solutions.

This article covers the three processes that are necessary to complete these tasks:

1. Deploy the required agents to Azure VMs by using Azure Policy.
2. Deploy the required agents to on-premises servers.
3. Enable and configuring the solutions.

> **NOTE**
> Create the required Log Analytics workspace and Azure Automation account before you onboard virtual machines to Azure server management services.

Use Azure Policy to deploy extensions to Azure VMs

All the management solutions that are discussed in Azure management tools and services require that the Log Analytics agent is installed on virtual machines in Azure as well as on-premises servers. You can onboard your Azure VMs at scale by using Azure Policy. Assign policy to ensure that the agent is installed on your Azure VMs and connected to the correct Log Analytics workspace.

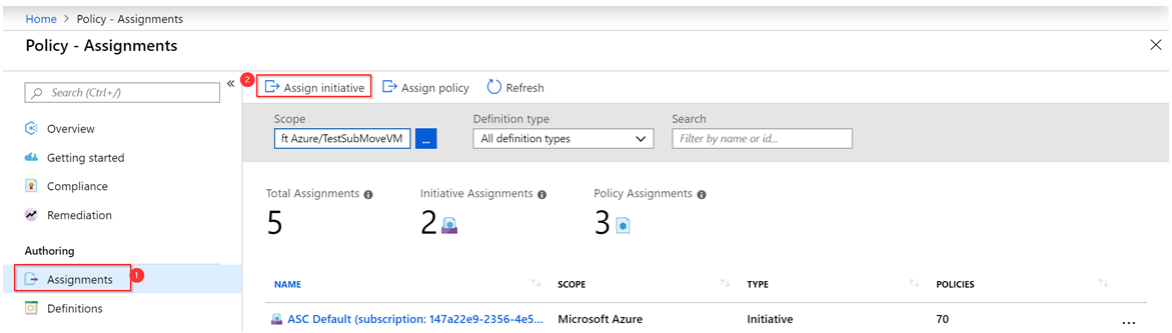
Azure Policy has a built-in policy initiative that includes the Log Analytics agent and the Microsoft Dependency Agent, which is required by Azure Monitor for VMs.

> **NOTE**
> For more information about various agents for monitoring Azure, see Overview of the Azure monitoring agents.

Assign policies

To assign the policies that described in the previous section:

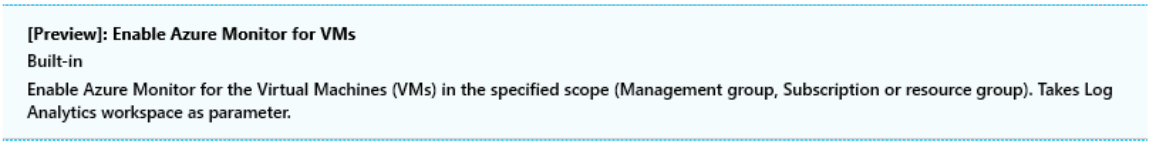
1. In the Azure portal, go to **Policy** > **Assignments** > **Assign initiative**.



2. On the **Assign policy** page, set the **Scope** by selecting the ellipsis (...) and then selecting either a

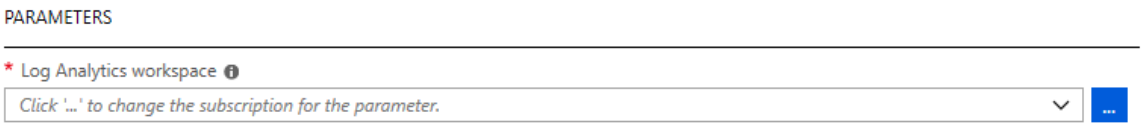
management group or subscription. Optionally, select a resource group. Then choose **Select** at the bottom of the **Scope** page. The scope determines which resources or group of resources the policy is assigned to.

3. Select the ellipsis (...) next to **Policy definition** to open the list of available definitions. To filter the initiative definitions, enter **Azure Monitor** in the **Search** box:

4. The **Assignment name** is automatically populated with the policy name that you selected, but you can change it. You can also add an optional description to provide more information about this policy assignment. The **Assigned by** field is automatically filled based on who is signed in. This field is optional, and it supports custom values.

5. For this policy, select **Log Analytics workspace** for the Log Analytics agent to associate.

6. Select the **Managed Identity location** check box. If this policy is of the type `DeployIfNotExists`, a managed identity will be required to deploy the policy. In the portal, the account will be created as indicated by the check box selection.

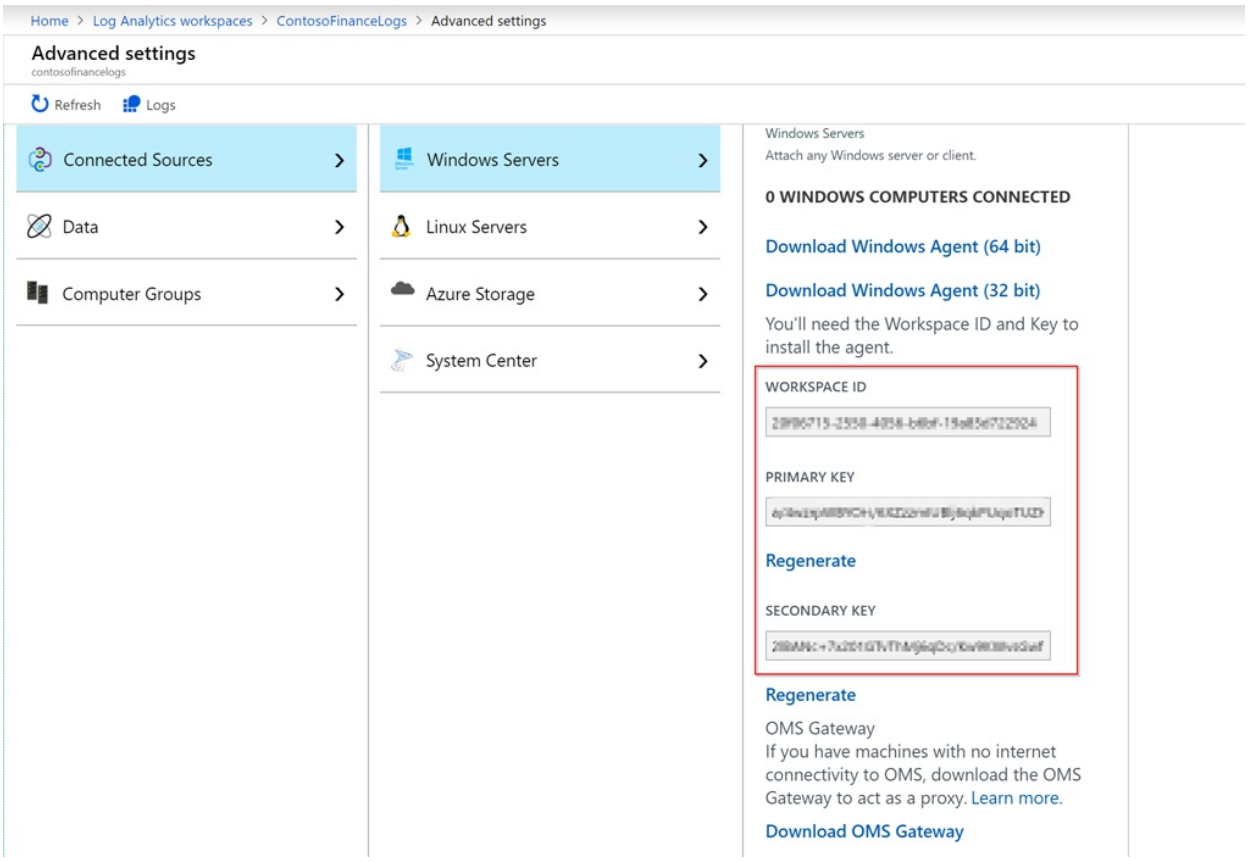
7. Select **Assign**.

After you complete the wizard, the policy assignment will be deployed to the environment. It can take up to 30 minutes for the policy to take effect. To test it, create new VMs after 30 minutes, and check if the Log Analytics agent is enabled on the VM by default.

Install agents on on-premises servers

> **NOTE**
> Create the required Log Analytics workspace and Azure Automation account before you onboard Azure server management services to servers.

For on-premises servers, you need to download and install the Log Analytics agent and the Microsoft Dependency Agent manually and configure them to connect to the correct workspace. You must specify the workspace ID and key information. To get that information, go to your Log Analytics workspace in the Azure portal, then select **Settings** > **Advanced settings**.



Enable and configure solutions

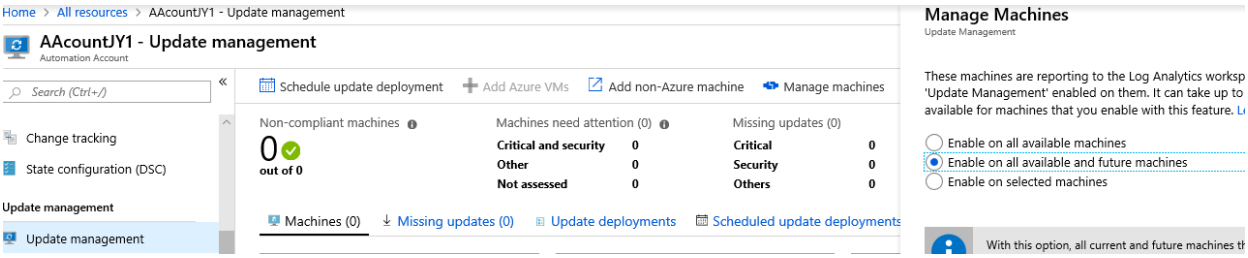
To enable solutions, you need to configure the Log Analytics workspace. Onboarded Azure VMs and on-premises servers will get the solutions from the Log Analytics workspaces that they're connected to.

Update Management

The Update Management solution and the Change Tracking and Inventory solution require both a Log Analytics workspace and an Azure Automation account. To ensure that these resources are properly configured, we recommend that you onboard through your Automation account. For more information, see Onboard the Update Management solution and the Change Tracking and Inventory solution.

We recommend that you enable the Update Management solution for all servers. Update Management is free for Azure VMs and on-premises servers. If you enable Update Management through your Automation account, a scope configuration is created in the workspace. Manually update the scope to include machines that are covered by the Update Management solution.

To cover your existing servers as well as future servers, you need to remove the scope configuration. To do this, view your Automation account in the Azure portal. Select **Update Management** > **Manage machine** > **Enable on all available and future machines**. This setting allows all Azure VMs that are connected to the workspace to use Update Management.



Change Tracking and Inventory solutions

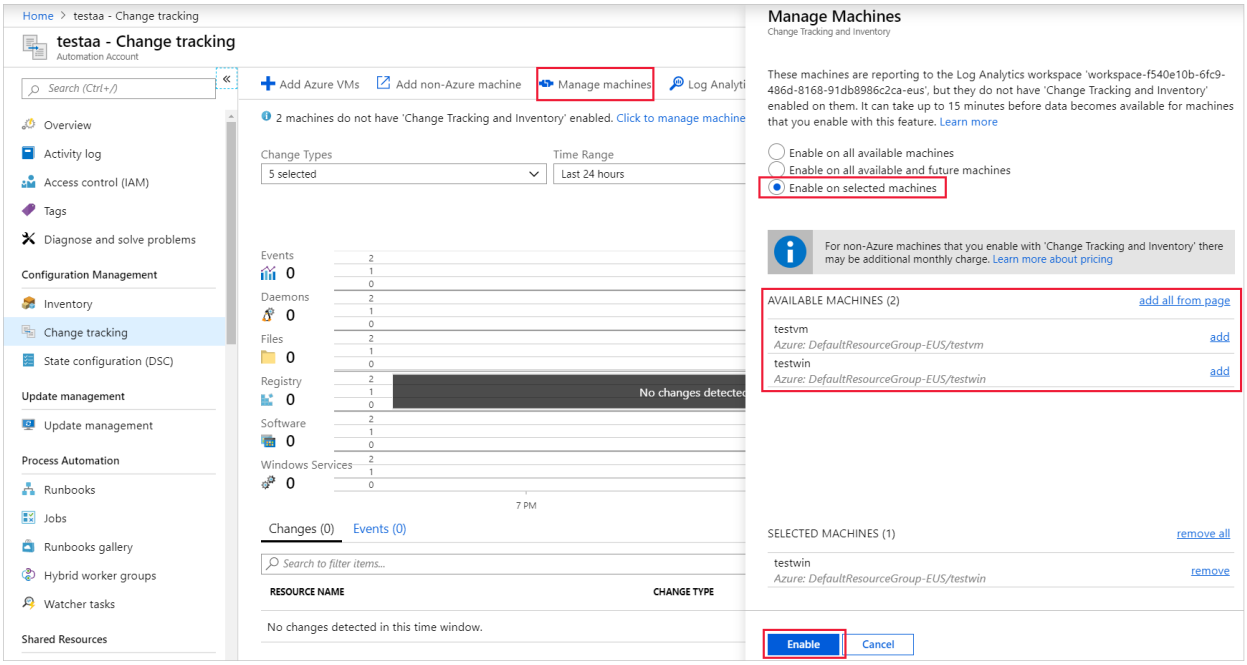
To onboard the Change Tracking and Inventory solutions, follow the same steps as for Update Management. For more information about how to onboard these solutions from your Automation account, see Onboard the

Update Management solution and the Change Tracking and Inventory solution.

The Change Tracking and Inventory solution is free for Azure VMs and costs $6 per node per month for on-premises servers. This cost covers change tracking, inventory, and Desired State Configuration. If you want to enroll only specific on-premises servers, you can opt in those servers. We recommend that you onboard all your production servers.

Opt in via the Azure portal

1. Go to the Automation account that has Change Tracking and Inventory enabled.
2. Select **Change tracking**.
3. Select **Manage machines** in the upper-right pane.
4. Select **Enable on selected machines**. Then select **Add** next to the machine name.
5. Select **Enable** to enable the solution for those machines.



Opt in by using saved searches

Alternatively, you can configure the scope configuration to opt in on-premises servers. Scope configuration uses saved searches.

To create or modify the saved search, follow these steps:

1. Go to the Log Analytics workspace that is linked to the Automation account that you configured in the preceding steps.
2. Under **General**, select **Saved searches**.
3. In the **Filter** box, enter **Change Tracking** to filter the list of saved searches. In the results, select **MicrosoftDefaultComputerGroup**.
4. Enter the computer name or the VMUUID to include the computers that you want to opt in for Change Tracking and Inventory.

```
Heartbeat
| where AzureEnvironment=~"Azure" or Computer in~ ("list of the on-premises server names", "server1")
| distinct Computer
```

> **NOTE**
> The server name must exactly match the value in the expression, and it shouldn't contain a domain name suffix.

1. Select **Save**. By default, the scope configuration is linked to the **MicrosoftDefaultComputerGroup** saved search. It will be automatically updated.

Azure activity log

Azure activity log is also part of Azure Monitor. It provides insight into subscription-level events that occur in Azure.

To implement this solution:

1. In the Azure portal, open **All services**, then select **Management + Governance** > **Solutions**.
2. In the **Solutions** view, select **Add**.
3. Search for **Activity Log Analytics** and select it.
4. Select **Create**.

You need to specify the **Workspace name** of the workspace that you created in the previous section where the solution is enabled.

Azure Log Analytics Agent Health

The Azure Log Analytics Agent Health solution reports on the health, performance, and availability of your Windows and Linux servers.

To implement this solution:

1. In the Azure portal, open **All services**, then select **Management + Governance** > **Solutions**.
2. In the **Solutions** view, select **Add**.
3. Search for **Azure Log Analytics agent health** and select it.
4. Select **Create**.

You need to specify the **Workspace name** of the workspace that you created in the previous section where the solution is enabled.

After creation is complete, the workspace resource instance displays **AgentHealthAssessment** when you select **View** > **Solutions**.

Antimalware assessment

The antimalware assessment solution helps you identify servers that are infected or at increased risk of infection by malware.

To implement this solution:

1. In the Azure portal, open **All services**, select **Management + Governance** > **Solutions**.
2. In the **Solutions** view, select **Add**.
3. Search for and then select **Antimalware Assessment**.
4. Select **Create**.

You need to specify the **Workspace name** of the workspace that you created in the previous section where the solution is enabled.

After creation is complete, the workspace resource instance displays **AntiMalware** when you select **View** > **Solutions**.

Azure Monitor for VMs

You can enable Azure Monitor for VMs through the view page for the VM instance, as described in Enable management services on a single VM for evaluation. You shouldn't enable solutions directly from the **Solutions** page as you do for the other solutions that are described in this article. For large-scale deployments, it may be easier to use automation to enable the correct solutions in the workspace.

Azure Security Center

We recommend that you onboard all your servers at least to the Free tier of Azure Security Center. This option provides basic security assessments and actionable security recommendations for your environment. The Standard tier provides additional benefits. For more information, see Azure Security Center pricing.

To enable the Free tier of Azure Security Center, follow these steps:

1. Go to the **Security Center** portal page.
2. Under **POLICY & COMPLIANCE**, select **Security policy**.
3. Find the Log Analytics workspace resource that you created in the pane on the right side.
4. Select **Edit settings** for that workspace.
5. Select **Pricing tier**.
6. Choose the **Free** option.
7. Select **Save**.

Next steps

Learn how to use automation to onboard servers and create alerts.

Automate onboarding and alert configuration

Automate onboarding

5/18/2021 • 2 minutes to read • Edit Online

To improve the efficiency of deploying Azure server management services, consider automating deployment as discussed in previous sections of this guidance. The script and the example templates provided in the following sections are starting points for developing your own automation of onboarding processes.

This guidance is supported by a GitHub repository of sample code. The repository provides example scripts and Azure Resource Manager templates to help you automate the deployment of Azure server management services.

The sample files illustrate how to use Azure PowerShell cmdlets to automate the following tasks:

- Create a Log Analytics workspace. (Or, use an existing workspace if it meets the requirements. For details, see Workspace planning.
- Create an Automation account, or use an existing account that meets the requirements. For more information, see Workspace planning.
- Link the Automation account and the Log Analytics workspace. This step isn't required if you're onboarding by using the Azure portal.
- Enable the Update Management solution and the Change Tracking and Inventory solution for the workspace.
- Onboard Azure VMs by using Azure Policy. A policy installs the Log Analytics agent and the Microsoft Dependency Agent on the Azure VMs.
- Auto-enable Azure Backup for VMs using Azure Policy
- Onboard on-premises servers by installing the Log Analytics agent on them.

The files described in the following table are used in this sample. You can customize them to support your own deployment scenarios.

FILE NAME	DESCRIPTION
`New-AMSDeployment.ps1`	The main, orchestrating script that automates onboarding. It creates resource groups, and location, workspace, and Automation accounts, if they don't exist already. This PowerShell script requires an existing subscription.
`Workspace-AutomationAccount.json`	A Resource Manager template that deploys the workspace and Automation account resources.
`WorkspaceSolutions.json`	A Resource Manager template that enables the solutions you want in the Log Analytics workspace.
`ScopeConfig.json`	A Resource Manager template that uses the opt-in model for on-premises servers with the Change Tracking and Inventory solution. Using the opt-in model is optional.
`Enable-VMInsightsPerfCounters.ps1`	A PowerShell script that enables Azure Monitor for VMs and configures performance counters.

FILE NAME	DESCRIPTION
`ChangeTracking-FileList.json`	A Resource Manager template that defines the list of files that will be monitored by change tracking.

Use the following command to run `New-AMSDeployment.ps1`:

```
.\New-AMSDeployment.ps1 -SubscriptionName '{Subscription Name}' -WorkspaceName '{Workspace Name}' -
WorkspaceLocation '{Azure Location}' -AutomationAccountName {Account Name} -AutomationAccountLocation
{Account Location}
```

Next steps

Learn how to set up basic alerts to notify your team of key management events and issues.

Set up basic alerts

Set up basic alerts

5/4/2021 • 2 minutes to read • Edit Online

A key part of managing resources is getting notified when problems occur. Alerts proactively notify you of critical conditions, based on triggers from metrics, logs, or service-health issues. As part of onboarding the Azure server management services, you can set up alerts and notifications that help keep your IT teams aware of any problems.

Azure Monitor alerts

Azure Monitor offers alerting capabilities to notify you, via email or messaging, when things go wrong. These capabilities are based on a common data-monitoring platform that includes logs and metrics generated by your servers and other resources. By using a common set of tools in Azure Monitor, you can analyze data that's combined from multiple resources and use it to trigger alerts. These triggers can include:

- Metric values.
- Log search queries.
- Activity log events.
- The health of the underlying Azure platform.
- Tests for website availability.

See the list of Azure Monitor data sources for a more detailed description of the sources of monitoring data that this service collects.

For details about manually creating and managing alerts by using the Azure portal, see the Azure Monitor documentation.

Automated deployment of recommended alerts

In this guide, we recommend that you create a set of 15 alerts for basic infrastructure monitoring. Find the deployment scripts in the Alert Toolkit GitHub repository.

This package creates alerts for:

- Low disk space
- Low available memory
- High CPU use
- Unexpected shutdowns
- Corrupted file systems
- Common hardware failures

The package uses HPE server hardware as an example. Change the settings in the associated configuration file to reflect your OEM hardware. You can also add more performance counters to the configuration file. To deploy the package, run the `New-CoreAlerts.ps1` file.

Next steps

Learn about operations and security mechanisms that support your ongoing operations.

Ongoing management and security

Phase 3: Ongoing management and security

5/4/2021 • 2 minutes to read • Edit Online

After you've onboarded Azure server management services, you'll need to focus on the operations and security configurations that will support your ongoing operations. We'll start with securing your environment by reviewing the Azure Security Center. We'll then configure policies to keep your servers in compliance and automate common tasks. This section covers the following topics:

- Address security recommendations: Azure Security Center provides suggestions to improve the security of your environment. When you implement these recommendations, you see the impact reflected in a security score.
- Enable the guest configuration policy: Use the Azure Policy Guest Configuration feature to audit the settings in a virtual machine. For example, you can check whether any certificates are about to expire.
- Track and alert on critical changes: When you're troubleshooting, the first question to consider is, "What's changed?" In this article, you'll learn how to track changes and create alerts to proactively monitor critical components.
- Create update schedules: Schedule the installation of updates to ensure that all your servers have the latest ones.
- Common Azure Policy examples: This article provides examples of common management policies.

Address security recommendations

Azure Security Center is the central place to manage security for your environment. You'll see an overall assessment and targeted recommendations.

We recommend that you review and implement the recommendations provided by this service. For information about additional benefits of Azure Security Center, see Follow Azure Security Center recommendations.

Next steps

Learn how to enable the Azure Policy Guest Configuration feature.

Guest configuration policy

Azure Policy Guest Configuration extension

5/18/2021 • 2 minutes to read • Edit Online

You can use the Azure Policy Guest Configuration extension to audit the configuration settings in a virtual machine. Guest configuration is currently supported only on Azure VMs.

To find the list of guest configuration policies, search for **Guest Configuration** on the Azure Policy portal page, or run this cmdlet in a PowerShell window to find the list:

```
Get-AzPolicySetDefinition | where-object {$_.Properties.metadata.category -eq "Guest Configuration"}
```

NOTE

Guest configuration functionality is regularly updated to support additional policy sets. Check for new supported policies periodically and evaluate whether they'll be useful.

Deployment

Use the following example PowerShell script to deploy these policies to:

- Verify that password security settings in Windows and Linux computers are set correctly.
- Verify that certificates aren't close to expiration on Windows VMs.

Before you run this script, use the `Connect-AzAccount` cmdlet to sign in. When you run the script, you must provide the name of the subscription that you want to apply the policies to.

```
    # Assign Guest Configuration policy.

    param (
        [Parameter(Mandatory=$true)]
        [string] $SubscriptionName
    )

    $Subscription = Get-AzSubscription -SubscriptionName $SubscriptionName
    $scope = "/subscriptions/" + $Subscription.Id

    $PasswordPolicy = Get-AzPolicySetDefinition -Name "3fa7cbf5-c0a4-4a59-85a5-cca4d996d5a6"
    $CertExpirePolicy = Get-AzPolicySetDefinition -Name "b6f5e05c-0aaa-4337-8dd4-357c399d12ae"

    New-AzPolicyAssignment -Name "PasswordPolicy" -DisplayName "[Preview]: Audit that password security
settings are set correctly inside Linux and Windows machines" -Scope $scope -PolicySetDefinition
$PasswordPolicy -AssignIdentity -Location eastus

    New-AzPolicyAssignment -Name "CertExpirePolicy" -DisplayName "[Preview]: Audit that certificates are not
expiring on Windows VMs" -Scope $scope -PolicySetDefinition $CertExpirePolicy -AssignIdentity -Location
eastus
```

Next steps

Learn how to enable change tracking and alerting for critical file, service, software, and registry changes.

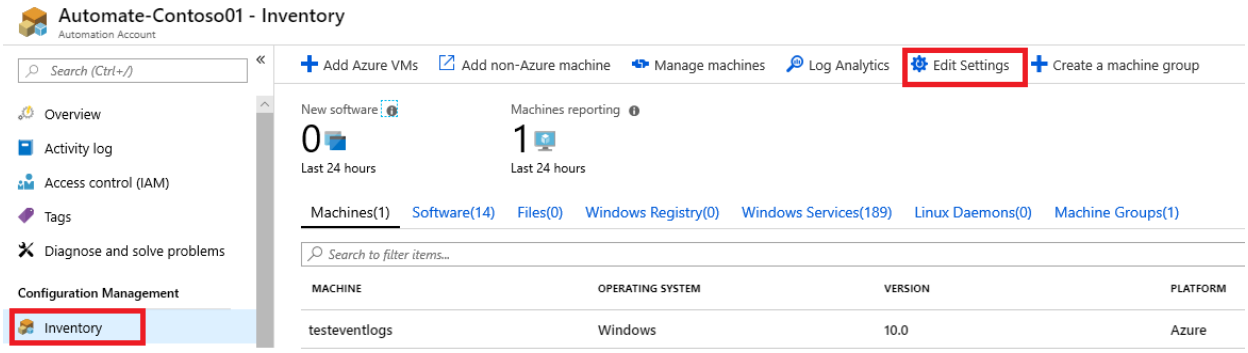
Enable tracking and alerting for critical changes

Enable tracking and alerting for critical changes

5/4/2021 • 3 minutes to read • Edit Online

Azure Change Tracking and Inventory provide alerts on the configuration state of your hybrid environment and changes to that environment. It can report critical file, service, software, and registry changes that might affect your deployed servers.

By default, the Azure Automation inventory service doesn't monitor files or registry settings. The solution does provide a list of registry keys that we recommend for monitoring. To see this list, go to your Azure Automation account in the Azure portal, then select **Inventory** > **Edit settings**.



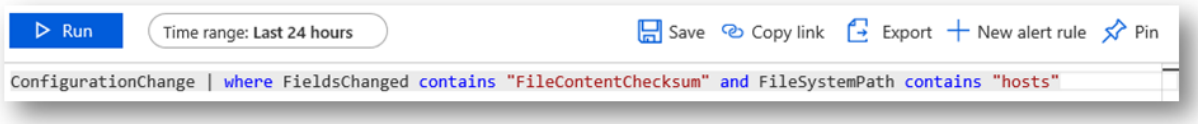
For more information about each registry key, see Registry key change tracking. Select any key to evaluate and then enable it. The setting is applied to all VMs that are enabled in the current workspace.

You can also use the service to track critical file changes. For example, you might want to track the `C:\windows\system32\drivers\etc\hosts` file because the OS uses it to map host names to IP addresses. Changes to this file could cause connectivity problems or redirect traffic to dangerous websites.

To enable file-content tracking for the hosts file, follow the steps in Enable file content tracking.

You can also add an alert for changes to files that you're tracking. For example, you might want to set an alert for changes to the hosts file. To do this, select **Log Analytics** on the command bar or **Log Search** for the linked Log Analytics workspace. In Log Analytics, use the following query to search for changes to the hosts file:

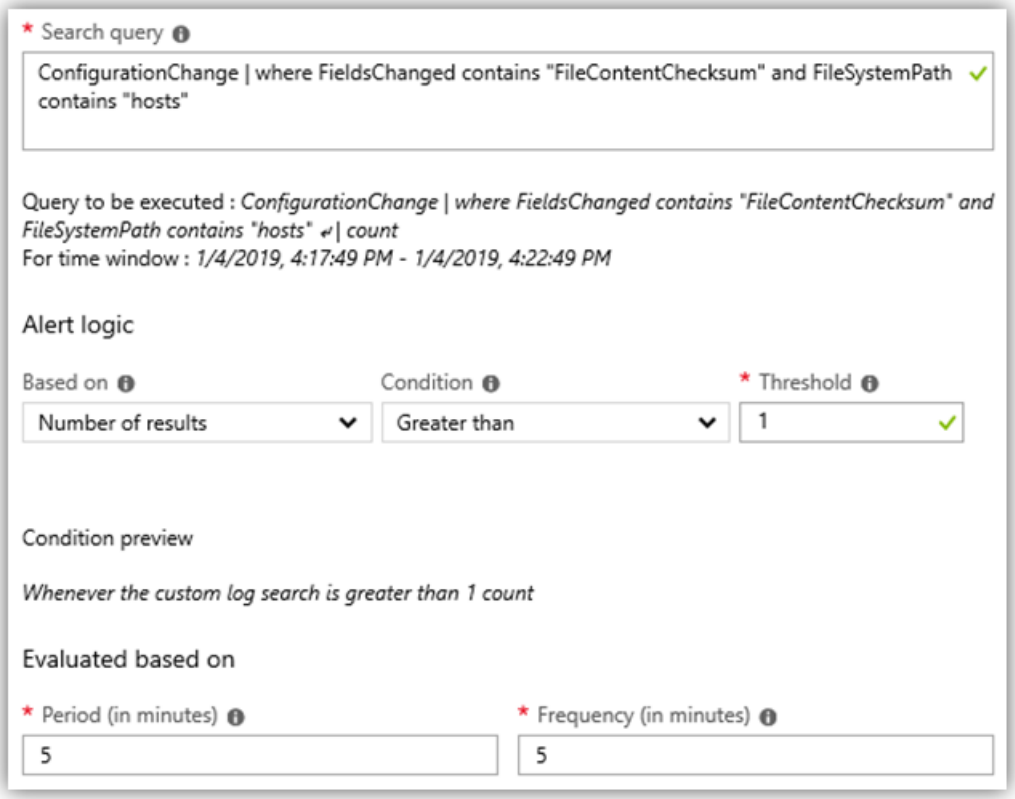
```
ConfigurationChange | where FieldsChanged contains "FileContentChecksum" and FileSystemPath contains "hosts"
```



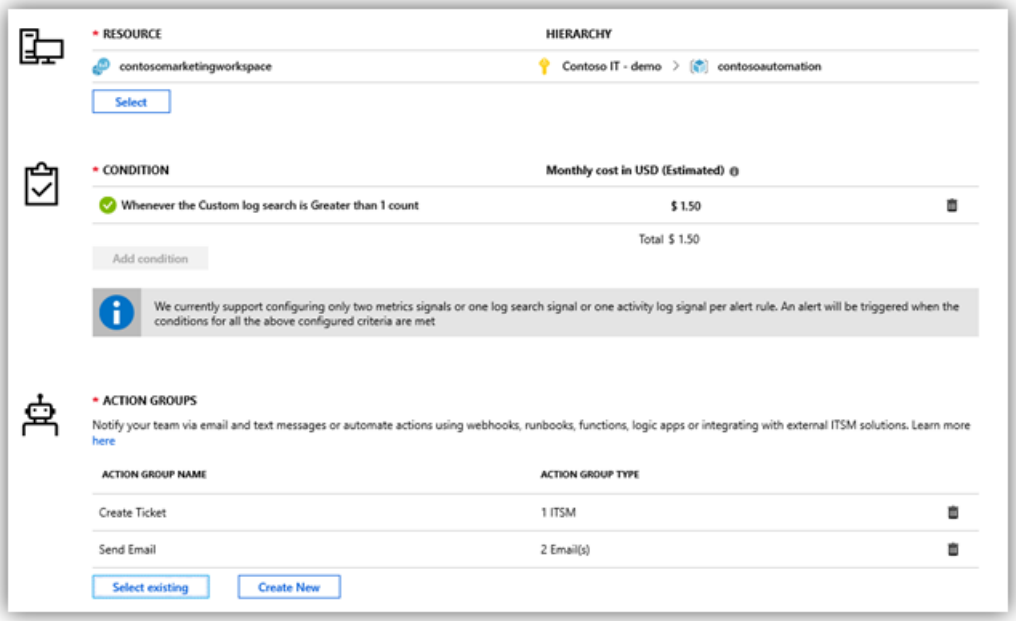
This query searches for changes to the contents of files that have a path that contains the word "hosts." You can also search for a specific file by changing the path parameter. (For example: `FileSystemPath == "c:\\windows\\system32\\drivers\\etc\\hosts"`.)

After the query returns the results, select **New alert rule** to open the alert-rule editor. You can also get to this editor via Azure Monitor in the Azure portal.

In the alert-rule editor, review the query and change the alert logic if you need to. In this case, we want the alert to be raised if any changes are detected on any machine in the environment.



After you set the condition logic, you can assign action groups to perform actions in response to the alert. In this example, when the alert is raised, emails are sent and an ITSM ticket is created. You can take many other useful actions, like triggering an Azure function, an Azure Automation runbook, a webhook, or a logic app.



After you've set all the parameters and logic, apply the alert to the environment.

Tracking and alerting examples

This section shows other common scenarios for tracking and alerting that you might want to use.

Driver file changed

Use the following query to detect if driver files are changed, added, or removed. It's useful for tracking changes to critical system files.

```
ConfigurationChange | where ConfigChangeType == "Files" and FileSystemPath contains "
c:\\windows\\system32\\drivers\\"
```

Specific service stopped

Use the following query to track changes to system-critical services.

```
ConfigurationChange | where SvcState == "Stopped" and SvcName contains "w3svc"
```

New software installed

Use the following query for environments that need to lock down software configurations.

```
ConfigurationChange | where ConfigChangeType == "Software" and ChangeCategory == "Added"
```

Specific software version is or isn't installed on a machine

Use the following query to assess security. This query references `ConfigurationData`, which contains the logs for inventory and provides the last-reported configuration state, not changes.

```
ConfigurationData | where SoftwareName contains "Monitoring Agent" and CurrentVersion != "8.0.11081.0"
```

Known DLL changed through the registry

Use the following query to detect changes to well-known registry keys.

```
ConfigurationChange | where RegistryKey == "HKEY_LOCAL_MACHINE\\System\\CurrentControlSet\\Control\\Session
Manager\\KnownDlls"
```

Next steps

Learn how Azure Automation can create update schedules to manage updates for your servers.

Create update schedules

Create update schedules

5/18/2021 • 2 minutes to read • Edit Online

You can manage update schedules by using the Azure portal or the new PowerShell cmdlet modules.

To create an update schedule via the Azure portal, see Schedule an update deployment.

The `Az.Automation` module now supports configuring Update Management by using Azure PowerShell. The `New-AzAutomationUpdateManagementAzureQuery` cmdlet allows you to use tags, location, and saved searches to configure update schedules for a flexible group of machines.

Example script

The example script in this section illustrates the use of tagging and querying to create dynamic groups of machines that you can apply update schedules to. It performs the following actions. You can refer to the implementations of the specific actions when you create your own scripts.

- Creates an Azure Automation update schedule that runs every Saturday at 8:00 AM.
- Creates a query for any machines that match these criteria:
  - Deployed in the `westus`, `eastus`, or `eastus2` Azure location.
  - Has an `Owner` tag applied with a value set to `JaneSmith`.
  - Has a `Production` tag applied with a value set to `true`.
- Applies the update schedule to the queried machines and sets a two-hour update window.

Before you run the example script, you'll need to sign in by using the `Connect-AzAccount` cmdlet. When you start the script, provide the following information:

- The target subscription ID
- The target resource group
- Your Log Analytics workspace name
- Your Azure Automation account name

```
<#
    .SYNOPSIS
        This script orchestrates the deployment of the solutions and the agents.
    .Parameter SubscriptionName
    .Parameter WorkspaceName
    .Parameter AutomationAccountName
    .Parameter ResourceGroupName

#>

param (
    [Parameter(Mandatory=$true)]
    [string] $SubscriptionId,

    [Parameter(Mandatory=$true)]
    [string] $ResourceGroupName,

    [Parameter(Mandatory=$true)]
    [string] $WorkspaceName,

    [Parameter(Mandatory=$true)]
    [string] $AutomationAccountName,

    [Parameter(Mandatory=$false)]
    [string] $scheduleName = "SaturdayCriticalSecurity"
)

Import-Module Az.Automation

$startTime = ([DateTime]::Now).AddMinutes(10)
$schedule = New-AzAutomationSchedule -ResourceGroupName $ResourceGroupName `
    -AutomationAccountName $AutomationAccountName `
    -StartTime $startTime `
    -Name $scheduleName `
    -Description "Saturday patches" `
    -DaysOfWeek Saturday `
    -WeekInterval 1 `
    -ForUpdateConfiguration

# Using AzAutomationUpdateManagementAzureQuery to create dynamic groups.

$queryScope = @("/subscriptions/$SubscriptionID/resourceGroups/")

$query1Location =@("westus", "eastus", "eastus2")
$query1FilterOperator = "Any"
$ownerTag = @{ "Owner"= @("JaneSmith") }
$ownerTag.Add("Production", "true")

$DGQuery = New-AzAutomationUpdateManagementAzureQuery -ResourceGroupName $ResourceGroupName `
    -AutomationAccountName $AutomationAccountName `
    -Scope $queryScope `
    -Tag $ownerTag

$AzureQueries = @($DGQuery)

$UpdateConfig = New-AzAutomationSoftwareUpdateConfiguration -ResourceGroupName $ResourceGroupName `
    -AutomationAccountName $AutomationAccountName `
    -Schedule $schedule `
    -Windows `
    -Duration (New-TimeSpan -Hours 2) `
    -AzureQuery $AzureQueries `
    -IncludedUpdateClassification Security,Critical
```

Next steps

See examples of how to implement common policies in Azure that can help manage your servers.

Common policies in Azure

Common Azure Policy examples

5/18/2021 • 2 minutes to read • Edit Online

Azure Policy can help you apply governance to your cloud resources. This service can help you create guardrails that ensure company-wide compliance to governance policy requirements. To create policies, use either the Azure portal or PowerShell cmdlets. This article provides PowerShell cmdlet examples.

> **NOTE**
> With Azure Policy, enforcement policies (`DeployIfNotExists`) aren't automatically deployed to existing VMs. Remediation is required to keep VMs in compliance. For more information, see Remediate noncompliant resources with Azure Policy.

Common policy examples

The following sections describe some commonly used policies.

Restrict resource regions

Regulatory and policy compliance often depends on control of the physical location where resources are deployed. You can use a built-in policy to allow users to create resources only in certain allowed Azure regions.

To find this policy in the portal, search for "location" on the policy definition page. Or run this cmdlet to find the policy:

```
Get-AzPolicyDefinition | Where-Object { ($_.Properties.policyType -eq 'BuiltIn') `
  -and ($_.Properties.displayName -like '*location*') }
```

The following script shows how to assign the policy. Change the `$SubscriptionID` value to point to the subscription that you want to assign the policy to. Before you run the script, use the `Connect-AzAccount` cmdlet to sign in.

```


# Specify the value for $SubscriptionID.

$SubscriptionID = <subscription ID>
$scope = "/subscriptions/$SubscriptionID"

# Replace the -Name GUID with the policy GUID you want to assign.

$AllowedLocationPolicy = Get-AzPolicyDefinition -Name "e56962a6-4747-49cd-b67b-bf8b01975c4c"

# Replace the locations with the ones you want to specify.

$policyParam = '{ "listOfAllowedLocations":{"value":["eastus","westus"]}}'
New-AzPolicyAssignment -Name "Allowed Location" -DisplayName "Allowed locations for resource creation" -
Scope $scope -PolicyDefinition $AllowedLocationPolicy -Location eastus -PolicyParameter $policyParam
```

You can also use this script to apply the other policies that are discussed in this article. Just replace the GUID in the line that sets `$AllowedLocationPolicy` with the GUID of the policy that you want to apply.

Block certain resource types

Another common built-in policy that's used to control costs can also be used to block certain resource types.

To find this policy in the portal, search for "allowed resource types" on the policy definition page. Or run this cmdlet to find the policy:

```
Get-AzPolicyDefinition | Where-Object { ($_.Properties.policyType -eq "BuiltIn") -and
($_.Properties.displayName -like "*allowed resource types") }
```

After you identify the policy that you want to use, you can modify the PowerShell sample in the Restrict resource regions section to assign the policy.

Restrict VM size

Azure offers a wide range of VM sizes to support various workloads. To control your budget, you could create a policy that allows only a subset of VM sizes in your subscriptions.

Deploy antimalware

You can use this policy to deploy the Microsoft Antimalware Extension with a default configuration to VMs that aren't protected by antimalware.

The policy GUID is `2835b622-407b-4114-9198-6f7064cbe0dc`.

The following script shows how to assign the policy. To use the script, change the `$SubscriptionID` value to point to the subscription that you want to assign the policy to. Before you run the script, use the `Connect-AzAccount` cmdlet to sign in.

```
# Specify the value for $SubscriptionID.

$subscriptionID = <subscription ID>
$scope = "/subscriptions/$subscriptionID"

$antimalwarePolicy = Get-AzPolicyDefinition -Name "2835b622-407b-4114-9198-6f7064cbe0dc"

# Replace location "eastus" with the value that you want to use.

New-AzPolicyAssignment -Name "Deploy Antimalware" -DisplayName "Deploy default Microsoft IaaSAntimalware
extension for Windows Server" -Scope $scope -PolicyDefinition $antimalwarePolicy -Location eastus -
AssignIdentity
```

Next steps

Learn about other server-management tools and services that are available.

Azure server management tools and services

Azure server management tools and services

8/4/2021 • 4 minutes to read • Edit Online

As is discussed in the overview of this guidance, the suite of Azure server management services covers these areas:

- Migrate
- Secure
- Protect
- Monitor
- Configure
- Govern

The following sections briefly describe these management areas and provide links to detailed content about the main Azure services that support them.

Migrate

Migration services can help you migrate your workloads into Azure. To provide the best guidance, the Azure Migrate service starts by measuring on-premises server performance and assessing suitability for migration. After Azure Migrate completes the assessment, you can use Azure Site Recovery and Azure Database Migration Service to migrate your on-premises machines to Azure.

Secure

Azure Security Center is a comprehensive security management application. By onboarding to Security Center, you can quickly get an assessment of the security and regulatory compliance status of your environment. For instructions on onboarding your servers to Azure Security Center, see Configure Azure management services for a subscription.

Protect

To protect your data, you need to plan for backup, high availability, encryption, authorization, and related operational issues. These topics are covered extensively online, so here we'll focus on building a business continuity and disaster recovery (BCDR) plan. We'll include references to documentation that describes in detail how to implement and deploy this type of plan.

When you build data-protection strategies, first consider breaking down your workload applications into their different tiers. This approach helps because each tier typically requires its own unique protection plan. To learn more about designing applications to be resilient, see Designing resilient applications for Azure.

The most basic data protection is backup. To speed up the recovery process if servers are lost, back up not just data but also server configurations. Backup is an effective mechanism to handle accidental data deletion and ransomware attacks. Azure Backup can help you protect your data on Azure and on-premises servers running Windows or Linux. For details about what Azure Backup can do and for how-to guides, see the Azure Backup service overview.

If a workload requires real-time business continuity for hardware failures or datacenter outage, consider using data replication. Azure Site Recovery provides continuous replication of your VMs, a solution that provides bare-minimum data loss. Site Recovery also supports several replication scenarios, such as replication:

- Of Azure VMs between two Azure regions.
- Between servers on-premises.
- Between on-premises servers and Azure.

For more information, see the complete Azure Site Recovery replication matrix.

For your file-server data, another service to consider is Azure File Sync. This service helps you centralize your organization's file shares in Azure Files, while preserving the flexibility, performance, and compatibility of an on-premises file server. To use this service, follow the instructions for deploying Azure File Sync.

Monitor

Azure Monitor provides a view into various resources, like applications, containers, and virtual machines. It also collects data from several sources:

- Azure Monitor for VMs provides an in-depth view of VM health, performance trends, and dependencies. The service monitors the health of the operating systems of your Azure Virtual Machines, virtual-machine scale sets, and machines in your on-premises environment.
- Log Analytics is a feature of Azure Monitor. Its role is central to the overall Azure management story. It serves as the data store for log analysis and for many other Azure services. It offers a rich query language and an analytics engine that provides insights into the operation of your applications and resources.
- Azure activity log is also a feature of Azure Monitor. It provides insight into subscription-level events that occur in Azure.

Configure

Several services fit into this category. They can help you to:

- Automate operational tasks.
- Manage server configurations.
- Measure update compliance.
- Schedule updates.
- Detect changes to your servers.

These services are essential to supporting ongoing operations:

- Update Management automates the deployment of patches across your environment, including deployment to operating-system instances running outside of Azure. It supports both Windows and Linux operating systems, and tracks key OS vulnerabilities and nonconformance caused by missing patches.
- Change Tracking and Inventory provides insight into the software that's running in your environment, and highlights any changes that have occurred.
- Azure Automation lets you run Python and PowerShell scripts or runbooks to automate tasks across your environment. When you use Azure Automation with the Hybrid Runbook Worker, you can extend your runbooks to your on-premises resources as well.
- Azure Automation State Configuration enables you to push PowerShell Desired State Configuration (DSC) configurations directly from Azure. DSC also lets you monitor and preserve configurations for guest operating systems and workloads.

Govern

Adopting and moving to the cloud creates new management challenges. It requires a different mindset as you shift from an operational management burden to monitoring and governance. The Cloud Adoption Framework starts with governance. The framework explains how to migrate to the cloud, what the journey will look like, and who should be involved.

The governance design for standard organizations often differs from governance design for complex enterprises. To learn more about governance best practices for a standard organization, see the standard enterprise governance guide. To learn more about governance best practices for a complex enterprise, see the governance guide for complex enterprises.

Billing information

To learn about pricing for Azure management services, go to these pages:

- Azure Site Recovery
- Azure Backup
- Azure Monitor
- Azure Security Center
- Azure Automation, including:
  - Desired State Configuration
  - Update Management solution
  - Change Tracking and Inventory solution
- Azure Policy
- Azure File Sync service

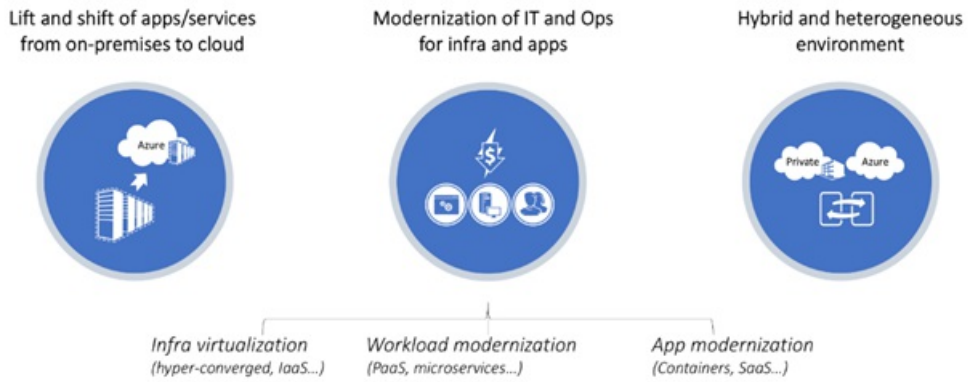
> **NOTE**
> The Azure Update Management solution is free, but there's a small cost related to data ingestion. As a rule of thumb, the first 5 gigabytes (GB) per month of data ingestion are free. We generally observe that each machine uses about 25 MB per month. So, about 200 machines per month are covered for free. For more servers, multiply the number of additional servers by 25 MB per month. Then, multiply the result by the storage price for the additional storage that you need. For information about costs, see Azure Storage pricing. Each additional server typically has a nominal impact on cost.

Cloud monitoring guide: Introduction

5/4/2021 • 3 minutes to read • Edit Online

The cloud fundamentally changes how enterprises procure and use technology resources. In the past, enterprises assumed ownership of and responsibility for all levels of technology, from infrastructure to software. Now, the cloud offers the potential for enterprises to provision and consume resources as needed.

Although the cloud offers nearly unlimited flexibility in terms of design choices, enterprises seek proven and consistent methodology for the adoption of cloud technologies. Each enterprise has different goals and timelines for cloud adoption, making a one-size-fits-all approach to adoption nearly impossible.



This digital transformation also enables an opportunity to modernize your infrastructure, workloads, and applications. Depending on business strategy and objectives, adopting a hybrid cloud model is likely part of the migration journey from on-premises to operating fully in the cloud. During this journey, IT teams are challenged to adopt and realize rapid value from the cloud. IT must also understand how to effectively monitor the application or service that's migrating to Azure, and continue to deliver effective IT operations and DevOps.

Stakeholders want to use cloud-based, software as a service (SaaS) monitoring and management tools. They need to understand what services and solutions deliver to achieve end-to-end visibility, reduce costs, and focus less on infrastructure and maintenance of traditional software-based IT operations tools.

However, IT often prefers to use the tools they've already made a significant investment in. This approach supports their service operations processes to monitor both cloud models, with the eventual goal of transitioning to a SaaS-based offering. IT prefers this approach not only because it takes time, planning, resources, and funding to switch. It's also because of confusion about which products or Azure services are appropriate or applicable to achieve the transition.

The goal of this guide is to provide a detailed reference to help enterprise IT managers, business decision makers, application architects, and application developers understand:

- Azure monitoring platforms, with an overview and comparison of their capabilities.
- The best-fit solution for monitoring hybrid, private, and Azure native workloads.
- The recommended end-to-end monitoring approach for both infrastructure and applications. This approach includes deployable solutions for migrating these common workloads to Azure.

This guide isn't a how-to article for using or configuring individual Azure services and solutions, but it does reference those sources when they're applicable or available. After you've read it, you'll understand how to successfully operate a workload by following best practices and patterns.

If you're unfamiliar with Azure Monitor and System Center Operations Manager, and you want to get a better understanding of what makes them unique and how they compare to each other, review the overview of our

monitoring platforms.

Audience

This guide is useful primarily for enterprise administrators, IT operations, IT security and compliance, application architects, workload development owners, and workload operations owners.

How this guide is structured

This article is part of a series. The following articles are meant to be read together, in order:

- Introduction (this article)
- Cloud monitoring strategy
- Monitoring platform strategy for cloud deployment models
- Observability
- Collect the right data
- Alerting

Products and services

A few software and services are available to help you monitor and manage a variety of resources that are hosted in Azure, your corporate network, or other cloud providers. They are:

- System Center Operations Manager
- Azure Monitor (includes Log Analytics and Application Insights)
- Azure Policy and Azure Blueprints
- Azure Arc
- Azure Automation
- Azure Logic Apps
- Azure Event Hubs

This first version of the guide covers our current monitoring platforms: Azure Monitor and System Center Operations Manager. It also outlines our recommended strategy for monitoring each of the cloud deployment models. Also included is the first set of monitoring recommendations, starting with data collection, observability, and alerting.

Next steps

Monitoring strategy for cloud deployment models

Cloud monitoring guide: Monitoring strategy for cloud deployment models

7/21/2021 • 17 minutes to read • Edit Online

This article includes our recommended monitoring strategy for each of the cloud deployment models, based on the following criteria:

- You must maintain your commitment to Operations Manager or another enterprise monitoring platform, because it's integrated with your IT operations processes, knowledge, and expertise, or certain functionality isn't available yet in Azure Monitor.
- You must monitor workloads both on-premises and in the public cloud, or just in the cloud.
- Your cloud migration strategy includes modernizing IT operations and moving to our cloud monitoring services and solutions.
- You might have critical systems that are air-gapped or physically isolated, or are hosted in a private cloud or on physical hardware, and these systems need to be monitored.

Our strategy includes support for monitoring infrastructure (compute, storage, and server workloads), application (end-user, exceptions, and client), and network resources. It delivers a complete, service-oriented monitoring perspective.

Azure cloud monitoring

Azure Monitor is the Azure native platform service that provides a single source for monitoring Azure resources. It's designed for cloud solutions that:

- Are built on Azure.
- Support a business capability that's based on virtual machine (VM) workloads or complex architectures that use microservices and other platform resources.

It monitors all layers of the stack, starting with tenant services, such as Azure Active Directory Domain Services, and subscription-level events and Azure Service Health.

It also monitors infrastructure resources, such as VMs, storage, and network resources. At the top layer, it monitors your application.

By monitoring each of these dependencies, and collecting the right signals that each can emit, you get the observability of applications and the key infrastructure you need.

Our recommended approach to monitoring each layer of the stack is summarized in the following table:

LAYER	RESOURCE	SCOPE	METHOD
Application	A web-based application that runs on .NET, .NET Core, Java, JavaScript, and Node.js platform on an Azure VM, Azure App Service, Azure Service Fabric, Azure Functions, and Azure Cloud Services.	Monitor a live web application to automatically detect performance anomalies, identify code exceptions and issues, and collect user behavior analytics.	Application Insights (a feature of Azure Monitor).

LAYER	RESOURCE	SCOPE	METHOD
Azure resources - platform as a service (PaaS)	Azure database services (for example, SQL or MySQL).	Azure SQL Database performance metrics.	Enable diagnostics logging to stream SQL data to Azure Monitor Logs.
Azure resources - infrastructure as a service (IaaS)	1. Azure Storage 2. Azure load balancing services 3. Network security groups 4. Azure Virtual Machines 5. Azure Kubernetes Service / Azure Container Instances	1. Capacity, availability, and performance. 2. Performance and diagnostics logs (activity, access, performance, and firewall). 3. Monitor events when rules are applied, and the rule counter for how many times a rule is applied to deny or allow. 4. Monitor capacity, availability, and performance in a guest VM operating system (OS). Map application dependencies hosted on each VM, including the visibility of active network connections between servers, inbound and outbound connection latency, and ports across any TCP-connected architecture. 5. Monitor capacity, availability, and performance of workloads running on containers and container instances.	• For items 1 through 5 in the first column, platform metrics and the activity log are automatically collected and available in Azure Monitor for analysis and alerting. • Configure diagnostic settings to forward resource logs to Azure Monitor Logs. • Enable Azure Monitor for VMs. • Enable Azure Monitor for containers.
Network	Communication between your virtual machine and one or more endpoints (another VM, a fully qualified domain name, a uniform resource identifier, or an IPv4 address).	Monitor reachability, latency, and network topology changes that occur between the VM and the endpoint.	Azure Network Watcher.

LAYER	RESOURCE	SCOPE	METHOD
Azure subscription	Azure Service Health and basic resource health from the perspective of the Azure service.	• Administrative actions performed on a service or resource. • Service health of an Azure service is in a degraded or unavailable state. • Health issues detected with an Azure resource from the Azure service perspective. • Operations performed with Azure autoscale indicating a failure or exception. • Operations performed with Azure Policy indicating that an allowed or denied action occurred. • Record of alerts generated by Azure Security Center.	Delivered in the activity log for monitoring and alerting by using Azure Monitor.
Azure tenant	Azure Active Directory	Azure AD audit logs and sign-in logs.	Enable diagnostics logging, and configure streaming to Azure Monitor Logs.

Hybrid cloud monitoring

For many organizations, transition to the cloud must be approached gradually, where the hybrid cloud model is the most common first step in the journey. You carefully select the appropriate subset of applications and infrastructure to begin your migration, while you avoid disruption to your business. However, because we offer two monitoring platforms that support this cloud model, IT decision makers might be uncertain as to which platform is the best choice to support their business and IT operational goals.

In this section, we address the uncertainty by reviewing several factors and offering an understanding of which platform to consider.

Keep in mind the following key technical aspects:

- You need to collect data from Azure resources that support the workload, then forward the data to your existing on-premises or managed service provider tools.
- You need to maintain your current investment in System Center Operations Manager, and configure it to monitor IaaS and PaaS resources that are running in Azure. Optionally, because you're monitoring two environments with different characteristics, based on your requirements, you need to determine how integrating with Azure Monitor supports your strategy.
- As part of your modernization strategy to standardize on a single tool to reduce cost and complexity, you need to commit to Azure Monitor for monitoring the resources in Azure and on your corporate network.

The following table summarizes the requirements that Azure Monitor and System Center Operations Manager support with monitoring the hybrid cloud model based on a common set of criteria.

REQUIREMENT	AZURE MONITOR	OPERATIONS MANAGER
Infrastructure requirements	No	Yes Requires, at a minimum, a management server and a SQL Server instance to host the operational database and the reporting data warehouse database. The complexity increases when high availability and disaster recovery are required, and there are machines in multiple sites, untrusted systems, and other complex design considerations.
Limited connectivity - no internet or isolated network	No	Yes
Limited connectivity - controlled internet access	Yes	Yes
Limited connectivity - frequently disconnected	Yes	Yes
Configurable health monitoring	No	Yes
Web app availability test (isolated network)	Yes, limited Azure Monitor has limited support in this area and requires custom firewall exceptions.	Yes
Web app availability test (globally distributed)	No	Yes
Monitor VM workloads	Yes, limited Can collect IIS and SQL Server error logs, Windows events, and performance counters. Requires creating custom queries, alerts, and visualizations.	Yes Supports monitoring most of the server workloads with available management packs. Requires either the Log Analytics Windows agent or Operations Manager agent on the VM, reporting back to the management group on the corporate network.
Monitor Azure IaaS	Yes	Yes Supports monitoring most of the infrastructure from the corporate network. Tracks availability state, metrics, and alerts for Azure VMs, SQL, and storage via the Azure management pack.
Monitor Azure PaaS	Yes	Yes, limited Based on what's supported in the Azure management pack.

REQUIREMENT	AZURE MONITOR	OPERATIONS MANAGER
Azure service monitoring	Yes	Yes Although there's no native monitoring of Azure Service Health provided today through a management pack, you can create custom workflows to query Service Health alerts. Use the Azure REST API to get alerts through your existing notifications.
Modern web application monitoring	Yes	No
Legacy web application monitoring	Yes, limited, varies by SDK Supports monitoring older versions of .NET and Java web applications.	Yes, limited
Monitor Azure Kubernetes Service containers	Yes	No
Monitor Docker or Windows containers	Yes	No
Network performance monitoring	Yes	Yes, limited Supports availability checks, and collects basic statistics from network devices by using the Simple Network Management Protocol (SNMP) from the corporate network.
Interactive data analysis	Yes	No Relies on SQL Server Reporting Services canned or custom reports, third-party visualization solutions, or a custom Power BI implementation. There are scale and performance limitations with the Operations Manager data warehouse. Integrate with Azure Monitor Logs as an alternative for data aggregation requirements. You achieve integration by configuring the Log Analytics connector.
End-to-end diagnostics, root-cause analysis, and timely troubleshooting	Yes	Yes, limited Supports end-to-end diagnostics and troubleshooting only for on-premises infrastructure and applications. Uses other System Center components or partner solutions.

REQUIREMENT	AZURE MONITOR	OPERATIONS MANAGER
Interactive visualizations (dashboards)	Yes	Yes, limited Delivers essential dashboards with its HTML5 web console or an advanced experience from partner solutions, such as Squared Up and Savision.
Integration with IT or DevOps tools	Yes	Yes, limited

Collect and stream monitoring data to third-party or on-premises tools

To collect metrics and logs from Azure infrastructure and platform resources, you need to enable Azure Diagnostics logs for those resources. Additionally, with Azure VMs, you can collect metrics and logs from the guest OS by enabling the Azure Diagnostics extension. To forward the diagnostics data that's emitted from your Azure resources to your on-premises tools or managed service provider, configure Event Hubs to stream the data to them.

Monitor with System Center Operations Manager

Although System Center Operations Manager was originally designed as an on-premises solution to monitor across applications, workloads, and infrastructure components that are running in your IT environment, it evolved to include cloud-monitoring capabilities. It integrates with Azure, Microsoft 365, and Amazon Web Services (AWS). It can monitor across these diverse environments with management packs that are designed and updated to support them.

For customers who have made significant investments in Operations Manager to achieve comprehensive monitoring that's tightly integrated with their IT service management processes and tools, or for customers new to Azure, it's understandable to ask the following questions:

- Can Operations Manager continue to deliver value, and does it make business sense?
- Do the features in Operations Manager make it the right fit for our IT organization?
- Does integrating Operations Manager with Azure Monitor provide the cost-effective and comprehensive monitoring solution that we require?

If you've already invested in Operations Manager, you don't need to focus on planning a migration to replace it immediately. With Azure or other cloud providers that exist as an extension of your own on-premises network, Operations Manager can monitor the guest VMs and Azure resources as if they were on your corporate network. This approach requires a reliable network connection between your network and the virtual network in Azure that has sufficient bandwidth.

To monitor the workloads that are running in Azure, you need:

- The System Center Operations Manager management pack for Azure. It collects performance metrics emitted by Azure services such as web and worker roles, Application Insights availability tests (web tests), Azure Service Bus, and so on. The management pack uses the Azure REST API to monitor the availability and performance of these resources. Some Azure service types have no metrics or predefined monitors in the management pack, but you can still monitor them through the relationships defined in the Azure management pack for discovered services.
- The management pack for Azure SQL Database to monitor the availability and performance of Azure SQL databases and Azure SQL Database instances using the Azure REST API and T-SQL queries to SQL Server system views.
- To monitor the guest OS and workloads that are running on the VM, such as SQL Server, IIS, or Apache Tomcat, you need to download and import the management pack that supports the application, service,

and OS.

Knowledge is defined in the management pack, which describes how to monitor the individual dependencies and components. Both Azure management packs require performing a set of configuration steps in Azure and Operations Manager before you can begin monitoring these resources.

At the application tier, Operations Manager offers basic application performance monitoring capabilities for some legacy versions of .NET and Java. If certain applications within your hybrid cloud environment operate in an offline or network-isolated mode, such that they can't communicate with a public cloud service, Operations Manager application performance monitoring (APM) might be a viable option for certain limited scenarios. For applications that are not running on legacy platforms but are hosted both on-premises and in any public cloud that allows communication through a firewall (either direct or via a proxy) to Azure, use Azure Monitor Application Insights. This service offers deep, code-level monitoring, with first-class support for ASP.NET, ASP.NET Core, Java, JavaScript, and Node.js.

For any web application that can be reached externally, you should enable a type of synthetic transaction known as availability monitoring. It's important to know whether your application or a critical HTTP/HTTPS endpoint that your application relies on, is available and responsive. With Application Insights availability monitoring, you can run tests from multiple Azure datacenters and provide insight into the health of your application from a global perspective.

Although Operations Manager is capable of monitoring resources that are hosted in Azure, there are several advantages to including Azure Monitor, because its strengths overcome the limitations in Operations Manager and can establish a strong foundation to support eventual migration from it. Here we review each of those strengths and weaknesses, with our recommendation to include Azure Monitor in your hybrid monitoring strategy.

Disadvantages of using Operations Manager by itself

- Analyzing monitoring data in Operations Manager is commonly performed by using predefined views that are provided by management packs accessed from the console, from SQL Server Reporting Services (SSRS) reports, or from custom views that end users have created. Ad hoc data analysis isn't possible out of the box. Operations Manager reporting is inflexible. The data warehouse that provides long-term retention of the monitoring data doesn't scale or perform well. And expertise in writing T-SQL statements, developing a Power BI solution, or using third-party solutions is required to support the requirements for the various personas in the IT organization.

- Alerting in Operations Manager doesn't support complex expressions or include correlation logic. To help reduce noise, alerts are grouped to show the relationships between them and to identify their causes.

Advantages of using Operations Manager with Azure Monitor

- Azure Monitor is the way to work around the limitations of Operations Manager. It complements the Operations Manager data warehouse database by collecting important performance and log data. Azure Monitor delivers better analytics, performance (when querying large data volume), and retention than the Operations Manager data warehouse.

  With the Azure Monitor Logs query language, you can create much more complex and sophisticated queries. You can run queries across terabytes of data in seconds. You can quickly transform your data into pie charts, time charts, and many other visualizations. To analyze this data, you're no longer constrained by working with Operations Manager reports that are based on SQL Server Reporting Services, custom SQL queries, or other workarounds.

- You can deliver an improved alerting experience by implementing the Azure Monitor Alert Management solution. Alerts that are generated in the Operations Manager management group can be forwarded to the Azure Monitor Log Analytics workspace. You can configure the subscription that's responsible for forwarding alerts from Operations Manager to Azure Monitor Logs to forward only certain alerts. For example, you can forward only alerts that meet your criteria for querying in support of problem

management for trends, and investigation of the root cause of failures or problems, through a single pane of glass. Additionally, you can correlate other log data from Application Insights or other sources, to gain insight that help improve user experience, increase uptime, and reduce time to resolve incidents.

- You can monitor cloud-native infrastructure and applications, from a simple or multitier architecture in Azure using Azure Monitor, and you can use Operations Manager to monitor on-premises infrastructure. This monitoring includes one or more VMs, multiple VMs placed in an availability set or virtual machine scale set, or a containerized application deployed to Azure Kubernetes Service (AKS) that's running on Windows Server or Linux containers.

  If you need comprehensive monitoring of Microsoft or third-party workloads running on your Azure VMs, and you have advanced scenarios that cannot be evaluated based on log or performance data alone, use System Center Operations Manager. Its management packs delivers advanced logic, which includes a service and health model, to determine the operational health of the workload.

- By using the Map feature of Azure Monitor for VMs, you can monitor standard connectivity metrics from network connections between your Azure VMs and on-premises VMs. These metrics include response time, requests per minute, traffic throughput, and links. You can identify failed connections, troubleshoot, perform migration validation, perform security analysis, and verify the overall architecture of the service. Map can automatically discover application components on Windows and Linux systems, and map the communication between services. This automation helps you identify connections and dependencies you were unaware of, plan and validate migration to Azure, and minimize speculation during incident resolution.

- By using Network Performance Monitor, you can monitor the network connectivity between:

  - Your corporate network and Azure.
  - Mission-critical multitier applications and microservices.
  - User locations and web-based applications (HTTP/HTTPS).

    This strategy delivers visibility of the network layer, without the need for SNMP. It can also present, in an interactive topology map, the hop-by-hop topology of routes between the source and destination endpoint. It's a better choice than attempting to accomplish the same result with network monitoring in Operations Manager or with other network monitoring tools currently used in your environment.

Monitor with Azure Monitor

Although a migration to the cloud presents numerous challenges, it also provides opportunities. It enables your organization to migrate from one or more on-premises enterprise monitoring tools to not only potentially reduce capital expenditures and operating costs, but also to benefit from the advantages that a cloud monitoring platform such as Azure Monitor can deliver at cloud scale. Examine your monitoring and alerting requirements, configuration of existing monitoring tools, and workloads transitioning to the cloud. After your plan is finalized, configure Azure Monitor.

- Monitor the hybrid infrastructure and applications, from a simple or multitier architecture where components are hosted between Azure, other cloud providers, and your corporate network. The components might include one or more VMs, multiple VMs placed in an availability set or virtual machine scale set, or a containerized application that's deployed to Azure Kubernetes Service (AKS) running on Windows Server or Linux containers.

- Use Azure Arc to prepare your servers, virtual machines, Kubernetes clusters, and databases across your environment for management as if they are running in Azure. Azure Arc delivers consistent inventory, management, governance, and security with familiar Azure services and management capabilities.

- Enable Azure Monitor for VMs, Azure Monitor for containers, and Application Insights to detect and

diagnose issues between infrastructure and applications. For a more thorough analysis and correlation of data collected from the multiple components or dependencies supporting the application, you need to use Azure Monitor Logs.

- Create intelligent alerts that apply to a core set of applications and service components, help reduce alert noise with dynamic thresholds for complex signals, and use alert aggregation based on machine learning algorithms to help identify the issue quickly.
- Define a library of queries and dashboards to support the requirements of the various personas in the IT organization.
- Define standards and methods for enabling monitoring across the hybrid and cloud resources, a monitoring baseline for each resource, alert thresholds, and so on.
- Configure Azure role-based access control (Azure RBAC) so you grant users and groups only the access required to monitor data from the resources they manage.
- Include automation and self-service to enable each team to create, enable, and tune their monitoring and alerting configurations as needed.

Private cloud monitoring

You can achieve holistic monitoring of Azure Stack with System Center Operations Manager. Specifically, you can monitor the workloads that are running in the tenant, the resource level, on the virtual machines, and the infrastructure hosting Azure Stack (physical servers and network switches).

You can also achieve holistic monitoring with a combination of infrastructure monitoring capabilities that are included in Azure Stack. These capabilities help you view health and alerts for an Azure Stack region and the Azure Monitor service in Azure Stack, which provides base-level infrastructure metrics and logs for most services.

If you've already invested in Operations Manager, use the Azure Stack management pack to monitor the availability and health state of Azure Stack deployments, including regions, resource providers, updates, update runs, scale units, unit nodes, infrastructure roles, and their instances (logical entities comprised of the hardware resources). This management pack uses the health and update resource provider REST APIs to communicate with Azure Stack. To monitor physical servers and storage devices, use the OEM vendors' management pack (for example, provided by Lenovo, HPE, or Dell). Operations Manager can natively monitor the network switches to collect basic statistics by using SNMP. Monitoring the tenant workloads is possible with the Azure management pack by following two basic steps. Configure the subscription that you want to monitor, and then add the monitors for that subscription.

Next steps

Collect the right data

Cloud monitoring guide: Observability

8/9/2021 • 18 minutes to read • Edit Online

This article is intended to help organizations implement a consistent monitoring strategy quicker by ensuring *observability* is established in the Azure landing zone (that is, in every minimum viable product) for each monitoring solution.

As a planner, you formulate a monitoring plan for a service and include some monitoring objectives. One objective you want is for the monitoring consumer(s) to reach a level of comfort or confidence with the solution you are planning for them, as soon as possible. Then you can move on to other objectives, like building reports and custom dashboards.

We call this *adoption*.

The goal of this article is to take the monitoring strategy recommendations, and create a monitoring plan to support the goal of modernizing and evolving your organizations IT operations strategy. The second goal is to drive operational maturity by being observant and constantly iterating to improve how you monitor those services.

> **NOTE**
> In this article, a *monitoring solution* is the unit of production doing the monitoring of a service in the cloud, and a *monitoring target* is the service or thing that is being monitored. A solution encompasses all aspects of monitoring: the tool, the monitoring data, alerts, type of response, recovery actions, type of visualization, role-based access, and so on.

Why observability is important

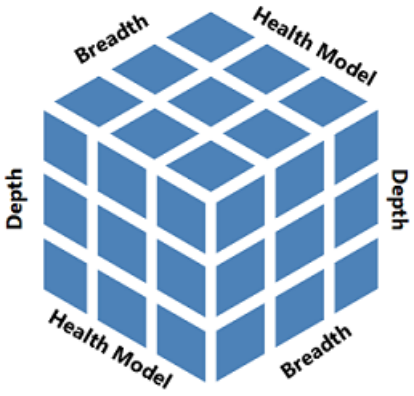
Simply, its observability *first* that drives the monitoring consumer to understand what is considered or perceived as *normal* operation of a service. In other words, you seek *total visibility*, a key monitoring principle, as soon as possible. Once initial observability is achieved, then you build upon that initial level of visibility to develop actionable alerts, create useful dashboards, and evaluate AIOps solutions. This allows you time to get comfortable with the underlying metric and log monitoring data.

> **NOTE**
> This is the opposite of the approach used in the past, when teams worked to define all of the monitoring requirements first on paper, before build, test, and deploy.

Whether the plan (that is, objective) targets an Azure resource type such as Key Vault, an entire resource group, or even Microsoft 365 Exchange Online, the first step is to *establish observability*.

This approach also simplifies your plans. In all cases, total visibility means achieving and sustaining, sufficient visibility across three dimensions or aspects:

- Monitoring-in-depth
- Monitoring in breadth
- Monitoring across the health model



Being observant isn't just an IT focus; remember the goal is to ensure end users can consume and business expectations are met. You can't monitor what you don't understand or know, and as a result you won't able to deliver the level of service availability promised to the business. Before the advent of cloud computing, Microsoft emphasized failure mode analysis during application design and development. Failure mode analysis helped developers consider how and when logic or other critical errors could occur in their code. And when it does, expose the condition in a meaningful way to allow the monitoring tool not only detect it and act on it, but also provide developers, operators, or system engineers with useful information to help better understand the applications behavior and make data-driven decisions. Today, the Cloud Adoption Framework recommends you follow that process as part of the architecture and design phases to build recovery of the Azure services in your design.

Bottom line, it's the observability you want, early, and in your minimum viable product.

Monitoring systems and observability

Infrastructure and application monitoring is complicated, it always has been and remains so even with the introduction of cloud computing. Business transformation applies technology to achieve current, and help shape, its future strategy. The cloud has further influenced the complicated nature of monitoring.

This is demonstrated in the following ways:

- Business (digital) transformation efforts shift toward hyper-exploitation of cloud technology.
- Monitoring becomes embedded into Azure resources and resource groups versus separate tooling you manage on-premises.
- Cloud-native monitoring architectures are SIEM-like, such as Azure Monitor. So it's expansive, log-driven, and orders of magnitude more flexible.

For architects, diagnostics form the core of their exploitation of more cost effective cloud-native monitoring structures that let IT manage services holistically across the different cloud models.

Architects must, like operators, understand what diagnostic information an IT infrastructure component or application emits. Combining multivariate, dynamical, time-series, eventful, stateful, and telemetric log streams into valuable intelligence depends on the following:

- The knowledge and experience of the developer or system engineer, who have a deep understanding of the monitoring target.
- Actual support and troubleshooting experience using the data, to find trouble or locate the causes of trouble.
- Review of past incidents to find non-technology reasons, that later can be automated (auto remediation).
- Guidance in the form of documentation, software, training, or consulting by the software or hardware vendor.

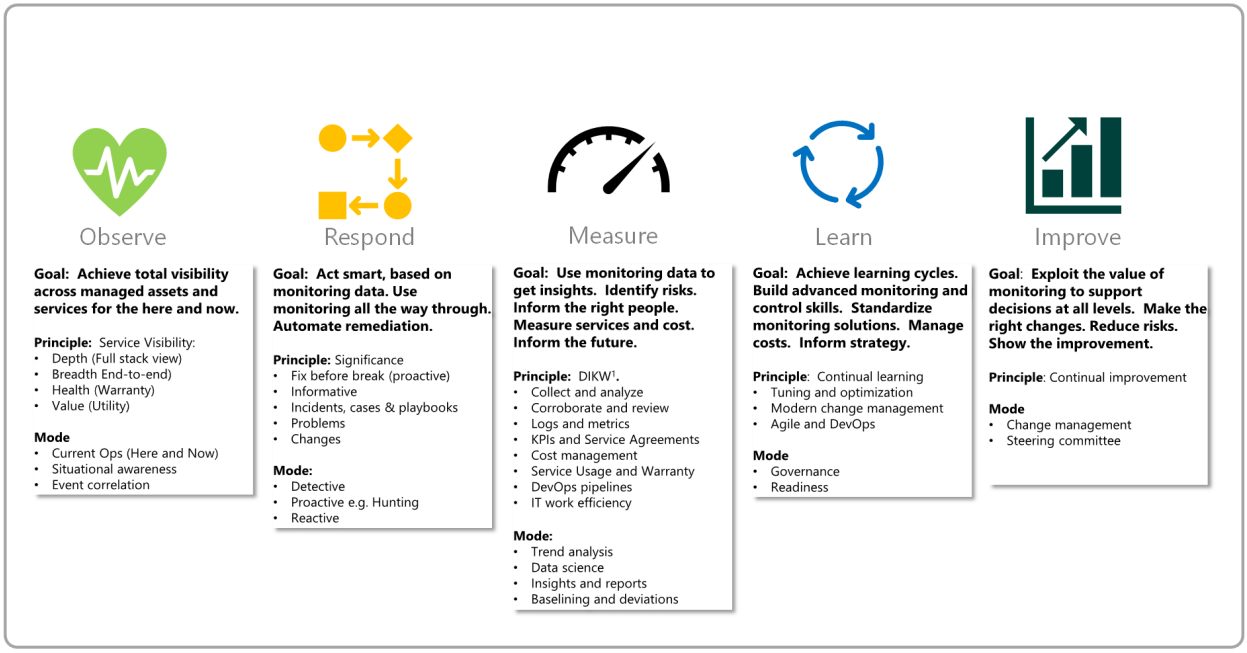
If you are familiar with System Center Operations Manager, Microsoft and its partners provide *management packs.* Management packs are technology-specific; for example, you import a SQL management pack Operations Manager automatically discovers and targets servers hosting SQL Server and begins monitoring them. Here, observability is more or less predefined by the product engineers at Microsoft and dozens of industries. With Operations Manager, you don't need to worry about north-south and east-west dependencies, so that observing the health of SQL is part of the larger IT service with networking, virtualization, and applications included. Owing to a common schema based on the familiar four-part *health model*, Operations Manager is designed for on-premises infrastructure. Infrastructure service architectures tend to be **fixed** in components and architectural patterns, relative to cloud services.

In the cloud, you have enormous flexibility in the types of services you can choose. Monitoring includes how they change over time, and they can be dynamic, global, and resilient. As a cloud architect, you are not constrained by fixed, on-premises thinking. Operations Manager can participate, but again, its strength is traditional on-premises infrastructure and applications. In contrast, Azure Monitor's architecture is much more flexible in supporting all three cloud models. To reach your observability goals in Azure Monitor, you have more freedom to decide on resources, where to place them geographically, and how to visualize the components working together.

With Azure Monitor, you can take advantage of existing workbooks that are included in Insights, which provide similar functionality that a management pack in Operations Manager does. Otherwise, you need to review the Azure documentation for each of the Azure services in order to understand how you can monitor and detect for known failures or symptoms that indicate potential failure.

Monitoring disciplines

The following figure shows the Observe discipline on the left, and later in this section we discuss how Microsoft Azure plays a central role here.



Observability and response

Observability is a *qualitative indicator that a monitoring solution helps the monitoring consumer achieve the satisfactory level of control of a defined service*. Where monitoring provides service consumers with a suitable range of monitoring *capabilities* and *perspectives*.

Response is the organization's ability to quickly ascertain *significance* of observed monitoring data (detected events, telemetric patterns, correlated information, an so on) such that for negative events it can rapidly restore

or remediate. For positive events or informational events, it can help to maintain service agreements, improve reliability, and reduce support costs. Events dynamically occur in the system or service, where monitoring solutions provide alerting, control loop automation, and raise incidents, problems, or changes in an external IT service management (ITSM) system. Realize that many events cannot or should not be automatically remediated.

Utility: In this sense, observability (and response) is about the operational use or utility of the monitoring solution. Azure Monitor provides Microsoft's perspective of our service resources, and gives you similar capabilities as an on-premises monitoring system. A monitoring solution needs visualizations, such as dashboards and workbooks, with role-based access.

Responsibility: Both the service consumer and service provider share in the need to learn and improve based on hard data. As such, you need to understand the cloud provider's responsibility vs. the customer/consumer's responsibility. For each Azure resource, you get perspectives based on logs or metrics, this data can be represented in resource-specific dashboards or custom visualizations based on your requirements, and shared with the necessary personas in the organization.

> **NOTE**
> Observability is a property of a system, stemming from system control theory. It's a measure of how well the internal states of a system can be inferred from its system *external* outputs, along with controllability, another property of a system. In addition, a *dynamical system* also that over time, a state observer measures or estimates the health state of the system. But in modern service management terms, refer to the Microsoft Operations Framework (MOF) **Service Monitoring and Control**," and the Information Technology Infrastructure Library (ITIL) v3 **Monitor and control loop**.

Before we go into details on observability, we need to highlight several monitoring-related terms we'll be using:

- **Asset:** Digital assets such as content in file shares, hardware, and software assets that are also called *targets*.
- **Focus:** Your scope in pursuit of objectives: narrow, broad, a single component, component class, component grouping, service.
- **Aspect:** Perspectives from stakeholders, such as the users, the business, and the service owner.
- **Coverage:** An indicator of visibility and a goal; to ensure all relevant assets are included in the monitoring scope.
- **Manageability:** The extent that a digital asset can be controlled, self-heals, related to change risk and action groups that diagnose or auto remediate.
- **Data source:** The primary location monitoring data comes from, such as an Azure Storage account, Azure Active Directory, or custom sources.
- **Frequency:** Continuous vs. occasional monitoring.

The art of being observant

Observability relies on what is being monitored and how. In Azure, there are multiple sources, and each delivers a different perspective of how something is behaving. Not to mention, Azure includes multiple tools to help analyze the different aspects of this data. Observing the health and performance of Azure services and non-Azure resources is the primary way you'll use Azure Monitor and its features. In Azure, Microsoft has an extensive catalog of services, and virtual machines are not the primary focus. In Azure, we provide the perspective of the *service provider* through different platform logs:

- Service health reported by Azure about service incidents and planned maintenance.

- The Azure activity log reports subscription-level events across all of the resources deployed in the subscription.
- Azure Resource Health reports on the current and past health of your resources.
- Azure Advisor to receive recommended solutions based on best practices to optimize your Azure deployments.

All other perspectives based on metrics and logs, are delivered through the various features of Azure Monitor. Or, depending on the Azure resource, you can view its platform metrics directly from that resource in the portal.

Observability is a term meant to set the expectation that Azure Monitor gives you total visibility of the health, performance, and other aspects of Azure services in depth and breadth. But, there are other activities or focus than service and component monitoring. Being observant shouldn't be considered something that only certain personas perform on an as-needed basis, as required by their role or function, or in support of a process. It's similar to astronomy. Observation isn't once and done, rather you'll want to monitor objects in space occasionally, over set time intervals or continuously. Some objects, such as black holes are difficult to observe. The kind of monitoring needed depends on the consumer. For example, amateur astronomers observe occasionally, while astrophysicists value the long-term measurement and continuous monitoring of radio emissions.

With proper planning, you can create a medium to complex service in Azure in a few hours, versus weeks or longer on-premises. If you did, you'd arrange a combination of more than a dozen Azure resources (service components) across regions. You'll govern and manage these resources using Azure Resource Manager and from using resource group monitoring in Azure, you'll get a picture of the service. Once in production, the service is dynamical and flexible. Consumption starts to ramp up and scale out. Capacity responds to demand. Events will occur, and service consumers can observe monitoring data, look at trends, and establish their significance.

The service can change in different, unpredictable ways over time. Hence we refer to this behavior as dynamical. Managers of cloud services observing the service over time needs to take into account that:

- Resources move across locations or geography
- Resources are added, deleted, or modified
- Consumption varies

As a monitoring service provider, your job is to facilitate monitoring solutions that provide value listed in the following order of priority: 1) to the service, 2) to stakeholders, and 3) to the primary consumers.

Your value stream needs to consider service observability much earlier and in the following ways:

- It monitors in sufficient depth and breadth.
- It provides coverage across the health model (availability, capacity/performance, security, conformance).
- It monitors utility also, what the service does, including the end-user experience.

In the monitoring strategy, we recommend establishing a monitoring plan that starts with what is minimally viable or most important to monitor and observe. This can be based on recommendations from Microsoft, other reputable sources, and your internal developers or system engineers depending on the resource or system. Establish initial visibility of the *consumption*, *performance*, *security*, and *availability* of the resources in your Azure resource groups and your Azure Active Directory tenant. Through observation, you learn how to interpret the data, and learn what's significant in order to fine-tune and optimize how the service is monitored. Value is achieved when incremental, and there is a co-creation of value here in that the consumers are working with the monitoring team (or in some cases the service provider) on co-creation of value.

Observability evolves gradually, starting with a minimally viable monitoring plan, and the effort to integrate tools and processes is underway. As you get comfortable with the data (that is, metrics, logs, and transactions), you are able to understand the behavior and signs of symptoms or issues from those resources or applications.

By getting familiar with the data, you build trust in working with Azure Monitor and the data. Through observability, you gain confidence and you are able to realize the cause and find answers that can help:

- Reduce monitoring blind spots by ensuring coverage of all needed monitoring components.
- Improve monitoring of resources and services to help identify the issue in the future. Focus on issues or symptoms that are predictable and reliable.
- Evaluate whether this wasn't anticipated, what can be done to improve the infrastructure, application, and so on.
- Rule out the infrastructure or application as the source, and determine whether a specific browser, browser version, or client operating system is the potential issue.
- Identify, and document or implement, workarounds that can be used to minimize any performance or availability issues with the application, infrastructure, and so on.
- Adapt to the dynamical and complex nature of an application. Continue learning its normal behavior patterns in order to narrow in on the unusual ones.
- Tune the type of data collected, aggregated, and alerted on to minimize storage cost for data you never analyze, alert on, or visualize.

At first, you no longer need to worry about the infrastructure or application as was the case in an on-premises world; you are now free to monitor or obtain monitoring data to support the needs of those responsible for managing and operating the workload.

Like any application or infrastructure service, a diverse list of stakeholders and roles need access to the monitoring features, data and reports. Start with the core system engineers, operations, or service providers responsible for supporting the workload before extending access to other stakeholders.

Fixed versus dynamical approach

On-premises monitoring of services in the datacenter is traditionally undertaken with a product such as System Center Operations Manager. Operations Manager's approach is firmly rooted in infrastructure and servers, with agents and operating systems. Application monitoring, whether containerized or not, is up-stack and dependent on infrastructure monitoring of servers down-stack. Vertically, you see the solution coverage from bottom-up, starting with networking and at the top, user-experience monitoring.

The long history of IaaS monitoring reveals that *observability can be predefined*, or if you wish, *pre-engineered*. An IaaS-based service is one where the product engineers can author Operations Manager management packs, in essence customizing the monitoring solution for most of the supported use cases.

Thus in the need to control infrastructure services by monitoring solutions, customers seek more of a fixed approach in most cases.

A fixed approach cannot occur in the cloud given the near infinite arrangement and combinations of resources both in space and in time. Each service can be unique. Service observability in Azure must be built depending on the flexible nature of the service (that is, dynamical, global, resilient, user-centric, and so on) or according to some set of architectural templates, which may not yet exist.

Observability is first and paramount for holistically seeing how all components are working together *before* establishing event significance (such as alerts, digests, or workbooks).

Again, cloud service monitoring is much more flexible and dynamical with faster speed of change.

Monitoring plan

You create a monitoring plan to describe the goals and objectives, requirements, and other important details. Then work to solicit an agreement between all of the relevant stakeholders in the organization. A monitoring plan should explain how to develop and operate one or more monitoring solutions. Begin develop your

monitoring plans early during the strategy and planning phases of the project.

While creating the plan, it's important to keep in mind the five disciplines of modern monitoring: observe, measure, respond, and learn, and improve.

The following provides a recommended outline and are considered to be the primary considerations for an individual plan for services or when standardizing cloud service features such as Azure resource types or Microsoft 365 services. The essence of the plan is to define the line of visibility between service provider (who will field solutions) and consumers (who will operate or derive value).

The schedule of when (and how) is also important to include for the work management approach (for example, DevOps vs. waterfall).

Business perspective

A comprehensive monitoring plan needs to take into consideration what the business needs with and from monitoring, and this must include a user-centric focus. While defining the plan, it's important to document and share the business requirements, and the following suggests the scope of this part of the plan.

- Stakeholders and consumers
- Business value streams and processes
- End-user perspective and utility
- Measurement and reporting requirements
- Identified risks and compliance control frameworks
- Access and control requirements
- Risk to the business

Service perspective

A comprehensive monitoring plan needs to take into consideration what the service owners need with and from monitoring. While defining the plan, it's important to document and share their requirements, and the following suggests the scope of this part of the plan.

- Stakeholders and consumers
- Roles and accountability's
- Definition of the service
- Access and control requirements
- Architectural considerations?
- Supplier and partner underpinning contracts
- Service agreements (SLAs, OLAs)
- Identify service warranty coverage
- Measurement and reporting requirements
- Risks

Technology perspective

This section of the plan represents the monitoring solution using information from the business and service perspective.

- User stories and scenarios
- Technical targets (for example, networking)
- Component dependency mapping
- Types (for example, cloud native, hybrid, on-premises)
- Observational
- Responsive
- Measurement

- Tuning and optimization

Key considerations

Summarize the plan to ensure it communicates and informs all relevant consumers, stakeholders, and management levels.

- **Production stages:** The monitoring solution should be ready for value when the service goes live, so planning can include lab or preproduction configuration (that is, in another subscription dedicated to supporting this) for experimenting and testing your assumptions. In the cloud, a production tenant is usually safe based on the overall governance.
- **Strategy:** Plans can also map back to monitoring and IT strategy so that the monitoring objectives trace back to the mission or business. It depends on whether the workload is cloud-native.
- **Targets:** In the plan, describe and analyze the target assets or services under consideration. If needed, map all of the components to be monitored, including service dependencies. Identify coverage gaps and determine who owns each part of the service.
- **Solution:** For the monitoring solution, identify the consumers, stakeholders, suppliers, partners, access, and instrumentation. Also monitoring aspects, focus, response, and reports and dashboards (availability, security, user experience, and so on).

Lean and agile considerations

- **Minimum viable product:** Let the plan define the minimum viable product, that is what is initially needed to go live, then continue to evolve the monitoring solution to maximize value. For example, you may define a future task later to build other log-based workbooks, pinning to Azure dashboards, and expanding stakeholder access to the Azure portal.
- **Start where you are and get value fast:** Rapidly and radically experiment, exploit with SaaS out of the box, because it's easy and valuable.
- **Know the guardrails:** Cloud is new and uncertain, so let experts guide you so that you don't add risk (for example, exposing sensitive monitoring data).
- **Microsoft 365 depends on Azure:** Any good plan considers your Azure tenant with Microsoft 365 as a major player. Microsoft 365 depends on Azure AD, and Azure Monitor provides Microsoft 365 integration with endpoint management.
- **Observability wins:** Focus on total visibility before alerting, as alerting is now a cost.
- **Log architecture:** Emitters, telemetry, signals, AI, and prepackaged solutions such as the Intelligent Security Graph. Focus more on a SIEM approach, how to amass disparate logs into log solutions such as Azure Monitor Logs.
- **Moving data:** Much more monitoring focus on emitter-to-dashboard and where data is going over the internet and across your cloud regions, ExpressRoute connections, and VPNs.
- **Activity monitoring:** Audit, sign-in, and activity logs are now easy to slice and dice for service owners and security.

Next steps

Collect the right data

Cloud monitoring guide: Collect the right data

5/4/2021 • 2 minutes to read • Edit Online

This article describes some considerations for collecting monitoring data in a cloud application.

To observe the health and availability of your cloud solution, you must configure the monitoring tools to collect a level of signals that are based on predictable failure states. These signals are the symptoms of the failure, not the cause. The monitoring tools use metrics and, for advanced diagnostics and root cause analysis, logs.

Plan for monitoring and migration carefully. Start by including the monitoring service owner, the manager of operations, and other related personnel during the planning phase, and continue engaging them throughout the development and release cycle. Their focus will be to develop a monitoring configuration that's based on the following criteria:

- What is the composition of the service? Are those dependencies monitored today? If so, are there multiple tools involved? Is there an opportunity to consolidate, without introducing risks?
- What is the SLA of the service, and how will I measure and report it?
- What should the service dashboard look like when an incident is raised? What should the dashboard look like for the service owner, and for the team that supports the service?
- What metrics does the resource produce that I need to monitor?
- How will the service owner, support teams, and other personnel be searching the logs?

How you answer those questions, and the criteria for alerting, determines how you'll use the monitoring platform. If you're migrating from an existing monitoring platform or set of monitoring tools, use the migration as an opportunity to reevaluate the signals you collect. This is especially true now that there are several cost factors to consider when you migrate or integrate with a cloud-based monitoring platform like Azure Monitor. Remember, monitoring data needs to be actionable. You need to have optimized data collected to give you "a 10,000 foot view" of the overall health of the service. The instrumentation that's defined to identify real incidents should be as simple, predictable, and reliable as possible.

Develop a monitoring configuration

The monitoring service owner and team typically follow a common set of activities to develop a monitoring configuration. These activities start at the initial planning stages, continue through testing and validating in a nonproduction environment, and extend to deploying into production. Monitoring configurations are derived from known failure modes, test results of simulated failures, and the experience of several people in the organization (the service desk, operations, engineers, and developers). Such configurations assume that the service already exists, it's being migrated to the cloud, and it hasn't been rearchitected.

For service-level quality results, monitor the health and availability of these services early in the development process. If you monitor the design of that service or application as an afterthought, your results won't be as successful.

To drive quicker resolution of the incident, consider the following recommendations:

- Define a dashboard for each service component.
- Use metrics to help guide further diagnosis and to identify a resolution or workaround of the issue if a root cause can't be uncovered.
- Use dashboard drill-down capabilities, or support customizing the view to refine it.
- If you need verbose logs, metrics should have helped target the search criteria. If the metrics didn't help, improve them for the next incident.

Embracing this guiding set of principles can help give you near-real-time insights, as well as better management of your service.

Next steps

Alerting strategy

Cloud monitoring guide: Alerting

5/4/2021 • 11 minutes to read • Edit Online

For years, IT organizations have struggled to combat the alert fatigue that's created by the monitoring tools deployed in the enterprise. Many systems generate a high volume of alerts often considered meaningless, while other alerts are relevant but are either overlooked or ignored. As a result, IT and developer operations have struggled to meet the service-level quality promised to internal or external customers. To ensure reliability, it's essential to understand the state of your infrastructure and applications. To minimize service degradation and disruption, or to decrease the effect of or reduce the number of incidents, you need to identify causes quickly.

Successful alerting strategy

You can't fix what you don't know is broken.

Alerting on what matters is critical. It's underpinned by collecting and measuring the right metrics and logs. You also need a monitoring tool capable of storing, aggregating, visualizing, analyzing, and initiating an automated response when conditions are met. You can improve the observability of your services and applications only if you fully understand its composition. You map that composition into a detailed monitoring configuration to be applied by the monitoring platform. This configuration includes the predictable failure states (the symptoms, not the cause of the failure) that make sense to alert for.

Consider the following principles for determining whether a symptom is an appropriate candidate for alerting:

- **Does it matter?** Is the issue symptomatic of a real problem or issue influencing the overall health of the application? For example, do you care whether the CPU utilization is high on the resource? Or that a particular SQL query running on a SQL database instance on that resource is consuming high CPU utilization over a sustained period? Because the CPU utilization condition is a real issue, you should alert on it. But you don't need to notify the team, because it doesn't help point to what is causing the condition in the first place. Alerting and notifying on the SQL query process utilization issue is both relevant and actionable.
- **Is it urgent?** Is the issue real, and does it need urgent attention? If so, the responsible team should be immediately notified.
- **Are your customers affected?** Are users of the service or application affected as a result of the issue?
- **Are other dependent systems affected?** Are there alerts from dependencies that are interrelated, and that can possibly be correlated to avoid notifying different teams all working on the same problem?

Ask these questions when you're initially developing a monitoring configuration. Test and validate the assumptions in a nonproduction environment, and then deploy into production. Monitoring configurations are derived from known failure modes, test results of simulated failures, and experience from different members of the team.

After the release of your monitoring configuration, you can learn a lot about what's working and what's not. Consider high alert volume, issues unnoticed by monitoring but noticed by end users, and what were the best actions to have taken as part of this evaluation. Identify changes to implement to improve service delivery, as part of an ongoing, continuous monitoring improvement process. It's not just about evaluating alert noise or missed alerts, but also the effectiveness of how you're monitoring the workload. It's about the effectiveness of your alert policies, process, and overall culture to determine whether you're improving.

Both System Center Operations Manager and Azure Monitor support alerts based on static or even dynamic thresholds and actions set up on top of them. Examples include alerts for email, SMS, and voice calls for simple notifications. Both of these services also support IT service management (ITSM) integration, to automate the creation of incident records and escalate to the correct support team, or any other alert management system

that uses a webhook.

When possible, you can use any of several services to automate recovery actions. These include System Center Orchestrator, Azure Automation, Azure Logic Apps, or autoscaling in the case of elastic workloads. While notifying the responsible teams is the most common action for alerting, automating corrective actions might also be appropriate. This automation can help streamline the entire incident management process. Automating these recovery tasks can also reduce the risk of human error.

Azure Monitor alerting

If you're using Azure Monitor exclusively, follow these guidelines as you consider speed, cost, and storage volume.

Depending on the feature and configuration you're using, you can store monitoring data in any of six repositories:

- **Azure Monitor metrics database:** A time-series database used primarily for Azure Monitor platform metrics, but also has Application Insights metric data mirrored into it. Information entering this database has the fastest alert times.
- **Application Insights resource:** A database that stores most Application Insights telemetry in log form.
- **Azure Monitor Log Analytics workspace:** The primary store for Azure log data. Other tools can route data to it and can be analyzed in Azure Monitor Logs. Because of ingestion and indexing, log alert queries have higher latency. This latency is generally 5-10 minutes, but can be higher under certain circumstances.
- **Activity log:** Used for all activity log and service health events. Dedicated alerting is possible. Holds subscription level events that occur on objects in your subscription, as seen from the outside of those objects. An example might be when a policy is set or a resource is accessed or deleted.
- **Azure Storage:** General-purpose storage that's supported by Azure Diagnostics and other monitoring tools. It's a low-cost option for long-term retention of monitoring telemetry. Alerting isn't supported from data that's stored in this service.
- **Event Hubs:** Generally used to stream data into on-premises or other partners' monitoring or ITSM tools.

Azure Monitor has four types of alerts, each somewhat tied to the repository that the data is stored in:

- Metric alert: Alerts on metric data in Azure Monitor. Alerts occur when a monitored value crosses a user-defined threshold, and then again when it returns to "normal" state.
- Log query alert: Available to alert on log data from Application Insights or Azure Monitor Logs. It can also alert based on cross-workspace queries.
- Activity log alert: Alerts on items in the activity log, with the exception of Azure Service Health data.
- Azure Service Health alert: A special type of alert that's used only for Azure Service Health issues that come from the activity log, such as outages and upcoming planned maintenance. Note that this type of alert is configured through Azure Service Health, a companion service to Azure Monitor.

Enable alerting through partner tools

If you're using an external alerting solution, route as much as you can through Azure Event Hubs, which is the fastest path out of Azure Monitor. You'll have to pay for ingestion into Event Hubs. If cost is an issue and speed isn't, you can use Azure Storage as a less expensive alternative. Just make sure that your monitoring or ITSM tools can read Azure Storage to extract the data.

Azure Monitor includes support for integrating with other monitoring platforms, and ITSM software such as

ServiceNow. You can use Azure alerting and still trigger actions outside of Azure, as required by your incident management or DevOps process. If you want to alert in Azure Monitor and automate the response, you can initiate automated actions by using Azure Functions, Azure Logic Apps, or Azure Automation, based on your scenario and requirements.

Specialized Azure monitoring offerings

Management solutions generally store their data Azure Monitor Logs. Two exceptions are Azure Monitor for VMs and Azure Monitor for containers. The following table describes the alerting experience based on the particular data type and where it's stored.

SOLUTION	DATA TYPE	ALERT BEHAVIOR
Azure Monitor for containers	Calculated average performance data from nodes and pods are written to the metrics database.	Create metric alerts if you want to be alerted based on variation of measured utilization performance, aggregated over time.
	Calculated performance data that uses percentiles from nodes, controllers, containers, and pods are written to the workspace. Container logs and inventory information are also written to the workspace.	Create log query alerts if you want to be alerted based on variation of measured utilization from clusters and containers. Log query alerts can also be configured based on pod-phase counts and status node counts.
Azure Monitor for VMs	Health criteria are metrics stored in the metrics database.	Alerts are generated when the health state changes from healthy to unhealthy. This alert supports only action groups that are configured to send SMS or email notifications.
	Map and guest operating system performance log data is written to the Log Analytics workspace.	Create log query alerts.

Fastest speed, driven by cost

Latency is one of the most critical decisions driving alerting and a quick resolution of issues affecting your service. If you require near-real-time alerting under five minutes, evaluate first if you have or can get alerts on your telemetry where it's stored by default. In general, this strategy is also the cheapest option, because the tool you're using is already sending its data to that location.

That said, there are some important footnotes to this rule.

Guest OS telemetry has multiple paths to get into the system.

- The fastest way to alert on this data is to import it as custom metrics. Do this by using the Azure Diagnostics extension and then using a metric alert. However, custom metrics are currently in preview and are more expensive than other options.
- The least expensive, but with some ingestion latency, is to send it to a Log Analytics workspace. Running the Log Analytics agent on the VM is the best way to get all guest operating system metric and log data into the workspace.
- You can send it for storage as a metric and a log in Azure Monitor by running both the Azure Diagnostics extension and the Log Analytics agent on the same VM. You can then alert quicker, but also use the guest operating system data as part of more complex queries when you combine it with other telemetry.

Importing data from on-premises: If you're trying to query and monitor across machines running in Azure and on-premises, you can use the Log Analytics agent to collect guest operating system data. You can then use a

feature called logs to metrics to collect and store as metrics in Azure Monitor. This method bypasses part of the ingestion process into Azure Monitor Logs, and the data is thus available sooner.

Minimize alerts

If you use a solution such as Azure Monitor for VMs and find the default health criteria that monitors performance utilization acceptable, don't create overlapping metric or log query alerts based on the same performance counters.

If you aren't using Azure Monitor for VMs, make the job of creating alerts and managing notifications easier by exploring the following features:

> **NOTE**
>
> These features apply only to metric alerts, alerts based on data that's being sent to the Azure Monitor metric database. The features don't apply to the other types of alerts. As mentioned previously, the primary objective of metric alerts is speed. If getting an alert in less than five minutes isn't of primary concern, you can use a log query alert instead.

- Dynamic thresholds: Dynamic thresholds look at the activity of the resource over a time period, and create upper and lower "normal behavior" thresholds. When the metric being monitored falls outside of these thresholds, you get an alert.
- Multisignal alerts: You can create a metric alert that uses the combination of two different inputs from two different resource types. For example, if you want to fire an alert when the CPU utilization of a VM is over 90 percent, and the number of messages in a certain Azure Service Bus queue feeding that VM exceeds a certain amount, you can do so without creating a log query. This feature works for only two signals. If you have a more complex query, feed your metric data into the Log Analytics workspace, and use a log query.
- Multiresource alerts: Azure Monitor allows a single metric alert rule that applies to all VM resources. This feature can save you time because you don't need to create individual alerts for each VM. Pricing for this type of alert is the same. Whether you create 50 alerts for monitoring CPU utilization for 50 VMs, or one alert that monitors CPU utilization for all 50 VMs, it costs you the same amount. You can use these types of alerts in combination with dynamic thresholds as well.

Used together, these features can save time by minimizing alert notifications and the management of the underlying alerts.

Limits on alerts

Be sure to note the limits on the number of alerts you can create. Some limits (but not all of them) can be increased by calling support.

Best query experience

If you're looking for trends across all your data, it makes sense to import all your data into Azure Monitor Logs, unless it's already in Application Insights. You can create queries across both workspaces, so there's no need to move data between them. You can also import activity log and Azure Service Health data into your Log Analytics workspace. You pay for this ingestion and storage, but you get all your data in one place for analysis and querying. This approach also gives you the ability to create complex query conditions and alert on them.

Cloud monitoring guide: Monitoring platforms overview

5/4/2021 • 14 minutes to read • Edit Online

Microsoft provides a range of monitoring capabilities from two products: System Center Operations Manager, which was designed for on-premises and then extended to the cloud, and Azure Monitor, which was designed for the cloud but can also monitor on-premises systems. These two offerings deliver core monitoring services, such as alerting, service uptime tracking, application and infrastructure health monitoring, diagnostics, and analytics.

Many organizations are embracing the latest practices for DevOps agility and cloud innovations to manage their heterogenous environments. Yet they are also concerned about their ability to make appropriate and responsible decisions about how to monitor those workloads.

This article provides a high-level overview of our monitoring platforms to help you understand how each delivers core monitoring functionality.

The story of System Center Operations Manager

In 2000, we entered the operations management field with Microsoft Operations Manager 2000. In 2007, we introduced a reengineered version of the product, System Center Operations Manager. It moved beyond simple monitoring of Windows Server instances and concentrated on robust, end-to-end service and application monitoring, including heterogenous platforms, network devices, and other application or service dependencies. It's an established, enterprise-grade monitoring platform for on-premises environments, in the same class as IBM Tivoli or HP Operations Manager in the industry. It has grown to support monitoring compute and platform resources running in Azure, Amazon Web Services (AWS), and other cloud providers.

The story of Azure Monitor

When Azure was released in 2010, monitoring of cloud services was provided with the Azure Diagnostics agent, which provided a way to collect diagnostics data from Azure resources. This capability was considered a general monitoring tool rather than an enterprise-class monitoring platform.

Application Insights was introduced to shift with changes in the industry where proliferation of cloud, mobile, and IoT devices was growing and the introduction of DevOps practices. It grew from application performance monitoring in Operations Manager to a service in Azure, where it delivers rich monitoring of web applications written in a variety of languages. In 2015, the preview of Application Insights for Visual Studio was announced and later, it became known as just Application Insights. It collects details about application performance, requests and exceptions, and traces.

In 2015, Azure Operational Insights was made generally available. It delivered the Log Analytics service that collected and searched data from machines in Azure, on-premises, or other cloud environments, and connected to System Center Operations Manager. Intelligence packs were offered that delivered a variety of prepackaged management and monitoring configurations that contained a collection of query and analytic logic, visualizations, and data collection rules for such scenarios as security auditing, health assessments, and alert management. Later, Azure Operational Insights became known as Log Analytics.

In 2016, the preview of Azure Monitor was announced at the Microsoft Ignite conference. It provided a common framework to collect platform metrics, resource diagnostics logs, and subscription-level activity log events from any Azure service that started using the framework. Previously, each Azure service had its own monitoring

method.

At the 2018 Ignite conference, we announced that the Azure Monitor brand expanded to include several different services originally developed with independent functionality:

- The original **Azure Monitor**, for collecting platform metrics, resource diagnostics logs, and activity logs for Azure platform resources only.
- **Application Insights**, for application monitoring.
- **Log Analytics**, the primary location for collecting and analyzing log data.
- A new **unified alerting service**, which brought together alert mechanisms from each of the other services mentioned earlier.
- **Azure Network Watcher**, for monitoring, diagnosing, and viewing metrics for resources in a virtual network.

The story of Operations Management Suite (OMS)

From 2015 until April 2018, Operations Management Suite (OMS) bundled the following Azure management services for licensing purposes:

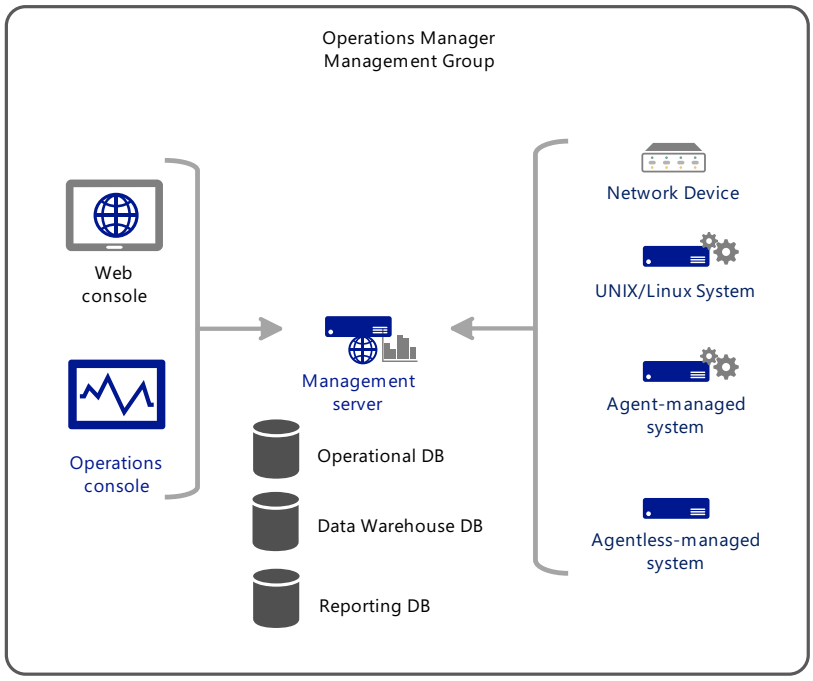
- Application Insights
- Azure Automation
- Azure Backup
- Operational Insights (later rebranded as Log Analytics)
- Site Recovery

The functionality of the services that were part of OMS did not change when OMS was discontinued. They were realigned under Azure Monitor.

Infrastructure requirements

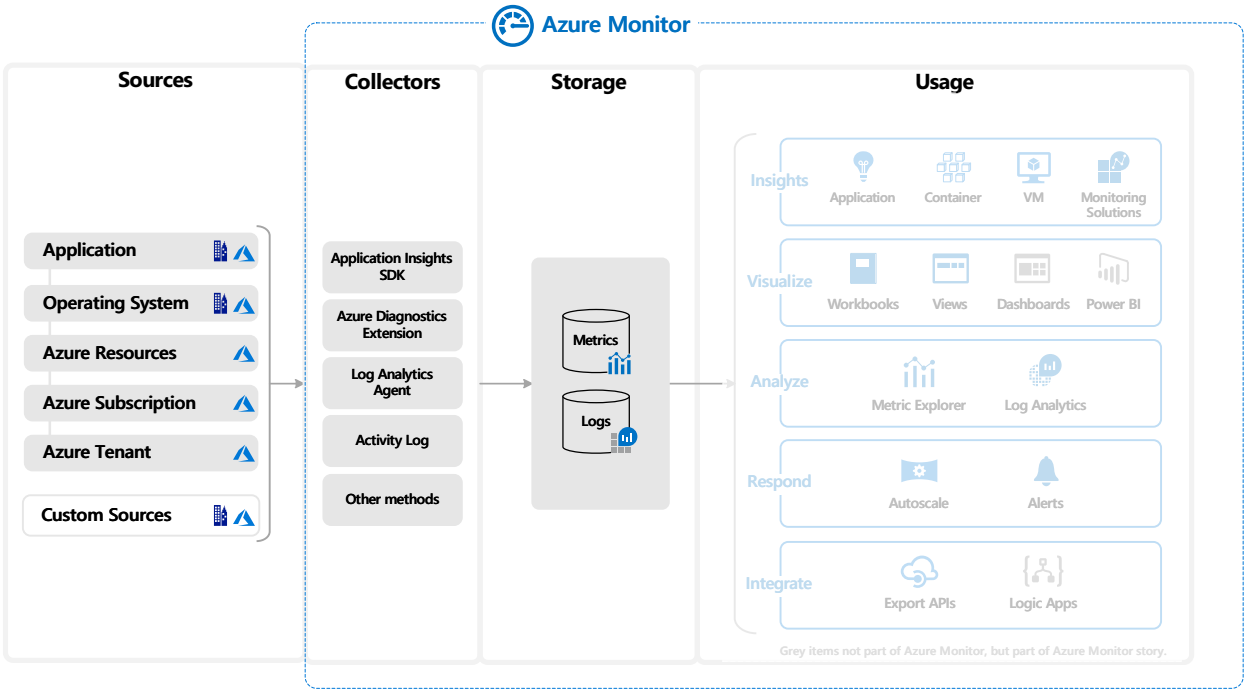
Operations Manager

Operations Manager requires significant infrastructure and maintenance to support a management group, which is a basic unit of functionality. At a minimum, a management group consists of one or more management servers, a SQL Server instance, hosting the operational and reporting data warehouse database, and agents. The complexity of a management group design depends on multiple factors, such as the scope of workloads to monitor, and the number of devices or computers supporting the workloads. If you require high availability and site resiliency, as is commonly the case with enterprise monitoring platforms, the infrastructure requirements and associated maintenance can increase dramatically.



Azure Monitor

Azure Monitor is a software as a service (SaaS) offering, so its supporting infrastructure runs in Azure and is managed by Microsoft. It's performs monitoring, analytics, and diagnostics at scale. It's available in all national clouds. Core parts of the infrastructure (collectors, metrics and logs store, and analytics) that support Azure Monitor are maintained by Microsoft.

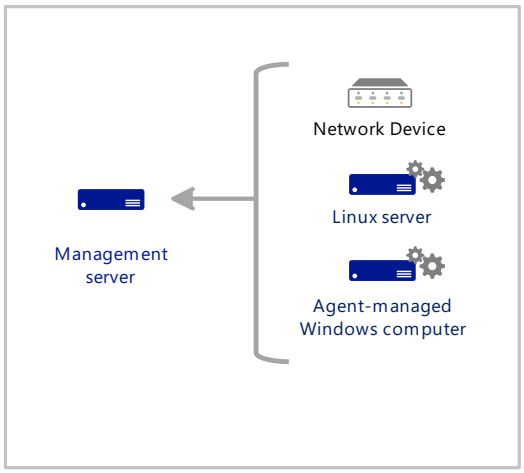


Data collection

Operations Manager

Agents

Operations Manager collects data directly only from agents that are installed on Windows computers. It can accept data from the Operations Manager SDK, but this approach is typically used for partners that extend the product with custom applications, not for collecting monitoring data. It can collect data from other sources, such as Linux computers and network devices, by using special modules that run on the Windows agent that remotely accesses these other devices.



The Operations Manager agent can collect from multiple data sources on the local computer, such as the event log, custom logs, and performance counters. It can also run scripts, which can collect data from the local computer or from external sources. You can write custom scripts to collect data that can't be collected by other means, or to collect data from a variety of remote devices that can't otherwise be monitored.

Management packs

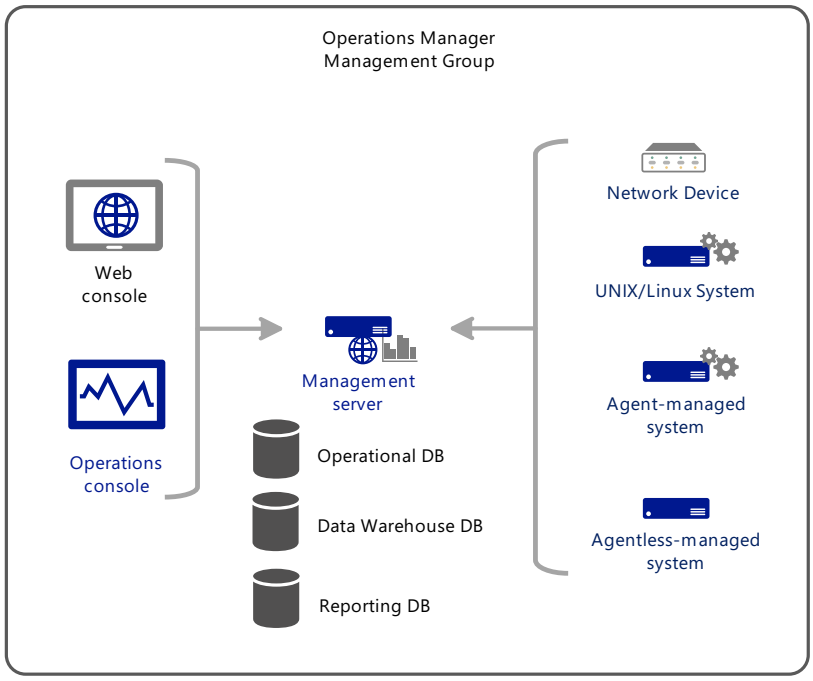
Operations Manager performs all monitoring with workflows (rules, monitors, and object discoveries). These workflows are packaged together in a management pack and deployed to agents. Management packs are available for a variety of products and services, which include predefined rules and monitors. You can also author your own management pack for your own applications and custom scenarios.

Monitoring configuration

Management packs can contain hundreds of rules, monitors, and object discovery rules. An agent runs all these monitoring settings from all the management packs that apply, which are determined by discovery rules. Each instance of each monitoring setting runs independently and acts immediately on the data that it collects. This is how Operations Manager can achieve near-real-time alerting and the current health state of monitored resources.

For example, a monitor might sample a performance counter every few minutes. If that counter exceeds a threshold, it immediately sets the health state of its target object, which immediately triggers an alert in the management group. A scheduled rule might watch for a particular event to be created and immediately fire an alert when that event is created in the local event log.

Because these monitoring settings are isolated from each other and work from the individual sources of data, Operations Manager has challenges correlating data between multiple sources. It's also difficult to react to data after it's been collected. You can run workflows that access the Operations Manager database, but this scenario isn't common, and it's typically used for a limited number of special purpose workflows.



Azure Monitor

Data sources

Azure Monitor collects data from a variety of sources, including Azure infrastructure and platform resources, agents on Windows and Linux computers, and monitoring data collected in Azure Storage. Any REST client can write log data to Azure Monitor by using an API, and you can define custom metrics for your web applications. Some metric data can be routed to different locations, depending on its usage. For example, you might use the data for "as-fast-as-possible" alerting or for long-term trend analysis searches in conjunction with other log data.

Monitoring solutions and insights

Monitoring solutions use the logs platform in Azure Monitor to provide monitoring for a particular application or service. They typically define data collection from agents or from Azure services, and provide log queries and views to analyze that data. They typically don't provide alert rules, which means that you must define your own alert criteria based on collected data.

Insights, such as Azure Monitor for containers and Azure Monitor for VMs, use the logs and metrics platform of Azure Monitor to provide a customized monitoring experience for an application or service in the Azure portal. They might provide health monitoring and alerting conditions, in addition to customized analysis of collected data.

Monitoring configuration

Azure Monitor separates data collection from actions taken against that data, which supports distributed microservices in a cloud environment. It consolidates data from multiple sources into a common data platform, and provides analysis, visualization, and alerting capabilities based on the collected data.

Data collected by Azure Monitor is stored as either logs or metrics, and different features of Azure Monitor rely on either. Metrics contain numerical values in time series that are well suited for near-real-time alerting and quick detection of issues. Logs contain text or numerical data and can be queried using a powerful language especially useful for performing complex analysis.

Because Azure Monitor separates data collection from actions against that data, it might be unable to provide near-real-time alerting in many cases. To alert on log data, queries are run on a recurring schedule defined in the alert. This behavior allows Azure Monitor to easily correlate data from all monitored sources, and you can interactively analyze data in a variety of ways. This is especially helpful for doing root cause analysis and identifying where else an issue might occur.

Health monitoring

Operations Manager

Management packs in Operations Manager include a service model that describes the components of the application being monitored and their relationship. Monitors identify the current health state of each component based on data and scripts on the agent. Health states roll up so that you can quickly view the summarized health state of monitored computers and applications.

Azure Monitor

Azure Monitor doesn't provide a user-definable method of implementing a service model or monitors that indicate the current health state of any service components. Because monitoring solutions are based on standard features of Azure Monitor, they don't provide state-level monitoring. The following features of Azure Monitor can be helpful:

- **Application Insights:** Builds a composite map of your web application, and provides a health state for each application component or dependency. This includes alerts status and drill-down to more detailed diagnostics of your application.
- **Azure Monitor for VMs:** Delivers a health-monitoring experience for the guest Azure VMs, similar to that of Operations Manager, when it monitors Windows and Linux virtual machines. It evaluates the health of key operating system components from the perspective of availability and performance to determine the current health state. When it determines that the guest VM is experiencing sustained resource utilization, disk-space capacity, or an issue related to core operating system functionality, it generates an alert to bring this state to your attention.
- **Azure Monitor for containers:** Monitors the performance and health of Azure Kubernetes Service or Azure Container Instances. It collects memory and processor metrics from controllers, nodes, and containers that are available in Kubernetes through the metrics API. It also collects container logs and inventory data about containers and their images. Predefined health criteria that are based on the collected performance data help you identify whether a resource bottleneck or capacity issue exists. You can also understand the overall performance, or the performance from a specific Kubernetes object type (pod, node, controller, or container).

Analyze data

Operations Manager

Operations Manager provides four basic ways to analyze data after it has been collected:

- **Health Explorer:** Helps you discover which monitors are identifying a health state issue and review knowledge about the monitor and possible causes for actions related to it.
- **Views:** Offers predefined visualizations of collected data, such as a graph of performance data or a list of monitored components and their current health state. Diagram views visually present the service model of an application.
- **Reports:** Allow you to summarize historical data that's stored in the Operations Manager data warehouse. You can customize the data that views and reports are based on. However, there is no feature to allow for complex or interactive analysis of collected data.
- **Operations Manager command shell:** Extends Windows PowerShell with an additional set of cmdlets, and can query and visualize collected data. This includes graphs and other visualizations, natively with PowerShell, or with the Operations Manager HTML-based web console.

Azure Monitor

With the powerful Azure Monitor analytics engine, you can interactively work with log data and combine them

with other monitoring data for trending and other data analysis. Views and dashboards allow you to visualize query data in a variety of ways from the Azure portal, and import it into Power BI. Monitoring solutions include queries and views to present the data they collect. Insights such as Application Insights, Azure Monitor for VMs, and Azure Monitor for containers include customized visualizations to support interactive monitoring scenarios.

Alerting

Operations Manager

Operations Manager creates alerts in response to predefined events, when a performance threshold is met, and when the health state of a monitored component changes. It includes the complete management of alerts, allowing you to set their resolution and assign them to various operators or system engineers. You can set notification rules that specify which alerts will send proactive notifications.

Management packs include various predefined alerting rules for different critical conditions in the application being monitored. You can tune these rules or create custom rules to the particular requirements of your environment.

Azure Monitor

With Azure Monitor, you can create alerts based on a metric crossing a threshold, or based on a scheduled query result. Although alerts based on metrics can achieve near-real-time results, scheduled queries have a longer response time, depending on the speed of data ingestion and indexing. Instead of being limited to a specific agent, log query alerts in Azure Monitor let you analyze data across all data stored in multiple workspaces. These alerts also include data from a specific Application Insights application by using a cross-workspace query.

Although monitoring solutions can include alert rules, you ordinarily create them based on your own requirements.

Workflows

Operations Manager

Management packs in Operations Manager contain hundreds of individual workflows, and they determine both what data to collect and what action to perform with that data. For example, a rule might sample a performance counter every few minutes, storing its results for analysis. A monitor might sample the same performance counter and compare its value to a threshold to determine the health state of a monitored object. Another rule might run a script to collect and analyze some data on an agent computer, and then fire an alert if it returns a particular value.

Workflows in Operations Manager are independent of each other, which makes analysis across multiple monitored objects difficult. These monitoring scenarios must be based on data after it's collected, which is possible but can be difficult, and it isn't common.

Azure Monitor

Azure Monitor separates data collection from actions and analysis taken from that data. Agents and other data sources write log data to a Log Analytics workspace and write metric data to the metric database, without any analysis of that data or knowledge of how it might be used. Monitor performs alerting and other actions from the stored data, which allows you to perform analysis across data from all sources.

Extend the base platform

Operations Manager

Operations Manager implements all monitoring logic in a management pack, which you either create yourself or obtain from us or a partner. When you install a management pack, it automatically discovers components of the application or service on different agents, and deploys appropriate rules and monitors. The management

pack contains health definitions, alert rules, performance and event collection rules, and views, to provide complete monitoring that supports the infrastructure service or application.

The Operations Manager SDK enables Operations Manager to integrate with third-party monitoring platforms or IT service management (ITSM) software. The SDK is also used by some partner management packs to support monitoring network devices and deliver custom presentation experiences, such as the Squared Up HTML5 dashboard or integration with Microsoft Office Visio.

Azure Monitor

Azure Monitor collects metrics and logs from Azure resources, with little to no configuration. Monitoring solutions add logic for monitoring an application or service, but they still work within the standard log queries and views in Azure Monitor. Insights, such as Application Insights and Azure Monitor for VMs, use the Azure Monitor platform for data collecting and processing. They also provide additional tools to visualize and analyze the data. You can combine data collected by insights with other data, by using core Azure Monitor features such as log queries and alerts.

Monitor supports several methods to collect monitoring or management data from Azure or external resources. You can then extract and forward data from the metric or log stores to your ITSM or monitoring tools. Or you can perform administrative tasks by using the Azure Monitor REST API.

Next steps

Monitoring the cloud deployment models

Skills readiness for cloud monitoring

3/6/2021 • 6 minutes to read • Edit Online

When planning your migration journey, the objective is to develop the plans necessary to guide implementation. The plans need to also include how you will operate these workloads before they are transitioned or released into production, and not afterwards. Business stakeholders expect valuable services, and they expect them without disruption. IT staff members realize they need to learn new skills and adapt so they are prepared to confidently use the integrated Azure services to effectively monitor resources in Azure and hybrid environments.

Developing the necessary skills can be accelerated with the following learning paths. They are organized starting with learning the fundamentals and then divided across three primary subject domains: infrastructure, application, and data analysis.

Fundamentals

- Introduction to Azure Resource Manager discusses the basic concepts of management and deployment of Azure resources. The IT staff managing the monitoring experience across the enterprise should understand management scopes, Azure role-based access control (Azure RBAC), using. Azure Resource Manager templates, and management of resources using Azure CLI and Azure PowerShell.
- Introduction to Azure Policy helps you learn how you can use Azure Policy to create, assign, and manage policies. Azure Policy can deploy and configure the Azure Monitor agents, enable monitoring with Azure Monitor for VMs and Azure Security Center, deploy diagnostic settings, audit guest configuration settings, and more.
- Introduction to Azure command-line interface (CLI), which is our cross-platform command-line experience for managing Azure resources. Also review the introduction to Azure PowerShell. As part of their beginner-level course, learning Azure management tools, LinkedIn offers sessions covering Azure CLI and PowerShell programming languages:
  - Use the Azure CLI.
  - Get started with Azure PowerShell
- Learn how to secure resources using policy, Azure role-based access control, and other Azure services by viewing implement resource management security in Azure.
- Monitoring Microsoft Azure Resources and Workloads helps you learn how to use Azure monitoring tools to monitor Azure network resources as well as resources located on-premises.
- Learn about planning and designing your monitoring deployments at-scale and automating actions by viewing Azure Monitor best practices and recommendations.

Infrastructure monitoring

- Design a Monitoring Strategy for Infrastructure in Microsoft Azure helps you learn foundational knowledge of monitoring capabilities and solutions in Azure.
- How to monitor your Kubernetes clusters provides an intermediate level deep dive to help you learn about monitoring your Kubernetes cluster with Azure Monitor for containers.
- Learn with Azure Monitor how to monitor data from Azure Storage and HDInsight.

- Microsoft Azure Database Monitoring Playbook explores the key monitoring capabilities that can be used to gain insight and actionable steps for Azure SQL Database, Azure SQL Data Warehouse, and Azure Cosmos DB.
- Monitoring Microsoft Azure Hybrid Cloud Networks is an advanced-level course that helps you learn how to use Azure monitoring tools to visualize, maintain, and optimize virtual networks and virtual private network connections for your hybrid cloud implementation.
- With Azure Arc for servers, learn how you can manage your Windows and Linux machines hosted outside of Azure similarly to how you manage virtual machines running in Azure.
- How to monitor your VMs provides an intermediate level deep dive to help you learn about monitoring your hybrid machines or servers, and Azure VM or virtual machine scale sets with Azure Monitor for VMs.

Application monitoring

- Understand how Azure Monitor helps you view availability and performance of your applications and services together from one place. Pluralsight offers the following courses to help:
  - Microsoft Azure DevOps Engineer: Optimize Feedback Mechanisms helps you prepare you to use Azure Monitor, including Application Insights, to monitor and optimize your web applications.
  - Capture and view page load times in your Azure web app. Get started with this course on using Azure Monitor Application Insights for end-to-end monitoring of your applications components running in Azure.
  - Microsoft Azure Database Monitoring Playbook helps you learn how to implement and use monitoring of Azure SQL Database, Azure SQL Data Warehouse, and Azure Cosmos DB.
  - Instrument applications with Azure Monitor Application Insights is a deep-dive course on using the Application Insights SDK to collect telemetry and events from an application with angular and Node.js components.
  - Application Debugging and Profiling is a recording from a Microsoft conference session on using and interpreting data provided by the Azure Monitor Application Insights snapshot debugger and profiler.

Data analysis

- Learn how to write log queries in Azure Monitor. The Kusto Query Language is the primary resource for writing Azure Monitor log queries to explore and analyze log data between the collected data from Azure and hybrid resource application dependencies, including the live application.
- Kusto Query Language (KQL) from Scratch is a comprehensive course that includes detailed examples covering a wide range of use-cases and techniques for log analysis in Azure Monitor Logs.

Deeper skills exploration

Various learning options beyond these initial options are available for developing skills.

Typical mappings of cloud IT roles

Microsoft and partners offer various options for all audiences to develop skills with Azure services.

- Map roles and skills: A resource for mapping your cloud career path. Learn about your cloud role and suggested skills. Follow a learning curriculum at your own pace to build the skills that you need most to stay relevant.

- Explore Azure certification training and exams to gain official recognition for your Azure knowledge.

Azure DevOps and project management

The hybrid cloud environment disrupts IT with undefined roles, responsibilities, and activities. Organizations must move to modern practices to manage services, including agile and DevOps methodologies, to better meet the transformation and optimization needs of today's businesses in a streamlined and efficient manner.

As part of migrating to a cloud monitoring platform, the IT team responsible for managing monitoring in the enterprise needs to include agile training and participation in DevOps activities. This also includes following the *Dev* in DevOps by taking requirements and turning into organized agile requirements, in order to deliver minimally viable monitoring solutions that are refined iteratively and in line with business needs. For source control to manage the iterative monitoring solution packages and any other related collateral, connect your Azure DevOps Server project with a GitHub Enterprise Server repository. This provides a link from commits and pull requests in GitHub to work items. You can use GitHub Enterprise for development in support of continuous monitoring integration and deployment, while using Azure Boards to plan and track your work.

To learn more, review the following:

- Get started with Azure DevOps.
- Learn about DevOps Dojo white belt foundation.
- Evolve your DevOps practices.
- Automate your deployments with Azure DevOps.

Other considerations

Customers often struggle to manage, maintain, and deliver the expected business (and to the IT organization) outcomes for the services that IT is charged with delivering. Monitoring is considered core to managing infrastructure and the business, with a focus on measuring quality of service and customer experience. In order to achieve those goals, lay the groundwork using ITSM in conjunction with DevOps, which will help the monitoring team mature how they manage, deliver, and support the monitoring service. Adopting an ITSM framework allows the monitoring team to function as a provider and gain recognition as a trusted business enabler by aligning to the strategic goals and needs of the organization.

Review the following to understand the updates made to the most popular ITSM framework ITIL 4 and the Cloud white paper, which focuses on joining existing ITIL guidance with best practices from DevOps, agile, and lean approaches. Also consider the IT4IT reference architecture that delivers an alternative blueprint on how to transform IT using a process agnostic framework.

Learn more

To discover more learning paths, browse the Microsoft Learn catalog. Use the roles filter to align learning paths with your role.

Centralize management operations

5/4/2021 • 2 minutes to read • Edit Online

For most organizations, using a single Azure Active Directory (Azure AD) tenant for all users simplifies management operations and reduces maintenance costs. This is because all management tasks can be by designated users, user groups, or service principals within that tenant.

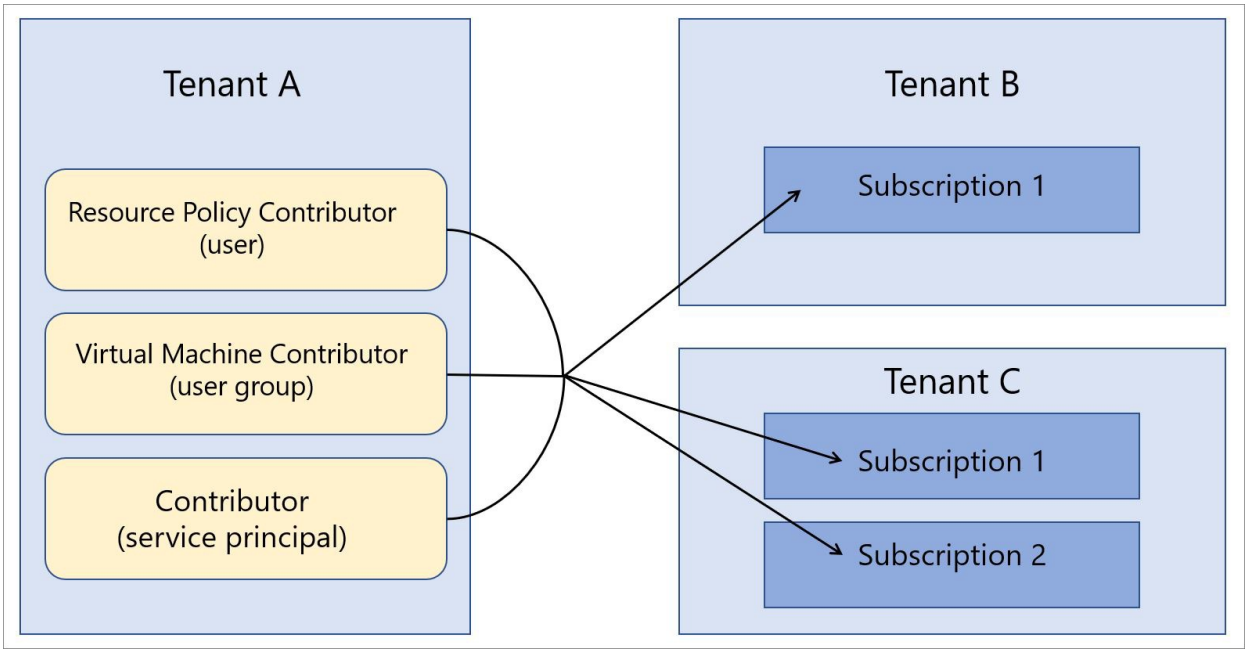
We recommend that you use only one Azure AD tenant for your organization, if possible. However, some situations might require an organization to maintain multiple Azure AD tenants for the following reasons:

- They are wholly independent subsidiaries.
- They're operating independently in multiple geographies.
- Certain legal or compliance requirements apply.
- There are acquisitions of other organizations (sometimes temporary until a long-term tenant consolidation strategy is defined).

When a multiple-tenant architecture is required, Azure Lighthouse provides a way to centralize and streamline management operations. Subscriptions from multiple tenants can be onboarded for Azure delegated resource management. This option allows specified users in the managing tenant to perform cross-tenant management functions in a centralized and scalable manner.

For example, let's say your organization has a single tenant, `Tenant A`. The organization then acquires two additional tenants, `Tenant B` and `Tenant C`, and you have business reasons that require you to maintain them as separate tenants.

Your organization wants to use the same policy definitions, backup practices, and security processes across all tenants. Because you already have users (including user groups and service principals) that are responsible for performing these tasks within `Tenant A`, you can onboard all of the subscriptions within `Tenant B` and `Tenant C` so that those same users in `Tenant A` can perform those tasks. `Tenant A` then becomes the managing tenant for `Tenant B` and `Tenant C`.



For more information, see Azure Lighthouse in enterprise scenarios.

Establish operations management processes

5/18/2021 • 9 minutes to read • Edit Online

As your enterprise begins to operate workloads in Azure, the next step is to establish a process for *operational management and fitness*. This process enumerates, implements, and iteratively reviews and optimizes the operational state for these workloads.

A process for operational fitness review ensures that the entire portfolio of workloads meet business commitments to performance, reliability, and cost. This process aligns the efforts of Central IT, Cloud Center of Excellence, and workload teams to deliver operational excellence at scale.

Establish a core process for operational fitness review

Create a process for operational fitness review to fully understand the problems that result from running workloads in a production environment, and how to remediate and resolve those problems. This article outlines a high-level process for operational fitness review that your enterprise can use to achieve this goal.

Operational fitness at Microsoft

From the outset, many teams across Microsoft have been involved in the development of the Azure platform. It's difficult to ensure quality and consistency for a project of such size and complexity. You need a robust process to enumerate and implement fundamental nonfunctional requirements on a regular basis.

The processes that Microsoft follows form the basis for the processes outlined in this article.

Understand roles & operating models

Operations management is a broad discipline involving multiple roles across the company. Depending on the organizations operating model, those roles may operate in a matrixed environment with a number of hand offs between centralized and decentralized operations teams.

- **Central IT / CCoE:** This centralized technology function is responsible for the configuration, operations, governance, and security of all technology assets in the technology portfolio.
- **Cloud Operations:** A function within the centralized technology organization, this centralized operations function is responsible for the health and operations of the technology portfolio. It is their responsibility to ensure the process is run smoothly, that each adjacent role in the process has the tools they need, & that each of the subsequent roles is held accountable for expectations of this process.
- **Cloud Strategy:** Provides knowledge of the business to identify and prioritize commitments to maintain operational requirements of various workloads. This role also compares the mitigation cost to the business impact, and drives the final decision on remediation.
- **Workload Team:** Accountable for development and operations of discreet workloads which map to specific supporting applications, services, and infrastructure, whether on-premises or in the cloud. The role requires deep knowledge of the workload architecture.

Each organization's operating model determines the accountability and day-to-day activities of the roles above:

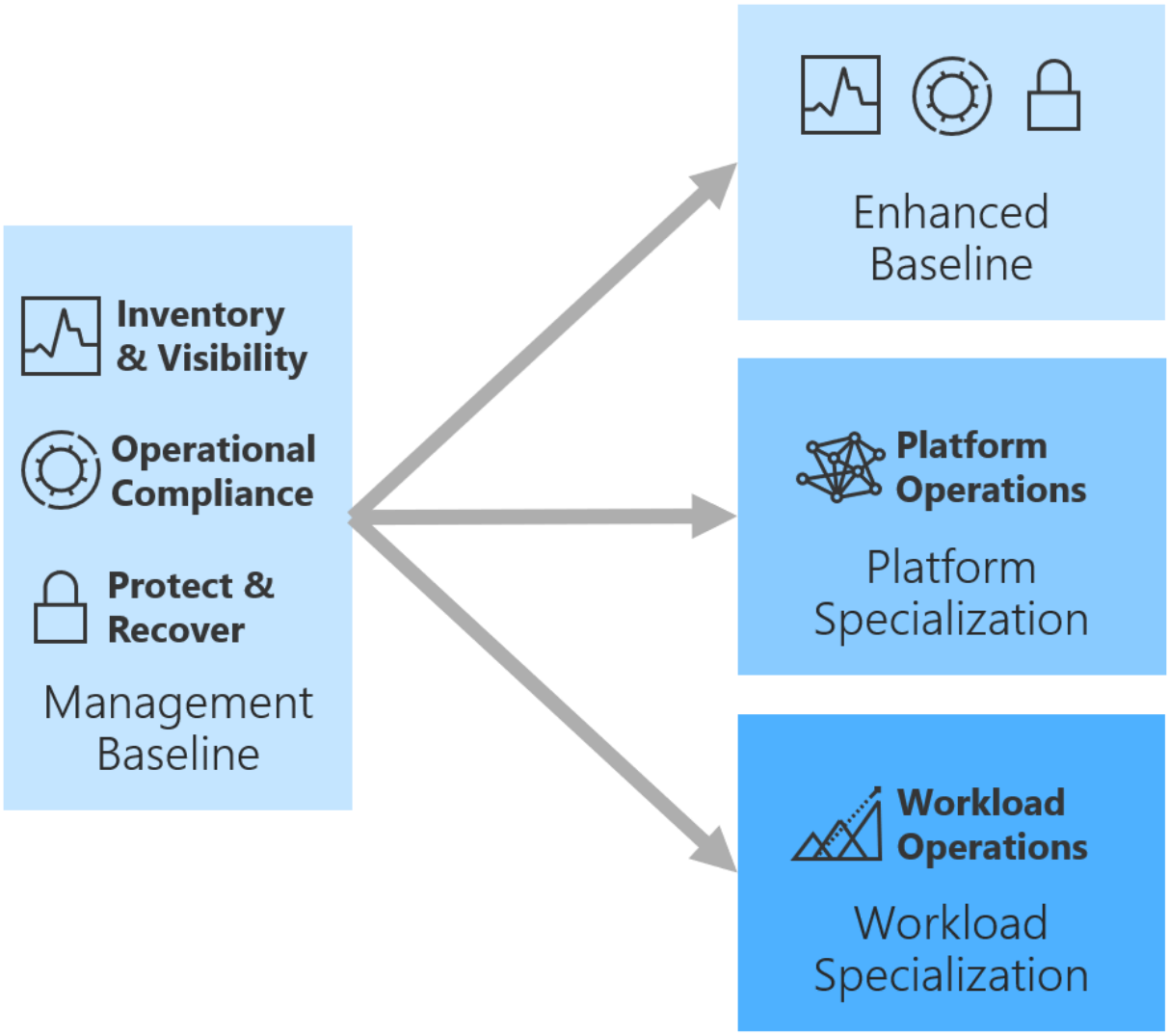
- **Centralized operations:** Central IT maintains full accountability for operations. Workload owners may have input to operations & configuration but they have no access to change production environments. Only Central IT and Cloud Operations can deliver operational change to improve operational fitness.
- **Decentralized operations:** Workload teams are fully accountable for operations, generally through a mature CI/CD pipeline and DevOps automation. In this model, there is no central support for configuration,

operations, governance, or security. This approach to operations is out of scope for the Cloud Adoption Framework. This operating model should see the Azure Well-Architected Framework for operational guidance.

- **Enterprise operations:** Cloud Center of Excellence is accountable for operations. Cloud Operations and Workload Teams each share responsibility for specific aspects of operational fitness.

Objective of the review

Operational fitness is evaluated across the portfolio using a few metrics: Reliability, Performance, and Cost. Together, these properties allow for rapid evaluation of the health and fitness of all assets in the portfolio. These metrics are evaluated across the three elevations of operations management.



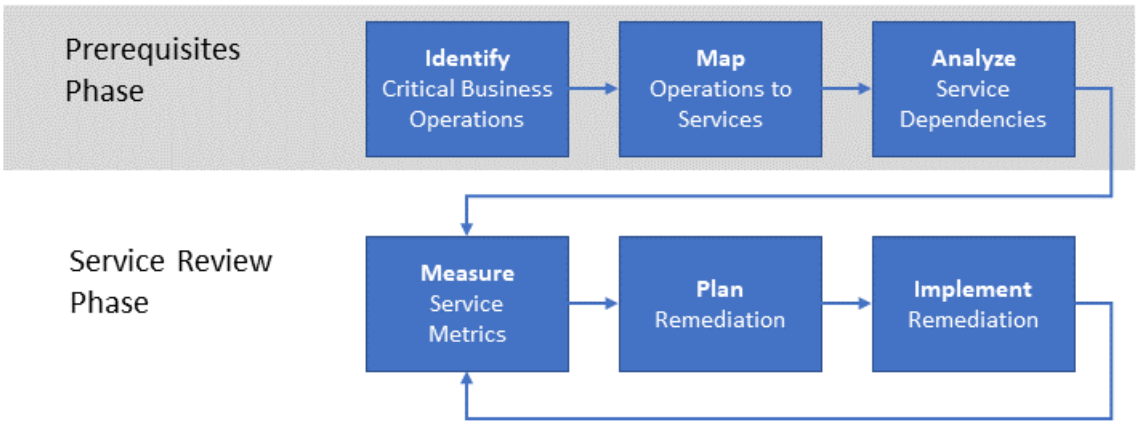
- **Operations baseline (or enhanced baseline):** Evaluates operational fitness across all deployed assets regardless of their function. This broad view of operations allows for sweeping changes and big impacts, but is limited by a lack of visibility into the architecture of individual workloads. All resources deployed in the cloud should be covered by an operations baseline with regular support from cloud operations. Some environments may require a higher degree of operational support to meet the needs of the enhanced baseline.
- **Platform operations:** Evaluates operational fitness of centralized technology platforms. This view of operations is more refined since it considers the architecture of the platform and how changes to the solution will impact operational fitness. Changes to central technology platforms can have a broad downstream impact on supported workloads. All mission-critical platforms should receive dedicated support from a central IT team.

- **Workload operations:** Evaluates operational fitness of an individual workload. This view of operations is most refined & should be considered when operational fitness improvements require changes to the architecture of a workload. Workload operations should adhere to the principles of the Azure Well-Architected Framework. All mission-critical workloads with an active DevOps cycle should receive dedicated support from a workload team.

The objective of the operational fitness review is to regularly evaluate operational fitness at all levels. Identified improvements can then be applied at the appropriate level to inform the changes required to manage the overall portfolio.

Process for operational fitness review

The key to maintaining the performance and continuity of an enterprise's portfolio is to implement a process for operational fitness review.



At a high level, the process has two phases. In the *prerequisites phase*, the requirements are established and mapped to supporting services. This phase occurs infrequently: perhaps annually or when new operations are introduced. The output of the prerequisites phase is used in the *flow phase*. The flow phase occurs more frequently, such as monthly.

Prerequisites phase

The steps in this phase capture the requirements for conducting a regular review of the portfolio and any mission critical workloads.

1. **Identify critical business operations.** Identify the enterprise's mission-critical business operations based on agreed upon business commitments. Business operations are independent from any supporting service functionality. In other words, business operations represent the actual activities that the business needs to perform and that are supported by a set of IT services.

   The term *mission-critical* (or *business-critical*) reflects a severe impact on the business if the operation is impeded. For example, an online retailer might have a business operation, such as "enable a customer to add an item to a shopping cart" or "process a credit card payment." If either of these operations fails, a customer can't complete the transaction and the enterprise fails to realize sales.

2. **Map operations to services.** Map the critical business operations to the IT services (Baseline, Platform or Workload operations) that support them. Any technology platform or workload required to support a critical business function should also be identified to map operations and services to responsible teams.

3. **Analyze service dependencies.** Most business operations require orchestration among multiple supporting workloads and technology platforms. It's important to understand the dependencies between each set of supporting assets, and the flow of mission-critical transactions through these services.

   Also consider the dependencies between on-premises services and Azure services. In the shopping-cart

example, the inventory stock-management service might be hosted on-premises and ingest data entered by employees from a physical warehouse. However, it might store data off-premises in an Azure service, such as Azure Storage, or a database, such as Azure Cosmos DB.

An output from these activities is a set of *scorecard metrics* for operations management. The scorecard measures criteria such as Reliability, Performance, and Cost. Scorecard metrics express the operational criteria that you expect the service to meet.

The scorecard should be expressed in simple terms to facilitate meaningful discussion between the business owners, cloud operations, and workload teams. For example, a scorecard metric for reliability might be color-coded based on attainment of the agreed SLA. Green means meeting the defined SLA, yellow means failing to meet the defined criteria but actively implementing a planned remediation, and red means failing to meet the defined criteria with no plan or action.

It's important to emphasize that these metrics should directly reflect business commitments.

Service-review phase

The service-review phase is the core of the operational fitness review. It involves these steps:

1. **Measure service metrics.** Use the scorecard metrics to monitor performance at each level of operations management, to ensure that the services meet the business commitments. Inventory & visibility services within the operations baseline is essential. If you can't monitor a set of resources with respect to the business commitments, consider the corresponding scorecard metrics to be red. In this case, the first step for remediation is to implement the appropriate service monitoring. For example, if the business expects a service to operate with 99.99 percent availability, but there is no production telemetry in place to measure availability, assume that you're not meeting the requirement.
2. **Plan remediation.** For each business commitment for which metrics fall below an acceptable threshold, determine the appropriate operations team to complete the required remediation. That team is responsible for calculating the cost of remediating the service to bring operations to an acceptable level. If the cost of remediating the issue is greater than the budget allocated to that service, Central IT/CCoE should review with the cloud strategy team to evaluate addition investments.
3. **Implement remediation.** After the cloud operations or workload team gain acceptance on a plan for remediation, implement it. Report the status of the implementation whenever you review scorecard metrics.

This process is iterative. The Central IT / CCoE team is accountable for managing the process & reporting on progress to the cloud strategy team. This team should meet regularly to review existing remediation projects, kick off the fundamental review of new workloads, and track the enterprise's overall scorecard. The team should also have the authority to hold remediation teams (Cloud operations or workload operations) accountable if they're behind schedule or fail to meet metrics.

Review meeting

We recommend that your operational fitness be reviewed on a regular basis. Central IT / CCoE and Cloud operations team are required attendance in the review. Cloud strategy and workload operations teams are encouraged to attend but are operational. Example cadence, the core team might meet monthly to align on plans & hold various operations teams accountable. Quarterly, the cloud strategy and all workload teams could join to understand status and metrics.

Adapt the details of the process and meeting to fit your specific needs. We recommend the following considerations as a starting point:

- Centralized operations: Workload teams are unlikely to participate actively in the process, but should be included in any reports for visibility.

- De-centralized operations: Cloud operations team should share best practices used to improve operations of tech platforms with the workload teams. Workload teams should share changes to their respective workloads to identify improvements that could be applied to tech platforms and the operations baseline.

Recommended resources

- Azure Automanage. Azure Automanage automatically monitors operational fitness across the operations baseline & automates the application of various remediation strategies across the portfolio.
- Azure Advisor. Azure Advisor provides personalized recommendations based on your usage and configurations to help optimize your resources. By default this tool provides recommendations across a subscription to improve the operations baseline. It can also be used more granularly to identify improvements to tech platforms or individual workloads.
- Microsoft Azure Well-Architected Framework: Guidance to improve workload operations or to guide decentralized operations.

IT management and operations in the cloud

5/4/2021 • 2 minutes to read • Edit Online

As a business moves to a cloud-based model, the importance of proper management and operations can't be overstated. Unfortunately, few organizations are prepared for the IT management shift that's required for success in building a cloud-first operating model. This section of the Cloud Adoption Framework outlines the operating model, processes, and tooling that have proven successful in the cloud. Each of these areas represents a minor but fundamental change in the way the business should view IT operations and management as it begins to adopt the cloud.

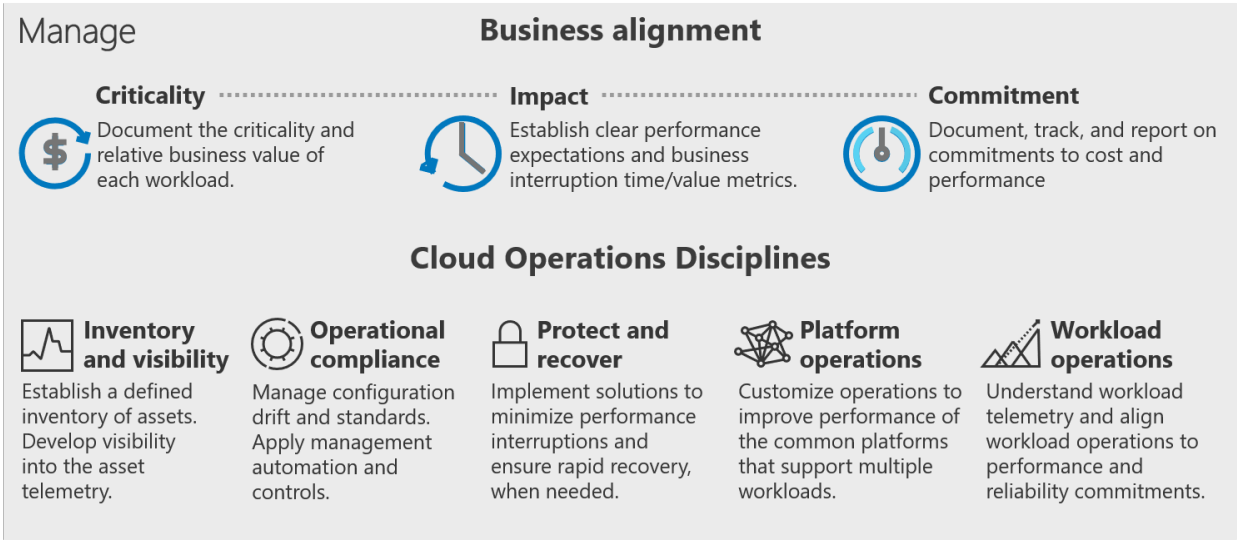
Brief history of IT management

Before the cloud, IT management grew from a simple acquisition function. Acquisition of technical equipment to support business processes required technical expertise and deep experience with a specific group of equipment vendors. IT management consolidated the selection, acquisition, and configuration of IT assets. Generally, the acquired assets included storage, computing power, networking, and other similar assets that are required to power the desired business function. As the primary subject matter experts on the equipment, IT was also tasked with operating the equipment to ensure maximum performance and minimal business disruptions.

When the business builds out new technology solutions, it has a clear need that can justify the significant expenses associated with acquiring assets, or even building out full datacenters. When it builds solutions, the business sees the acquisition costs as an investment in the future. After the business need is met, the perception of the same costs shifts. Costs that are associated with existing solutions are seen as operational drag that's created by past needs. That perception is why many businesses view IT as a cost center. It's also why many IT organizations experience regular cost-control exercises or reductions in IT staff.

Cloud management

The historical IT operating model was sufficient for over 20 years. But that model is now outdated and is less desirable than cloud-first alternatives. When IT management teams move to the cloud, they have an opportunity to rethink this model and drive greater value for the business. This article series outlines a modernized model of IT management.



Next steps

For a deeper understanding of the new cloud management model, start by understanding business alignment.

Understand business alignment

Create business alignment in cloud management

5/18/2021 • 2 minutes to read • Edit Online

In on-premises environments, IT assets (applications, virtual machines, VM hosts, disk, servers, devices, and data sources) are managed by IT to support workload operations. In IT terms, a workload is a collection of IT assets that support a specific business operation. To help support business operations, IT management delivers processes that are designed to minimize disruptions to those assets. When an organization moves to the cloud, management and operations shift a bit, creating an opportunity to develop tighter business alignment.

Business vernacular

The first step in creating business alignment is to ensure term alignment. IT management, like most engineering professions, has amassed a collection of jargon, or highly technical terms. Such terms can lead to confusion for business stakeholders and make it difficult to map management services to business value.

Fortunately, the process of developing a cloud adoption strategy and cloud adoption plan creates an ideal opportunity to remap these terms. The process also creates opportunities to rethink commitments to operational management, in partnership with the business. The following article series walks you through this new approach across three specific terms that can help improve conversations among business stakeholders:

- Criticality: Mapping workloads to business processes. Ranking criticality to focus investments.
- Impact: Understanding the impact of potential outages to aid in evaluating return on investment for cloud management.
- Commitment: Developing true partnerships, by creating and documenting agreements with the business.

> **NOTE**
> Underlying these terms are classic IT terms such as SLA, RTO, and RPO. For more information on mapping specific business and IT terms, see Business commitment in cloud management.

Operations management workbook

To help capture decisions that result from this conversation about term alignment, an operations management workbook is available on our GitHub site. This workbook does not perform SLA or cost calculations. It serves only to help capture such measures and forecast return on loss-avoidance efforts.

Alternatively, these same workloads and associated assets could be tagged directly in Azure, if the solutions are already deployed to the cloud.

Next steps

Start creating business alignment by defining workload criticality.

Define workload criticality

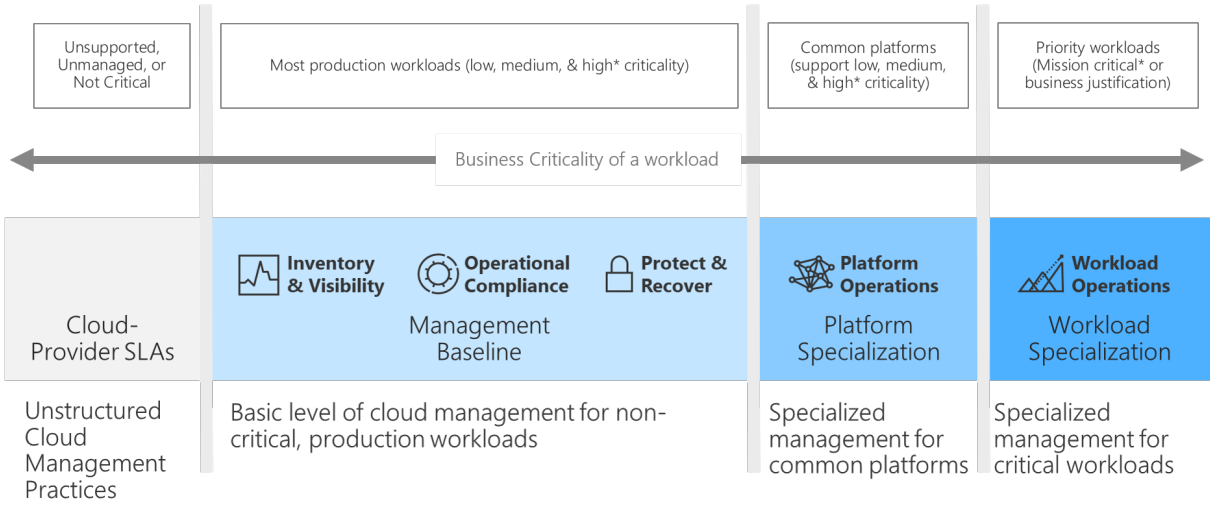
Business criticality in cloud management

5/24/2021 • 3 minutes to read • Edit Online

Across every business, there exist a small number of workloads that are too important to fail. These workloads are considered mission critical. When those workloads experience outages or performance degradation, the adverse impact on revenue and profitability can be felt across the entire company.

At the other end of the spectrum, some workloads can go months at a time without being used. Poor performance or outages for those workloads is not desirable, but the impact is isolated and limited.

Understanding the criticality of each workload in the IT portfolio is the first step toward establishing mutual commitments to cloud management. The following diagram illustrates a common alignment between the criticality scale to follow and the standard commitments made by the business.



Criticality scale

The first step in any business criticality alignment effort is to create a criticality scale. The following table presents a sample scale to be used as a reference, or template, for creating your own scale.

CRITICALITY	BUSINESS VIEW
Mission-critical	Affects the company's mission and might noticeably affect corporate profit-and-loss statements.
Unit-critical	Affects the mission of a specific business unit and its profit-and-loss statements.
High	Might not hinder the mission, but affects high-importance processes. Measurable losses can be quantified in the case of outages.
Medium	Impact on processes is likely. Losses are low or immeasurable, but brand damage or upstream losses are likely.

CRITICALITY	BUSINESS VIEW
Low	Impact on business processes isn't measurable. Neither brand damage nor upstream losses are likely. Localized impact on a single team is likely.
Unsupported	No business owner, team, or process that's associated with this workload can justify any investment in the ongoing management of the workload.

It's common for businesses to include additional criticality classifications that are specific to their industry, vertical, or specific business processes. Examples of additional classifications include:

- **Compliance-critical:** In heavily regulated industries, some workloads might be critical as part of an effort to maintain compliance requirements.
- **Security-critical:** Some workloads might not be mission critical, but outages could result in loss of data or unintended access to protected information.
- **Safety-critical:** When lives or the physical safety of employees and customers is at risk during an outage, it can be wise to classify workloads as safety-critical.

Importance of accurate criticality

Later in the cloud-adoption process, the cloud management team will use this classification to determine the amount of effort required to meet aligned levels of criticality. In on-premises environments, operations management is often purchased centrally and treated as a necessary business burden, with little or no additional operating costs. Like all cloud services, operations management is purchased on a per-asset basis as monthly operating costs.

Because there's a clear and direct cost to operations management in the cloud, it's important to properly align costs and desired criticality scales.

Select a default criticality

An initial review of every workload in the portfolio can be time consuming. To ensure that this effort doesn't block your broader cloud strategy, we recommend that your teams agree on a default criticality to apply to all workloads.

Based on the preceding criticality-scale table, we recommend that you adopt *medium* criticality as the default. Doing so will allow your cloud strategy team to quickly identify workloads that require a higher level of criticality.

Review operational compliance requirements

Understanding, identifying and designating business criticality is the first major step. The next management consideration topic is to ensure your operational compliance requirements, especially security compliances, are aligned. Performing the same methodologies for business criticality to your security compliance considerations will:

- Reveal the complexities of your business systems and their compliance requirements, such as sovereignty, industry, or privacy.
- Map any interdependencies with your development, operations, and security teams, ensuring that key point of contacts and subject matter experts are identified and well known across your organization.
- Ensure that your compliance adheres to any compliance audit and reporting requirements.

Apply the operational compliance content to ensure that your business requirements, criticality, and compliance

requirements are aligned.

Next, review and use the overview of the Azure Security Benchmark content to ensure the various control domains are identified and mapped to your operational-compliance.

Lastly, ensure the incorporation of the governance and security content in Security control v2: governance and strategy is documented within your overall business operational methodologies.

Use the template

The following steps apply if you're using the operations management workbook to plan for cloud management.

1. Record the criticality scale in the `Scale` worksheet.
2. Update each workload in either the `Example` worksheet or the `Clean Template` worksheet to reflect the default criticality in the `Criticality` column.
3. The business should enter the correct values to reflect any deviations from the default criticality.

Next steps

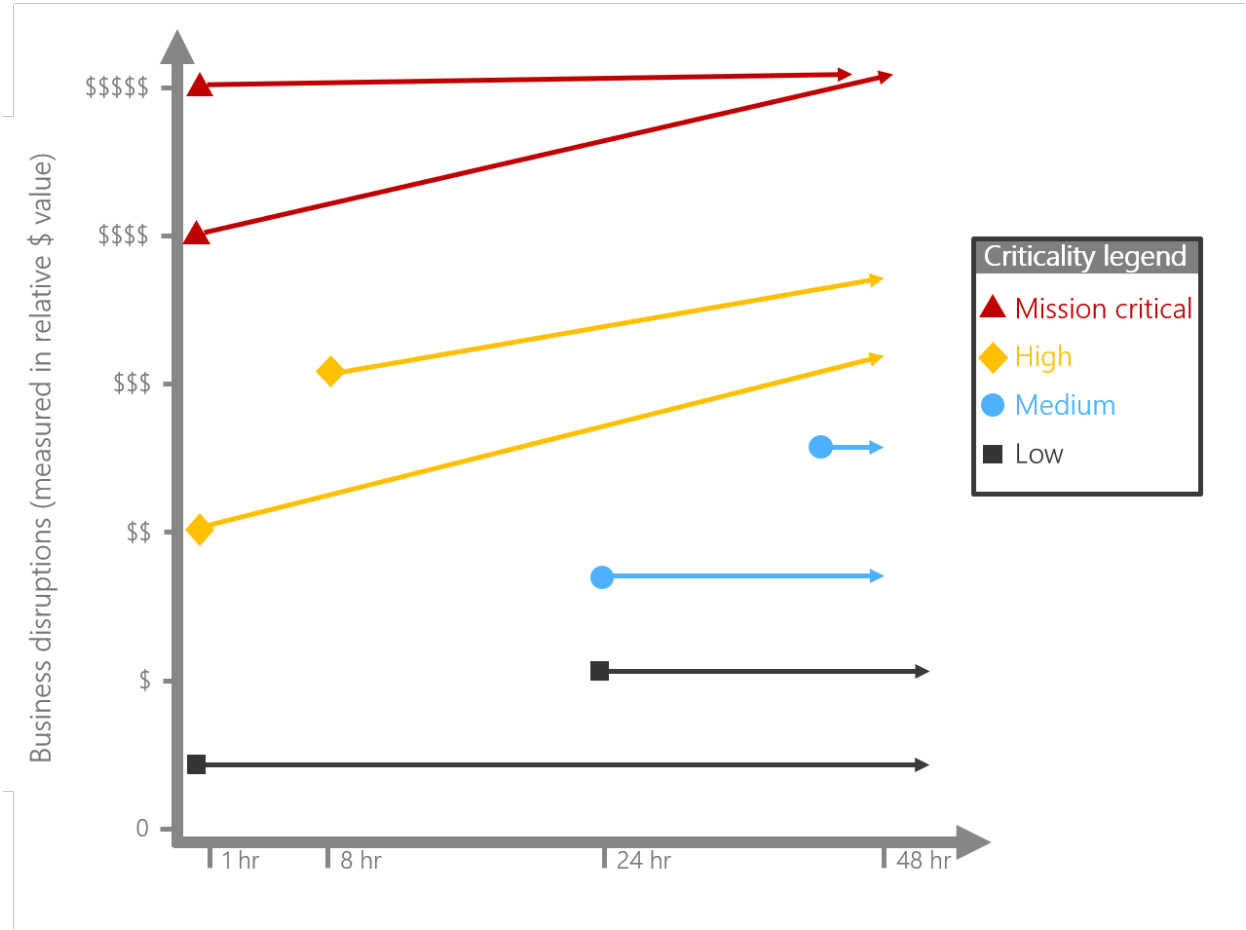
After your team has defined business criticality, you can calculate and record business impact.

Calculate and record business impact

Business impact in cloud management

5/4/2021 • 4 minutes to read • Edit Online

Assume the best, prepare for the worst. In IT management, it's safe to assume that the workloads required to support business operations will be available and will perform within agreed-upon constraints, based on the selected criticality. However, to manage investments wisely, it's important to understand the impact on the business when an outage or performance degradation occurs. This importance is illustrated in the following graph, which maps potential business interruptions of specific workloads to the business impact of outages across a relative value scale.



To create a fair basis of comparison for the impact on various workloads across a portfolio, a time/value metric is suggested. The time/value metric captures the adverse impact of a workload outage. Generally, this impact is recorded as a direct loss of revenue or operating revenue during a typical outage period. More specifically, it calculates the amount of lost revenue for a unit of time. The most common time/value metric is *Impact per hour*, which measures operating revenue losses per hour of outage.

A few approaches can be used to calculate impact. You can apply any of the options in the following sections to achieve similar outcomes. It's important to use the same approach for each workload when you calculate protected losses across a portfolio.

Start with estimates

Current operating models might make it difficult to determine an accurate impact. Fortunately, few systems need a highly accurate loss calculation. In the previous step, *Classify Criticality*, we suggested that you start all workloads with a default of *medium criticality*. Medium criticality workloads generally receive a standard level of

management support with a relatively low impact on operating cost. Only when a workload requires additional operational management resources might you require an accurate financial impact.

For all standardized workloads, business impact serves as a prioritization variable when you're recovering systems during an outage. Outside of those limited situations, the business impact creates little to no change in the operations management experience.

Calculate time

Depending on the nature of the workload, you could calculate losses differently. For high-paced transactional systems such as a real-time trading platform, losses per millisecond might be significant. Less frequently used systems, such as payroll, might not be used every hour. Whether the frequency of usage is high or low, it's important to normalize the time variable when you calculate financial impact.

Calculate total impact

When you want to consider additional management investments, it's more important that the business impact be more accurate. The following three approaches to calculating losses are ordered from most accurate to least accurate:

- **Adjusted losses:** If your business has experienced a major loss event in the past, such as a hurricane or other natural disaster, a claims adjuster might have calculated actual losses during the outage. These calculations are based on insurance industry standards for loss calculation and risk management. Using adjusted losses as the total amount of losses in a specific time frame can lead to highly accurate projections.
- **Historical losses:** If your on-premises environment has suffered historically from outages resulting from infrastructure instability, it can be a bit harder to calculate losses. But you can still apply the adjuster formulas used internally. To calculate historical losses, compare the deltas in sales, gross revenue, and operating costs across three time frames: before, during, and after outage. By examining these deltas, you can identify accurate losses when no other data is available.
- **Complete loss calculation:** If no historical data is available, you can derive a comparative loss value. In this model, you determine the average gross revenue per hour for the business unit. When you're projecting loss avoidance investments, it's not fair to assume that a complete system outage equates to a 100 percent loss of revenue. But you can use this assumption as a rough basis for comparing loss impacts and prioritizing investments.

Before you make certain assumptions about potential losses associated with workload outages, it's a good idea to work with your finance department to determine the best approach to such calculations.

Calculate workload impact

When you're calculating losses by applying historical data, you might have enough information to clearly determine the contribution of each workload to those losses. Performing this evaluation is where partnerships within the business are absolutely critical. After the total impact has been calculated, that impact must be attributed across each of the workloads. That distribution of impact should come from the business stakeholders, who should agree on the relative and cumulative impact of each workload. To that end, your team should solicit feedback from business executives to validate alignment. Such feedback is often equal parts emotion and subject matter expertise. It's important that this exercise represent the logic and beliefs of the business stakeholders who should have a say in budget allocation.

Use the template

If you're using the operations management workbook to plan for cloud management, consider doing the

following:

- Each business should update each workload in either the `Example` worksheet or the `Clean Template` worksheet, along with the `Time/Value Impact` of each workload. By default, `Time/Value Impact` represents the projected losses per hour associated with an outage of the workload.

Next steps

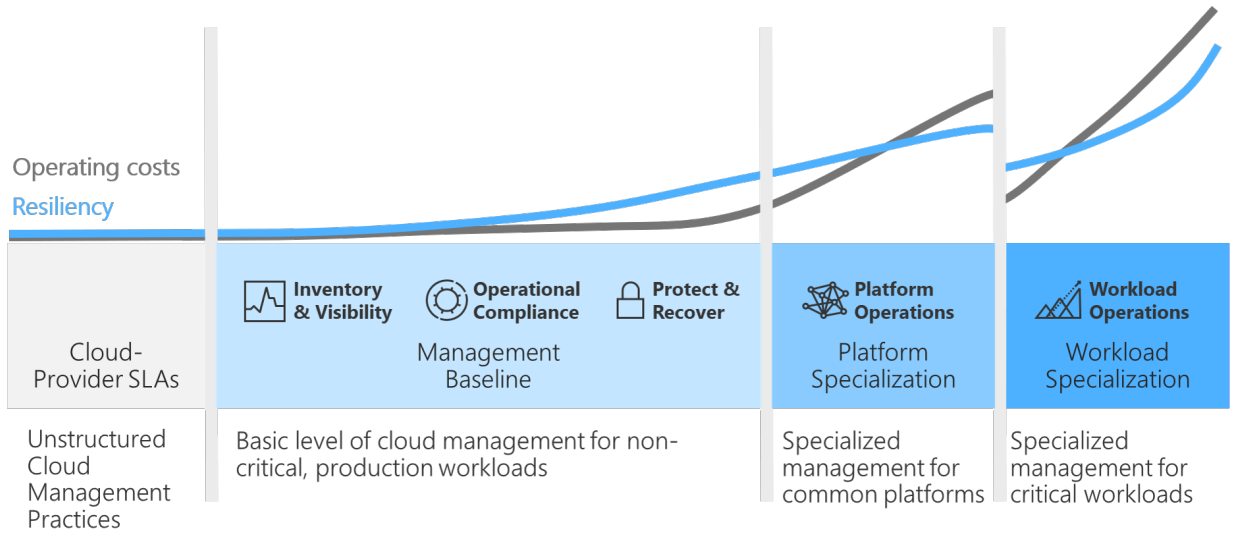
After the business has defined impact, you can align commitments.

Align management commitments with the business

Business commitment in cloud management

5/4/2021 • 9 minutes to read • Edit Online

Defining *business commitment* is an exercise in balancing priorities. The objective is to align the proper level of operational management at an acceptable operating cost. Finding that balance requires a few data points and calculations, which we've outlined in this article.



Commitments to business stability, via technical resiliency or other service-level agreement (SLA) impacts, are a business justification decision. For most workloads in an environment, a baseline level of cloud management is sufficient. For others, a 2x to 4x cost increase is easily justified because of the potential impact of any business interruptions.

The previous articles in this series can help you understand the classification and impact of interruptions to various workloads. This article helps you calculate the returns. As illustrated in the preceding image, each level of cloud management has inflection points where cost can rise faster than increases in resiliency. Those inflection points will prompt detailed business decisions and business commitments.

Determine a proper commitment with the business

For each workload in the portfolio, the cloud operations team and cloud strategy team should align on the level of management that's provided directly by the cloud operations team.

As you're establishing a commitment with the business, there are a few key aspects to align:

- IT operations prerequisites.
- Management responsibility.
- Cloud tenancy.
- Soft-cost factors.
- Loss avoidance ROI.
- Validation of management level.

To aid in your decision process, the remainder of this article describes each of these aspects in greater detail.

IT operations prerequisites

The Azure Management Guide outlines the management tools that are available in Azure. Before reaching a

commitment with the business, IT should determine an acceptable standard-level management baseline to be applied to all managed workloads. IT would then calculate a standard management cost for each of the managed workloads in the IT portfolio, based on counts of CPU cores, disk space, and other asset-related variables. IT would also estimate a composite SLA for each workload, based on the architecture.

> **TIP**
> IT operations teams often use a default minimum of 99.9 percent uptime for the initial composite SLA. They might also choose to normalize management costs based on the average workload, especially for solutions with minimal logging and storage needs. Averaging the costs of a few medium criticality workloads can provide a starting point for initial conversations.

> **TIP**
> If you're using the operations management workbook to plan for cloud management, the operations management fields should be updated to reflect these prerequisites. Those fields include *Commitment level*, *Composite SLA*, and *Monthly cost*. Monthly cost should represent the cost of the added operational management tools on a monthly basis.

The operations management baseline serves as an initial starting point to be validated in each of the following sections.

Management responsibility

In a traditional on-premises environment, the cost of managing the environment is commonly assumed to be a sunk cost that's owned by IT operations. In the cloud, management is a purposeful decision with direct budgetary impact. The costs of each management function can be more directly attributed to each workload that's deployed to the cloud. This approach allows for greater control, but it does create a requirement for cloud operations teams and cloud strategy teams to first commit to an agreement about responsibilities.

Organizations might also choose to outsource some of their ongoing management functions to a service provider. These service providers can use Azure Lighthouse to give organizations more precise control in granting access to their resources, along with greater visibility into the actions performed by the service providers.

- **Delegated responsibility:** Because there's no need to centralize and assume operational management overhead, IT operations for many organizations are considering new approaches. One common approach is referred to as *delegated responsibility*. In a cloud center of excellence model, platform operations and platform automation provide self-service management tools that can be used by business-led operations teams, independent of a centralized IT operations team. This approach gives business stakeholders complete control over management-related budgets. It also allows the cloud center of excellence (CCoE) team to ensure that a minimum set of guardrails has been properly implemented. In this model, IT acts as a broker and a guide to help the business make wise decisions. Business operations oversee day to day operations of dependent workloads.
- **Centralized responsibility:** Compliance requirements, technical complexity, and some shared service models might require a *Central IT team* model. In this model, IT continues to exercise its operations management responsibilities. Environmental design, management controls, and governance tooling might be centrally managed and controlled, which restricts the role of business stakeholders in making management commitments. But the visibility into the cost and architecture of cloud approaches makes it much easier for centralized IT to communicate the cost and level of management for each workload.
- **Mixed model:** Classification is at the heart of a *mixed model* of management responsibilities. Companies that are in the midst of a transformation from on-premises to cloud might require an on-premises-first operating model for a while. Companies with strict compliance requirements, or that

depend on long-term contracts with IT outsourcing vendors, might require a centralized operating model.

Regardless of their constraints, today's businesses must innovate. When rapid innovation must flourish, in the midst of a central-IT, centralized-responsibility model, a mixed-model approach might provide balance. In this approach, a central IT team provides a centralized operating model for all workloads that are mission-critical or contain sensitive information. At the same time, all other workload classifications might be placed in a cloud environment that's designed for delegated responsibilities. The centralized responsibility approach serves as the general operating model. The business then has flexibility to adopt a specialized operating model, based on its required level of support and sensitivity.

The first step is committing to a responsibility approach, which then shapes the following commitments.

Which organization will be responsible for day-to-day operations management for this workload?

Cloud tenancy

For most businesses, management is easier when all assets reside in a single tenant. However, some organizations might need to maintain multiple tenants. To learn why a business might require a multitenant Azure environment, see Centralize management operations with Azure Lighthouse.

Will this workload reside in a single Azure tenant, alongside all other workloads?

Soft-cost factors

The next section outlines an approach to comparative returns that are associated with levels of management processes and tooling. At the end of that section, each analyzed workload measures the cost of management relative to the forecast impact of business disruptions. That approach provides a relatively easy way to understand whether an investment in richer management approaches is warranted.

Before you run the numbers, it's important to look at the soft-cost factors. Soft-cost factors produce a return, but that return is difficult to measure through direct hard-cost savings that would be visible in a profit-and-loss statement. Soft-cost factors are important because they can indicate a need to invest in a higher level of management than is fiscally prudent.

A few examples of soft-cost factors would include:

- Daily workload usage by the board or CEO.
- Workload usage by the top *x%* of customers that leads to a greater revenue impact elsewhere.
- Impact on employee satisfaction.

The next data point that's required to make a commitment is a list of soft-cost factors. These factors don't need to be documented at this stage, but business stakeholders should be aware of the importance of these factors and their exclusion from the following calculations.

Calculate loss avoidance ROI

When it's calculating the relative return on operations management costs, the IT team that's responsible for cloud operations should complete the previously mentioned prerequisites and assume a minimum level of management for all workloads.

The next commitment to be made is an acceptance by the business of the costs associated with the baseline-managed offering.

Does the business agree to invest in the baseline offering to meet minimum standards of cloud operations?

If the business does not agree to that level of management, a solution must be devised that allows the business

to proceed, without materially affecting the cloud operations of other workloads.

If the business wants more than the standard management level, the remainder of this section will help validate that investment and the associated returns (in the form of loss avoidance).

Increased levels of management: Design principles and service catalog

For managed solutions, several design principles and template solutions can be applied in addition to the management baseline. Each of the design principles for reliability and resiliency adds operating cost to the workload. For IT and the business to agree on these additional commitments, it's important to understand potential losses that can be avoided through that increased investment.

The following calculations will walk through formulas to help you better understand the differences between losses and increased management investments. For guidance on calculating the cost of increased management, see Workload automation and Platform automation.

> **TIP**
> If you're using the operations management workbook to plan for cloud management, update the operations management fields to reflect to reflect each conversation. Those fields include *Commitment level*, *Composite SLA*, and *Monthly cost*. Monthly cost should represent the monthly cost of the added operational management tools. After they're updated, the fields will update the ROI formulas and each of the following fields.

Estimate outage (hours per year)

Composite SLA is the service-level agreement that's based on the deployment of each asset in the workload. That field drives *estimated outage* (labeled `Est.Outage` in the workbook). To calculate estimated outage in hours per year without using the workbook, apply the following formula:

> *Estimated outage = (1 - composite SLA percentage) × number of hours in a year*

The workbook uses the default value of *8,760 hours per year*.

Standard loss impact

Standard loss impact (labeled `Standard Impact` in the workbook) forecasts the financial impact of any outage, assuming that the *estimated outage* prediction proves accurate. To calculate this forecast without using the workbook, apply the following formula:

> *Standard impact = estimated outage @ three 9s of uptime × time-value impact*

This serves as a baseline for cost, should the business stakeholders choose to invest in a higher level of management.

Composite-SLA impact

Composite-SLA impact (labeled `Commitment level impact` in the workbook) provides updated fiscal impact, based on the changes to the uptime SLA. This calculation allows you to compare the projected financial impact of both options. To calculate this forecast impact without the spreadsheet, apply the following formula:

> *Composite-SLA impact = estimated outage × time-value impact*

The value represents the potential losses to be avoided by the changed commitment level and new composite SLA.

Comparison basis

Comparison basis evaluates standard impact and composite SLA impact to determine which is most appropriate in the return column.

Return on loss avoidance

If the cost of managing a workload exceeds the potential losses, the proposed investment in cloud management might not be fruitful. To compare the *Return on Loss Avoidance*, see the column labeled *Annual ROI*****. To calculate this column on your own, use the following formula:

> *Return on loss avoidance = (comparison basis - (monthly cost × 12)) ÷ (monthly cost × 12))*

Unless there are other soft-cost factors to consider, this comparison can quickly suggest whether there should be a deeper investment in cloud operations, resiliency, reliability, or other areas.

Validate the commitment

By this point in the process, commitments have been made: centralized or delegated responsibility, Azure tenancy, and level of commitment. Each commitment should be validated and documented to ensure that the cloud operations team, the cloud strategy team, and the business stakeholders are aligned on this commitment to manage the workload.

Next steps

After the commitments are made, the responsible operations teams can begin configuring the workload in question. To get started, evaluate various approaches to inventory and visibility.

Inventory and visibility options

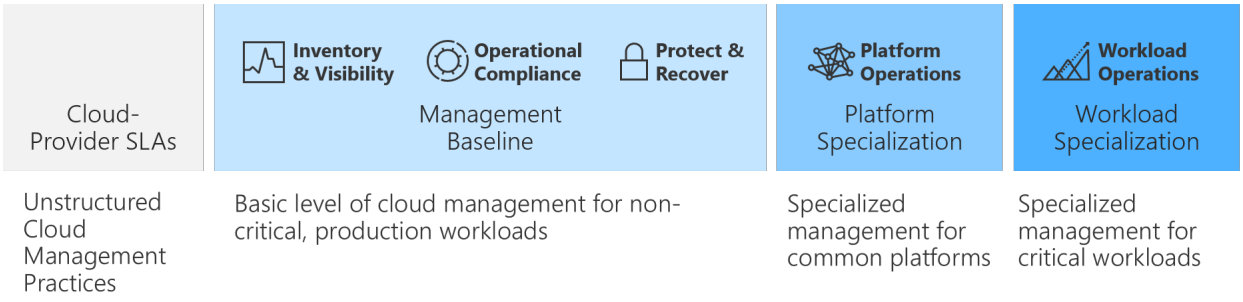
Cloud management disciplines

5/4/2021 • 3 minutes to read • Edit Online

The keys to proper management in any environment are consistency and repeatable processes. There are endless of options for the things that can be done in Azure. Likewise, there are countless approaches to cloud management. To provide consistency and repeatability, it's important to narrow those options to a consistent set of management processes and tools that will be offered for workloads hosted in the cloud.

Suggested management levels

Because the workloads in your IT portfolio vary, it's unlikely that a single level of management will suffice for each workload. To help you support a variety of workloads and business commitments, we suggest that your cloud operations team or platform operations team establish a few levels of operations management.



As a starting point, consider establishing the management levels that are shown in the preceding diagram and suggested in the following list:

- **Management baseline:** A cloud management baseline (or management baseline) is a defined set of tools, processes, and consistent pricing that serve as the foundation for all cloud management in Azure. To establish a cloud management baseline and determine which tools to include in the baseline offering to your business, review the list in the "cloud management disciplines" section.
- **Enhanced baseline:** Some workloads might require enhancements to the baseline that aren't necessarily specific to a single platform or workload. Although these enhancements aren't cost effective for every workload, there should be common processes, tools, and solutions for any workload that can justify the cost of the extra management support.
- **Platform specialization:** In any given environment, some common platforms are used by a variety of workloads. This general architectural commonality doesn't change when businesses adopt the cloud. Platform specialization is an elevated level of management that applies data and architectural subject matter expertise to provide a higher level of operational management. Examples of platform specialization would include management functions specific to SQL Server, containers, Active Directory, or other services that can be better managed through consistent, repeatable processes, tools, and architectures.
- **Workload specialization:** For workloads that are truly mission critical, there might be a cost justification to go much deeper into the management of that workload. Workload specialization applies workload telemetry to determine more advanced approaches to daily management. That same data often identifies automation, deployment, and design improvements that would lead to greater stability, reliability, and resiliency beyond what's possible with operational management alone.

Organizations might also choose to outsource functions related to one or more of these management levels to a service provider. These service providers can use Azure Lighthouse to provide greater precision and transparency.

The remaining articles in this series outline processes that are commonly found within each of these disciplines. In parallel, the Azure Management Guide demonstrates the tools that can support each of those processes. For assistance with building your management baseline, start with the Azure Management Guide. After you've established the baseline, this article series and the accompanying best practices can help expand that baseline to define other levels of management support.

Operations management discipline

Each suggested management level provides necessary operations management for all assets (apps, data, & infra) in the portfolio, with an increasing degree of specificity. This mapping is designed to make it easier for the relevant roles to find the most appropriate processes and tools to deliver on the required level of cloud management.

The operations management processes defines the iterative discipline required to deliver operations and meet business commitments. This iterative process delivers on three levels of operations on a recurring rhythm of business:

- **Operations baseline (or enhanced baseline):** Consistent operations management of all deployed assets. Centralized technology teams focus on portfolio health and implements broad sweeping changes.
- **Platform operations:** Elevated operations for mission critical technology platforms. Centralized teams focus on operational fitness of shared platforms.
- **Workload operations:** Workload operations for refined changes to defined workloads. Workload specific teams deliver architecture change based on the pillars of Azure Well-Architected Framework to improve operations through more granular improvements.

The operations management processes unite all three elevations of operations management in a holistic solution to improve operational fitness of all deployed assets regardless of the chosen operating model.

Next steps

The next step toward defining each level of cloud management is an understanding of inventory and visibility.

Inventory and visibility options

Inventory and visibility in cloud management

5/4/2021 • 6 minutes to read • Edit Online

Operational management has a clear dependency on data. Consistent management requires an understanding about what is managed (inventory) and how those managed workloads and assets change over time (visibility). Clear insights about inventory and visibility help empower the team to manage the environment effectively. All other operational management activities and processes build on these two areas.

A few classic phrases about the importance of measurements set the tone for this article:

- Manage what matters.
- You can only manage what you can measure.
- If you can't measure it, it might not matter.

The inventory and visibility discipline builds on these timeless phrases. Before you can effectively establish operational management processes, it's important to gather data and create the right level of visibility for the right teams.

Common customer challenges

Unless inventory and visibility processes are consistently applied, operational management teams can suffer from a higher volume of business interruptions, longer time to recovery, and greater amounts of effort required to troubleshoot and triage issues. As changes adversely affect higher priority applications and larger numbers of assets, each of these metrics grows even faster.

These challenges stem from a small number of questions that can be answered only through consistent data/telemetry:

- How does the current-state performance deviate from standard operational performance telemetry?
- What assets are causing the business interruptions at the workload level?
- Which assets must be remediated to return to acceptable performance of this workload or business process?
- When did the deviation start? What was the trigger?
- Which changes have been made to the underlying assets? By whom?
- Were the changes intentional? Malicious?
- How did changes affect performance telemetry?

It's difficult, if not impossible, to answer these questions without a rich, centralized source for logs and telemetry data. To enable cloud management by ensuring the consistent configuration that's required to centralize the data, the baseline service must first start by defining the processes. The processes should capture how such a configuration enforces data collection to support the components of inventory and visibility in the next section.

Components of inventory and visibility

Creating visibility on any cloud platform requires a few key components:

- Responsibility and visibility
- Inventory
- Central logging
- Change tracking
- Performance telemetry

Responsibility and visibility

When you establish commitments for each workload, management responsibility is a key factor. Delegated responsibility creates a need for delegated visibility. The first step toward inventory and visibility is to ensure that the responsible parties have access to the right data. Before you implement any cloud-native tools for visibility, ensure that each monitoring tool has been configured with proper access and scope for each operations team.

Inventory

If no one knows that an asset exists, it's difficult to manage the asset. Before an asset or workload can be managed, it must be inventoried and classified. The first technical step toward stable operations is a validation of inventory and classification of that inventory.

Central logging

Centralized logging is critical to the visibility that's required day to day by the operations management teams. All assets deployed to the cloud should record logs to a central location. In Azure, that central location is Log Analytics. The centralization of logging drives reports about change management, service health, configuration, and most other aspects of IT operations.

Enforcing the consistent use of central logging is the first step toward establishing repeatable operations. Enforcement can be accomplished through corporate policy. When possible, however, enforcement should be automated to ensure consistency.

Change tracking

Change is the one constant in a technology environment. Awareness and understanding of changes across multiple workloads is essential to reliable operations. Any cloud management solution should include a means of understanding the when, how, and why of technical change. Without those data points, remediation efforts are significantly hindered.

Performance telemetry

Business commitments about cloud management are driven by data. To properly maintain commitments, the cloud operations team must first understand the telemetry about the stability, performance, and operations of the workload, and the assets that support the workload.

The ongoing health and operations of the network, DNS, operating systems, and other foundational aspects of the environment are critical data points that factor into the overall health of any workload.

Processes

Perhaps more important than the features of the cloud management platform, the cloud management processes will realize operations commitments with the business. Any cloud management methodology should include, at a minimum, the following processes:

- **Reactive monitoring:** When deviations adversely affect business operations, who addresses those deviations? What actions do they take to remediate the deviations?
- **Proactive monitoring:** When deviations are detected but business operations are not affected, how are those deviations addressed, and by whom?
- **Commitment reporting:** How is adherence to the business commitment communicated to business stakeholders?
- **Budgetary reviews:** What is the process for reviewing those commitments against budgeted costs? What is the process for adjusting the deployed solution or the commitments to create alignment?
- **Escalation paths:** What escalation paths are available when any of the preceding processes fail to meet the needs of the business?

There are several more processes related to inventory and visibility. The preceding list is designed to provoke

thought within the operations team. Answering these questions will help develop some of the necessary processes, as well as likely trigger new, deeper questions.

Responsibilities

When you're developing processes for operational monitoring, it's equally important to determine responsibilities for daily operation and regular support of each process.

In a centralized IT organization, IT provides the operational expertise. The business would be consultative in nature, when issues require remediation.

In a cloud center of excellence organization, business operations would provide the expertise and hold responsibility for management of these processes. IT would focus on the automation and support of teams, as they operate the environment.

But these are the common responsibilities. Organizations often require a mixture of responsibilities to meet business commitments.

Act on inventory and visibility

Regardless of the cloud platform, the five components of inventory and visibility are used to drive most operational processes. All subsequent disciplines will build on the data that's being captured. The next articles in this series outline ways to act on that data and integrate other data sources.

Share visibility

Data without action produces little return. Cloud management might expand beyond cloud-native tools and processes. To accommodate broader processes, a cloud management baseline might need to be enhanced to include reporting, IT service management integration, or data centralization. Cloud management might need to include one or more of the following during various phases of operational maturity.

Report

Offline processes and communication about commitments to business stakeholders often require reporting. Self-service reporting or periodic reporting might be a necessary component of an enhanced management baseline.

IT service management (ITSM) integration

ITSM integration is often the first example of acting on inventory and visibility. When deviations from expected performance patterns arise, ITSM integration uses alerts from the cloud platform to trigger tickets in a separate IT service management tool to trigger remediation activities. Some operating models might require ITSM integration as an aspect of the enhanced management baseline.

Data centralization

There's a variety of reasons why a business might require multiple tenants within a single cloud provider. In those scenarios, data centralization is a required component of the enhanced management baseline, because it can provide visibility across each of those tenants or environments.

Next steps

Operational compliance builds on inventory capabilities by applying management automation and controls. See how operational compliance maps to your processes.

Plan for operational compliance

Operational compliance in cloud management

5/4/2021 • 2 minutes to read • Edit Online

Operational compliance builds on the discipline of inventory and visibility. As the first actionable step of cloud management, this discipline focuses on regular telemetry reviews and remediation efforts (both proactive and reactive remediation). This discipline is the cornerstone for maintaining balance between security, governance, performance, and cost.

Components of operations compliance

Maintaining compliance with operational commitments requires analysis, automation, and human remediation. Effective operational compliance requires consistency in a few critical processes:

- Resource consistency
- Environment consistency
- Resource configuration consistency
- Update consistency
- Remediation automation

Resource consistency

The most effective step that a cloud management team can take toward operational compliance is to establish consistency in resource organization and tagging. When resources are consistently organized and tagged, all other operational tasks become easier. For deeper guidance on resource consistency, see the Govern methodology. Specifically, review the initial governance foundation articles to learn how to start developing resource consistency.

Environment consistency

Establishing consistent environments, or landing zones, is the next most important step toward operational compliance. When landing zones are consistent and enforced through automated tools, it's significantly less complex to diagnose and resolve operational issues. For deeper guidance on environment consistency, see the readiness phase of the cloud adoption lifecycle. The exercises in this phase help build a repeatable process for defining and maturing a consistent, code-first approach to developing cloud-based environments.

Resource configuration consistency

As it builds on governance and readiness approaches, cloud management should include processes for the ongoing monitoring and evaluation of its adherence to resource consistency requirements. As workloads change or new versions are adopted, it's vital that cloud management processes evaluate any configuration changes, which are not easily regulated through automation.

When inconsistencies are discovered, some are addressed by consistency in updates and others can be automatically remediated.

Update consistency

Stability in approach can lead to more stable operations. But some changes are required within cloud management processes. In particular, regular patching and performance changes are essential to reducing interruptions and controlling costs.

One of the many values of a mature cloud management methodology is a focus on stabilizing and controlling necessary change.

Any cloud management baseline should include a means of scheduling, controlling, and possibly automating

necessary updates. Those updates should include patches at a minimum, but could also include performance, sizing, and other aspects of updating assets.

Remediation automation

As an enhanced baseline for cloud management, some workloads may benefit from automated remediation. When a workload commonly encounters issues that can't be resolved through code or architectural changes, automating remediation can help reduce the burden of cloud management and increase user satisfaction.

Many would argue that any issue that's common enough to automate should be resolved through resolution of technical debt. When a long-term resolution is prudent, it should be the default option. However, some business scenarios make it difficult to justify large investments in the resolution of technical debt. When such a resolution can't be justified, but remediation is a common and costly burden, automated remediation is the next best solution.

Next steps

Protection and recovery are the next areas to consider in a cloud management baseline.

Protect and recover

Protect and recover in cloud management

5/24/2021 • 5 minutes to read • Edit Online

After they've met the requirements for inventory and visibility and operational compliance, cloud management teams can anticipate and prepare for a potential workload outage. As they're planning for cloud management, the teams must start with an assumption that something will fail.

No technical solution can consistently offer a 100 percent uptime SLA. Solutions with the most redundant architectures claim to deliver on "six 9s" or 99.9999 percent uptime. But even a "six 9s" solution goes down for 31.6 seconds in any given year. Sadly, it's rare for a solution to warrant a large, ongoing operational investment that's required to reach "six 9s" of uptime.

Preparation for an outage allows the team to detect failures sooner and recover more quickly. The focus of this discipline is on the steps that come immediately after a system fails. How do you protect workloads, so that they can be recovered quickly when an outage occurs?

Translate protection and recovery conversations

The workloads that power business operations consist of applications, data, virtual machines (VMs), and other assets. Each of those assets might require a different approach to protection and recovery. The important aspect of this discipline is to establish a consistent commitment within the management baseline, which can provide a starting point during business discussions.

At a minimum, each asset that supports any given workload should have a baseline approach with a clear commitment to speed of recovery (recovery time objectives, or RTO) and risk of data loss (recovery point objectives, or RPO).

Recovery time objectives (RTO)

When disaster strikes, a recovery time objective is the amount of time it should take to recovery any system to its state prior to the disaster. For each workload, that would include the time required to restore minimum necessary functionality for the VMs and applications. It also includes the amount of time required to restore the data that's required by the applications.

In business terms, RTO represents the amount of time that the business process will be out of service. For mission-critical workloads, this variable should be relatively low, allowing the business processes to resume quickly. For lower-priority workloads, a standard level of RTO might not have a noticeable impact on company performance.

The management baseline should establish a standard RTO for non-mission-critical workloads. The business can then use that baseline as a way to justify additional investments in recovery times.

Recovery point objectives (RPO)

In most cloud management systems, data is periodically captured and stored through some form of data protection. The last time data was captured is referred to as a recovery point. When a system fails, it can be restored only to the most recent recovery point.

If a system has a recovery point objective that's measured in hours or days, a system failure would result in the loss of data for those hours or days between the last recovery point and the outage. A one-day RPO would theoretically result in the loss of all transactions in the day leading up to the failure.

For mission-critical systems, an RPO that's measured in minutes or seconds might be more appropriate to use to avoid a loss in revenue. But a shorter RPO generally results in an increase in overall management costs.

To help minimize costs, a management baseline should focus on the longest acceptable RPO. The cloud management team can then increase the RPO of specific platforms or workloads, which would warrant more investment.

Protect and recover workloads

Most of the workloads in an IT environment support a specific business or technical process. Systems that don't have a systemic impact on business operations often don't warrant the increased investments required to recover quickly or minimize data loss. By establishing a baseline, the business can clearly understand what level of recovery support can be offered at a consistent, manageable price point. This understanding helps the business stakeholders evaluate the value of an increased investment in recovery.

For most cloud management teams, an enhanced baseline with specific RPO/RTO commitments for various assets yields the most favorable path to mutual business commitments. The following sections outline a few common enhanced baselines that empower the business to easily add protection and recovery functionality through a repeatable process.

Protect and recover data

Data is arguably the most valuable asset in the digital economy. The ability to protect and recover data more effectively is the most common enhanced baseline. For the data that powers a production workload, loss of data can be directly equated to loss in revenue or loss of profitability. We generally encourage cloud management teams to offer a level of enhanced management baseline that supports common data platforms.

Before cloud management teams implement platform operations, it's common for them to support improved operations for a platform as a service (PaaS) data platform. For instance, it's easy for a cloud management team to enforce a higher frequency of backup or multiregion replication for Azure SQL Database or Azure Cosmos DB solutions. Doing so allows the development team to easily improve RPO by modernizing their data platforms.

To learn more about this thought process, see Platform operations discipline.

Protect and recover VMs

Most workloads have some dependency on virtual machines, which host various aspects of the solution. For the workload to support a business process after a system failure, some virtual machines must be recovered quickly.

Every minute of downtime on those virtual machines could cause lost revenue or reduced profitability. When VM downtime has a direct impact on the fiscal performance of the business, RTO is very important. Virtual machines can be recovered more quickly by using replication to a secondary site and automated recovery, a model that's referred to as a hot-warm recovery model. At the highest state of recovery, virtual machines can be replicated to a fully functional, secondary site. This more expensive approach is referred to as a high-availability, or hot-hot, recovery model.

Each of the preceding models reduces the RTO, resulting in a faster restoration of business process capabilities. However, each model also results in significantly increased cloud management costs.

Also note that, apart from replication for high-availability, backup should be enabled for scenarios such as accidental delete, data corruption and ransomware attacks.

For more information about this thought process, see Workload operations discipline.

Next steps

After this management baseline component is met, the team can look ahead to avoid outages in its platform operations and workload operations.

Platform operations Workload operations

Platform operations in cloud management

5/4/2021 • 6 minutes to read • Edit Online

A cloud management baseline that spans inventory and visibility, operational compliance, and protection and recovery might provide a sufficient level of cloud management for most workloads in the IT portfolio. However, that baseline is seldom enough to support the full portfolio. This article builds on the most common next step in cloud management, portfolio operations.

A quick study of the assets in the IT portfolio highlights patterns across the workloads that are being supported. Within those workloads, there will be common platforms. Depending on the past technical decisions within the company, those platforms could vary widely.

For some organizations, there will be a heavy dependence on SQL Server, Oracle, or other open-source data platforms. In other organizations, the commonalities might be rooted in the hosting platforms for virtual machines (VMs) or containers. Still others might have a common dependency on applications or enterprise resource planning (ERP) systems such as SAP or Oracle.

By understanding these commonalities, the cloud management team can specialize in higher levels of support for those prioritized platforms.

Establish a service catalog

The objective of platform operations is to create reliable and repeatable solutions, which the cloud adoption team can use to deliver a platform that provides a higher level of business commitment. That commitment could decrease the likelihood or frequency of downtime, which improves reliability. In the event of a system failure, the commitment could also help decrease the amount of data loss or time to recovery. Such a commitment often includes ongoing, centralized operations to support the platform.

As the cloud management team establishes higher degrees of operational management and specialization related to specific platforms, those platforms are added to a growing service catalog. The service catalog provides self-service deployment of platforms in a specific configuration, which adheres to ongoing platform operations. During the business-alignment conversation, cloud management and cloud strategy teams can propose service catalog solutions as a way for the business to improve reliability, uptime, and recovery commitments in a controlled, repeatable process.

For reference, some organizations refer to an early-stage service catalog as an *approved list*. The primary difference is that a service catalog comes with ongoing operational commitments from the cloud center of excellence (CCoE). An approved list is similar, in that it provides a preapproved list of solutions that a team can use in the cloud. However, typically there isn't an operational benefit associated with applications on an approved list.

Much like the debate between centralized IT and CCoE, the difference is one of priorities. A service catalog assumes good intent but provides operational, governance, and security guardrails that accelerate innovation. An approved list hinders innovation until operations, compliance, and security gates can be passed for a solution. Both solutions are viable, but they require the company to make subtle prioritization decisions to invest more in innovation or compliance.

Build the service catalog

Cloud management is seldom successful at delivering a service catalog in a silo. Proper development of the catalog requires a partnership across the central IT team or the CCoE. This approach tends to be most successful when an IT organization reaches a CCoE level of maturity, but could be implemented sooner.

When it's building the service catalog within a CCoE model, the cloud platform team builds out the desired-state platform. The cloud governance and cloud security teams validate governance and compliance within the deployment. The cloud management team establishes ongoing operations for that platform. And the cloud automation team packages the platform for scalable, repeatable deployment.

After the platform is packaged, the cloud management team can add it to the growing service catalog. From there, the cloud adoption team can use the package or others in the catalog during deployment. After the solution goes to production, the business realizes the extra benefits of improved operational management and potentially reduced business disruptions.

> **NOTE**
> Building a service catalog requires a great deal of effort and time from multiple teams. Using the service catalog or approved list as a gating mechanism will slow innovation. When innovation is a priority, service catalogs should be developed parallel to other adoption efforts.

Define your own platform operations

Although management tools and processes can help improve platform operations, that is often not enough to achieve the desired states of stability and reliability. True platform operations requires a focus on pillars of architecture excellence. When a platform justifies a deeper investment in operations, consider the following five pillars before the platform becomes a part of any service catalog:

- **Cost optimization:** Manage costs to maximize the value delivered.
- **Operational excellence:** Follow operational processes that keep a system running in production.
- **Performance efficiency:** Scale systems to adapt to changes in load.
- **Reliability:** Design systems to recover from failures and continue to function.
- **Security:** Protect applications and data from threats.

The Microsoft Azure Well-Architected Framework provides an approach to evaluating specific workloads for adherence to these pillars, in an effort to improve overall operations. These pillars can be applied to both platform operations and workload operations.

Get started with specific platforms

The platforms discussed in the next sections are common to typical Azure customers, and they can easily justify an investment in platform operations. Cloud management teams tend to start with them when they're building out platform operations requirements or a full service catalog.

PaaS data operations

Data is often the first platform to warrant platform operations investments. When data is hosted in a platform as a service (PaaS) environment, business stakeholders tend to request a reduced recovery point objective (RPO) to minimize data loss. Depending on the nature of the application, they might also request a reduction in recovery time objective (RTO). In either case, the architecture that supports PaaS-based data solutions can easily accommodate some increased level of management support.

In most scenarios, the cost of improving management commitments is easily justified, even for applications that are not mission critical. This platform operations improvement is so common that many cloud management teams see it more as an enhanced baseline, rather than as a true platform operations improvement.

IaaS data operations

When data is hosted in a traditional infrastructure as a service (IaaS) solution, the effort to improve RPO and RTO can be significantly higher. Yet the business stakeholders' desire to achieve better management commitments is seldom affected by a PaaS versus IaaS decision. If anything, an understanding of the

fundamental differences in architecture might prompt the business to ask for PaaS solutions or commitments that match what's available on PaaS solutions. Modernization of any IaaS data platforms should be considered as a first step into platform operations.

When modernization isn't an option, cloud management teams commonly prioritize IaaS-based data platforms as a first required service in the service catalog. Providing the business with a choice between standalone data servers and clustered, high-availability, data solutions makes the business commitment conversation much easier to facilitate. A basic understanding of the operational improvements and the increased costs will help the business make the best decision for the business processes and supporting workloads.

Other common platform operations

In addition to data platforms, virtual machine hosts tend to be a common platform for operations improvements. Most commonly, cloud platform and cloud management teams invest in improvements to VMware hosts or container solutions. Such investments can improve the stability and reliability of the hosts, which support the VMs, which in turn power the workloads. Proper operations on one host or container can improve the RPO or RTO of several workloads. This approach creates improved business commitments, but distributes the investment. Improved commitments and reduced costs combine to make it much easier to justify improvements to cloud management and platform operations.

Next steps

In parallel with improvements to platform operations, cloud management teams also focus on improving workload operations for the top 20 percent or less of production workloads.

Improve workload operations

Workload operations in cloud management

5/4/2021 • 5 minutes to read • Edit Online

Some workloads are critical to the success of the business. For those workloads, a management baseline is insufficient to meet the required business commitments to cloud management. Platform operations might not even be sufficient to meet business commitments. This highly important subset of workloads requires a specialized focus on the way the workload functions and how it's supported.

In return, the investment in workload operations can lead to improved performance, decreased risk of business interruption, and faster recovery when system failures occur. This article discusses an approach to investing in the continued operations of these high priority workloads to drive improved business commitments.

When to invest in workload operations

The *Pareto principle* (also known as the *80/20 rule*) states that 80 percent of effects come from 20 percent of the causes. When IT portfolios are allowed to grow organically over time, this rule is often illustrated in a review of the IT portfolio. Depending on the effect that requires investment, the cause can vary but the general principle holds true:

- 80 percent of system failures tend to be the result of 20 percent of the common errors or bugs.
- 80 percent of business value tends to come from 20 percent of the workloads in a portfolio.
- 80 percent of the effort to migrate to the cloud comes from 20 percent of the workloads being moved.
- 80 percent of cloud management efforts will support 20 percent of the service incidents or trouble tickets.
- 80 percent of business impact from an outage will come from 20 percent of the systems affected by the outage.

Workload operations should be applied only when the cloud adoption strategy, business outcomes, and operational metrics are each well understood. This is a paradigm shift from the classic view of IT. Traditionally, IT assumed that all workloads experienced the same degree of support and required similar levels of priority.

Before they invest in deep workload operations, both IT and the business should understand the business justifications and the expectations of increased investment in cloud management.

Start with the data

Workload operations begin with a deep understanding of workload performance and support requirements. Before the team invests in workload operations, it must have rich data about workload dependencies, application performance, database diagnostics, virtual machine telemetry, and incident history.

This data seeds the insights that drive workload operations decisions.

Continued observation

Initial data and ongoing telemetry can help formulate and test theories about the performance of a workload. But ongoing workload operations are rooted in a continued and expanded observation of workload performance, with a heavy focus on application and data performance.

Test the automation

At the application level, the first requirements of workload operations, is an investment in deep testing. For any application that's supported through workload operations, a test plan should be established and regularly executed to deliver functional and scale testing across the applications.

Regular test telemetry can provide immediate validation of various hypotheses about the operation of the workload. Improving operational and architectural patterns can be executed and tested. The resulting deltas provide a clear impact analysis to guide continued investments.

Understand releases

A clear understanding of release cycles and release pipelines is an important element of workload operations.

An understanding of cycles can prepare for potential interruptions and allow the team to proactively address any releases that might produce an adverse effect on operations. This understanding also allows the cloud management team to partner with adoption teams to continuously improve the quality of the product and address any bugs that might affect stability.

More importantly, an understanding of release pipelines can significantly improve the recovery point objective (RPO) of a workload. In many scenarios, the fastest and most accurate path to the recovery of an application is a release pipeline. For application layers that change only when a new release happens, it might be wise to invest more heavily in pipeline optimization than on the recovery of the application from traditional back-up processes.

Although a deployment pipeline can be the fastest path to recovery, it can also be the fastest path to remediation. When an application has a fast, efficient, and reliable release pipeline, the cloud management team has an option to automate deployment to a new host as a form of automated remediation.

There might be many other faster, more effective mechanisms for remediation and recovery. However, when the use of an existing pipeline can meet business commitments and capitalize on existing DevOps investments, the existing pipeline might be a viable alternative.

Clearly communicate changes to the workload

Change to any workload is among the biggest risks to workload operations. For any workload in the workload operations level of cloud management, the cloud management team should closely align with the cloud adoption teams to understand the changes coming from each release. This investment in proactive understanding will have a direct, positive impact on operational stability.

Improve outcomes

The data and communication investments in a workload will yield suggestions for improvements to ongoing operations in one of three areas:

- Technical debt resolution
- Automated remediation
- Improved system design

Technical debt resolution

The best workload operations plans still require remediation. As your cloud management team seeks to stay connected to understand adoption efforts and releases, the team likewise should regularly share remediation requirements to ensure that technical debt and bugs are a continued priority for your development teams.

Automated remediation

By applying the Pareto principle, we can say that 80 percent of negative business impact likely comes from 20 percent of the service incidents. When those incidents can't be addressed in normal development cycles, investments in remediation automation can significantly reduce business interruptions.

Improved system design

In the cases of technical debt resolution and automated remediation, system flaws are the common cause of most system outages. You can have the greatest impact on overall workload operations by adhering to a few design principles:

- **Scalability:** The ability of a system to handle increased load.
- **Availability:** The percentage of time that a system is functional and working.
- **Resiliency:** The ability of a system to recover from failures and continue to function.
- **Management:** Operations processes that keep a system running in production.
- **Security:** Protecting applications and data from threats.

To help improve overall operations, the Microsoft Azure Well-Architected Framework provides an approach to evaluating specific workloads for adherence to these pillars. Apply the pillars to both platform operations and workload operations.

Next steps

With a full understanding of the Manage methodology within the Cloud Adoption Framework, you are now armed to implement cloud management principles. Learn how to make this methodology actionable within your operations environment.

Apply this methodology

Apply design principles and advanced operations

5/4/2021 • 6 minutes to read • Edit Online

The first three cloud management disciplines describe a management baseline. At a minimum, a management baseline should include a standard business commitment to minimize business interruptions and accelerate recovery if service is interrupted. Most management baselines include a disciplined focus on maintaining "inventory and visibility," "operational compliance," and "protection and recovery."

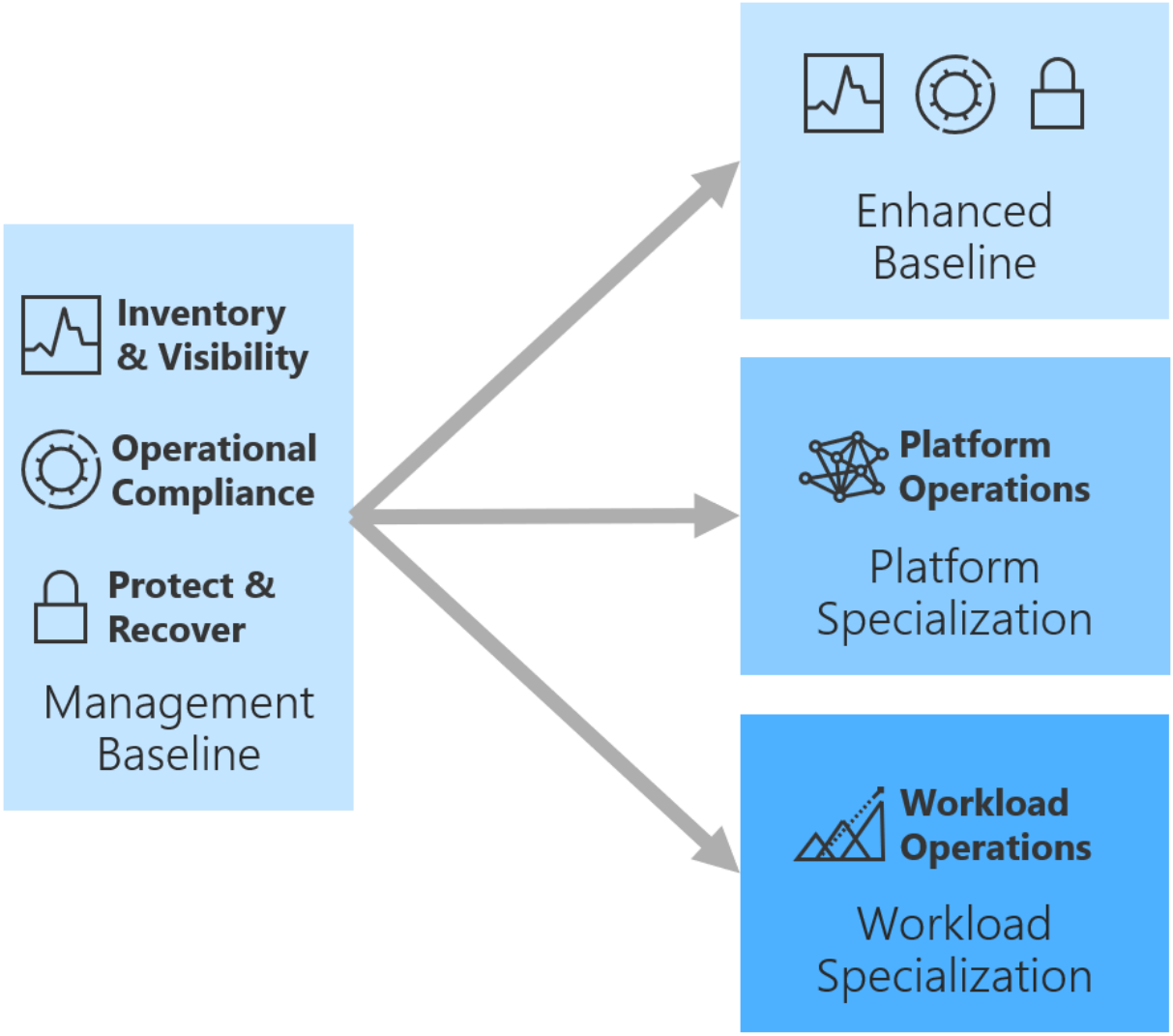
The purpose of a management baseline is to create a consistent offering that provides a minimum level of business commitment for all supported workloads. This baseline of common, repeatable management offerings allows the team to deliver a highly optimized degree of operational management, with minimal deviation. But that standard offering might not provide a rich enough commitment to the business.

The diagram in the next section illustrates three ways to go beyond the management baseline.

The management baseline should meet the minimum commitment required by 80 percent of the lowest criticality workloads in the portfolio. The baseline should not be applied to mission-critical workloads. Nor should it be applied to common platforms that are shared across workloads. Those workloads require a focus on design principles and advanced operations.

Advanced operations options

There are three suggested paths for improving business commitments beyond the management baseline, as shown in the following diagram:



Enhanced management baseline

As outlined in the Azure Management Guide, an enhanced management baseline uses cloud-native tools to improve uptime and decrease recovery times. The improvements are significant, but less so than with workload or platform specialization. The advantage of an enhanced management baseline is the equally significant reduction in cost and implementation time.

Management specialization

Aspects of workload and platform operations might require changes to design and architecture principles. Those changes could take time and might result in increased operating expenses. To reduce the number of workloads requiring such investments, an enhanced management baseline could provide enough of an improvement to the business commitment.

For workloads that warrant a higher investment to meet a business commitment, specialization of operations is key.

Areas of management specialization

There are two areas of specialization:

- **Platform specialization:** Invest in ongoing operations of a shared platform, distributing the investment across multiple workloads.
- **Workload specialization:** Invest in ongoing operations of a specific workload, generally reserved for

mission-critical workloads.

Central IT team or cloud center of excellence (CCoE)

Decisions between platform specialization and workload specialization are based on the criticality and impact of each workload. However, these decisions are also indicative of larger cultural decisions between central IT team and CCoE organizational models.

Workload specialization often triggers a cultural change. Traditional IT and centralized IT both build processes that can provide support at scale. Scale support is more achievable for repeatable services found in a management baseline, enhanced baseline, or even platform operations. Workload specialization doesn't often scale. This lack of scale makes it difficult for a centralized IT organization to provide necessary support without reaching organizational scale limitations.

Alternatively, a cloud center of excellence approach scales through purposeful delegation of responsibility and selective centralization. Workload specialization tends to better align with the delegated responsibility approach of a CCoE.

The natural alignment of roles in a CCoE is outlined as follows:

- The cloud platform team helps build common platforms that support multiple cloud adoption teams.
- The cloud automation team extends those platforms into deployable assets in a service catalog.
- Cloud management delivers the management baseline centrally and helps support the use of the service catalog.
- But the business unit (in the form of a business DevOps team or cloud adoption team) holds responsibility for day-to-day operations of the workload, pipeline, or performance.

As for aligning areas of management, central IT team and CCoE models can generally deliver on platform specialization, with minimal cultural change. Delivering on workload specialization might be more complex for central IT teams.

Management specialization processes

Within each specialization, the following four-step process is delivered in a disciplined, iterative approach. This approach requires partnership among cloud adoption, cloud platform, cloud automation, and cloud management experts to create a viable and informed feedback loop.

- **Improve system design:** Improve the design of common systems (platforms) or specific workloads to effectively minimize interruptions.
- **Automate remediation:** Some improvements are not cost effective. In such cases, it might make more sense to automate remediation and reduce the impact of interruptions.
- **Scale the solution:** As systems design and automated remediation are improved, you can scale those changes across the environment through the service catalog.
- **Continuous improvement:** You can use various monitoring tools to discover incremental improvements to address in the next pass of system design, automation, and scale.

Improve system design

Improving system design is the most effective approach to improving operations of any common platform. System design improvements can help increase stability and decrease business interruptions. Design of individual systems is out of scope for the environment view taken throughout the Cloud Adoption Framework.

As a complement to this framework, the Microsoft Azure Well-Architected Framework provides guiding tenets for improving the quality of a platform or a specific workload. The framework focuses on improvement across five pillars of architecture excellence:

- **Cost optimization:** Manage costs to maximize the value delivered.

- **Operational excellence:** Follow operational processes that keep a system running in production.
- **Performance efficiency:** Scale systems to adapt to changes in load.
- **Reliability:** Design systems to recover from failures and continue to function.
- **Security:** Protect applications and data from threats.

Most business interruptions equate to some form of technical debt, or deficiency in the architecture. For existing deployments, systems design improvements can be viewed as payments against existing technical debt. For new deployments, systems design improvements can be viewed as avoidance of technical debt. The next section, "Automated remediation," looks at ways to address technical debt that can't or shouldn't be addressed.

To improve system design, learn more about the Microsoft Azure Well-Architected Framework. As your system design improves, return to this article to find new opportunities to improve and scale the improvements across your environment.

Automated remediation

Some technical debt can't or shouldn't be addressed. Resolution could be too expensive to correct. It could be planned but might have a long project duration. The business interruption might not have a significant business impact, or the business priority is to recover quickly instead of investing in resiliency.

When resolution of technical debt isn't the desired path, automated remediation is commonly the desired next step. Using Azure Automation and Azure Monitor to detect trends and provide automated remediation is the most common approach to automated remediation.

For guidance on automated remediation, see Azure Automation and alerts.

Scale the solution with a service catalog

The cornerstone of platform specialization and platform operations is a well-managed service catalog. This is how improvements to systems design and remediation are scaled across an environment. The cloud platform team and cloud automation team align to create repeatable solutions to the most common platforms in any environment. However, if those solutions aren't consistently applied, cloud management can provide little more than a baseline offering.

To maximize adoption and minimize maintenance overhead of any optimized platform, the platform should be added to a service catalog. Each application in the catalog can be deployed for internal consumption via the service catalog, or as a marketplace offering for external consumers.

For information about publishing to a service catalog, see the series on publishing to a service catalog.

Continuous improvement

Platform specialization and platform operations both depend on strong feedback loops between adoption, platform, automation, and management teams. Grounding those feedback loops in data empowers each team to make wise decisions. For platform operations to achieve long-term business commitments, it's important to take advantage of insights that are specific to the centralized platform. Because containers and SQL Server are the two most common centrally managed platforms, consider beginning with continuous improvement data collection by reviewing the following articles:

- Container performance
- PaaS database performance
- IaaS database performance

Cloud operation and management antipatterns

5/4/2021 • 2 minutes to read • Edit Online

Customers often experience antipatterns during the operation or management phase of cloud adoption. By introducing new or modernized tool chains with caution, you can often avoid these antipatterns.

Antipattern: Focus on tooling, not business outcomes

Modernized IT tooling can improve work environments by relieving employees of tedious tasks. It's important to measure new IT tooling so that you can identify whether it improves business outcomes. A new or modernized tool chain doesn't automatically provide faster delivery or a better business outcome.

Example: Introduce a platform that doesn't improve performance

A company introduces a new, improved version of its continuous integration and continuous delivery (CI/CD) platform. The tool makes it easier to define delivery and deployment pipelines, so you can deploy features faster. The IT department is enthusiastic about delivering a platform that speeds up the pipeline configuration. Once a business unit uses the tool, it discovers that the time to market isn't significantly better, compared with the old platform. The final approval and release process isn't changed or improved.

Preferred outcome: Measure success with business outcomes

To keep your technology and business goals aligned, have leaders from both areas jointly define desired outcomes. Make sure these outcomes and goals are specific, measurable, achievable, reasonable, and time-bound (SMART). Ensure that the outcomes and goals have an impact on technology and the business. The Microsoft Cloud Adoption Framework for Azure can help to determine a proper commitment within the business.

Don't use simple technology outputs (such as faster deployment and pipeline configurations) to measure success. Instead, use technology and business outcomes. For help with this task, see Developer velocity.

Next steps

- Business commitment in cloud management

Security in the Microsoft Cloud Adoption Framework for Azure

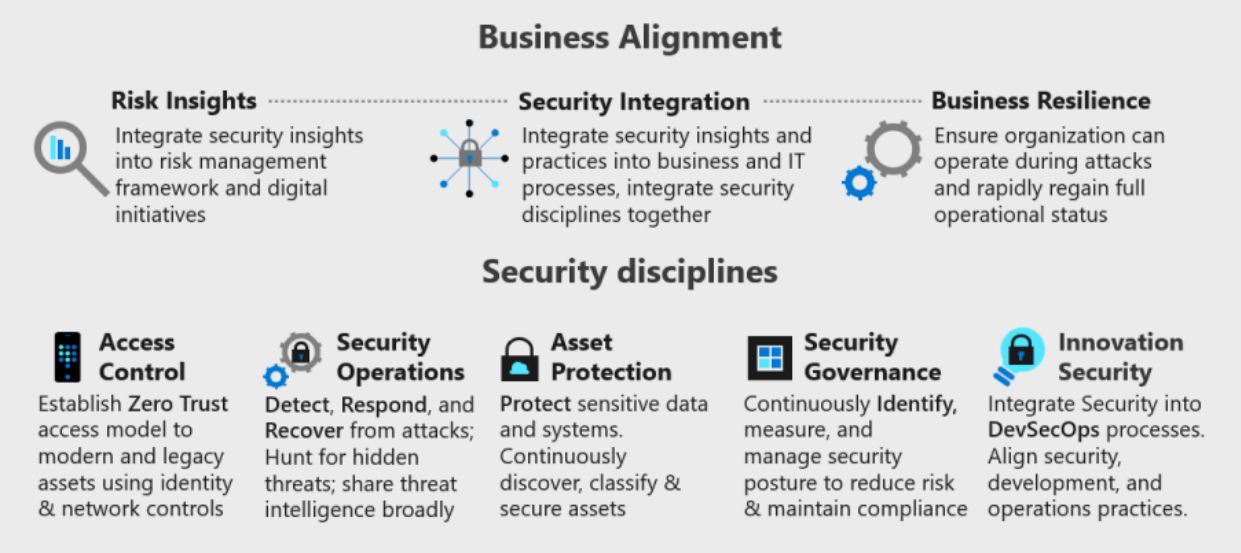
6/22/2021 • 8 minutes to read • Edit Online

Just as cloud adoption is a journey, cloud security is also an ongoing journey of incremental progress and maturity, not a static destination.

Envision a security end state

A journey without a target destination is just wandering. While this approach might eventually lead to enlightenment, business goals and constraints often require focusing on objectives and key results.

The Secure methodology provides a vision of the complete end state to guide the improvement of your security program over time. The following infographic provides a visual mapping of the key ways that security integrates with the larger organization and the disciplines within security.



The Cloud Adoption Framework provides security guidance for this security journey by providing clarity for the processes, best practices, models, and experiences. This guidance is based on the lessons learned and real world experiences of real customers, Microsoft's security journey, and work with organizations like NIST, The Open Group, and the Center for Internet Security (CIS).

Mapping to concepts, frameworks, and standards

Security itself is both a standalone organizational discipline and a quality/attribute that is integrated or overlaid on other disciplines, which make it difficult to precisely define and map in detail. The security industry uses many different frameworks to capture risk, plan controls, and operate. Here is a quick summary of how the disciplines in the CAF Secure methodology relate to other security concepts and guidance:

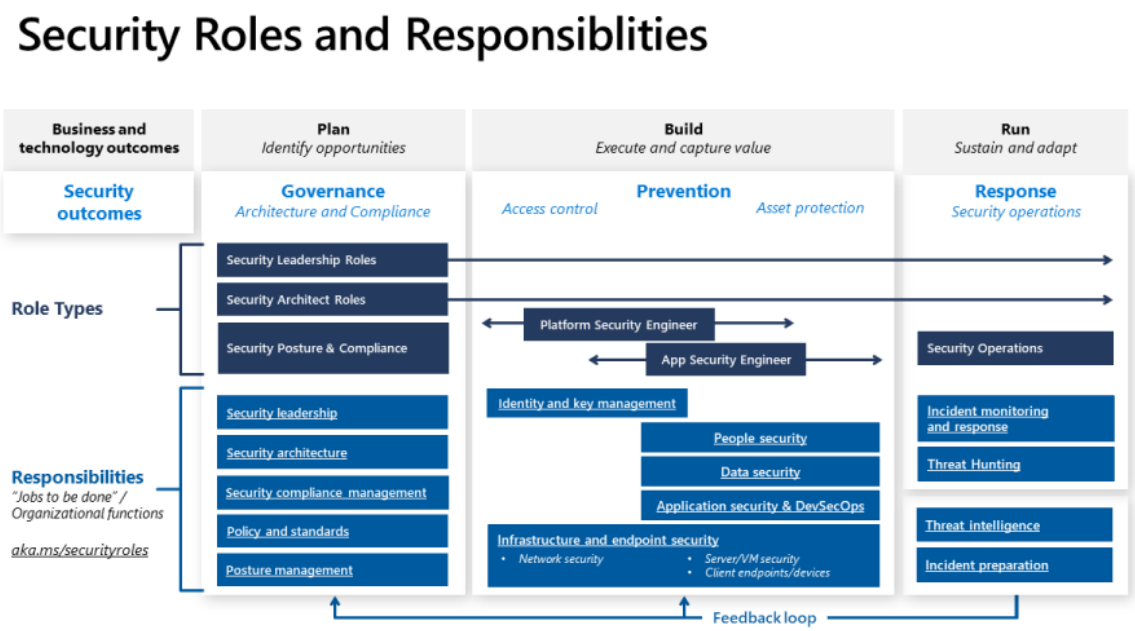
- **Zero trust:** Microsoft believes all security disciplines should follow the zero-trust principles of *assume breach*, *verify explicitly*, and *use least privilege access*. These principles underpin any sound security strategy and also must be balanced with *business enablement* goals. The first and most visible part of zero trust is in access control, so it's highlighted in the description of access control security discipline.
- **The Open Group:** These security disciplines map closely to the zero-trust components in the core principles white paper published by The Open Group, where Microsoft actively participates. The one

notable exception is that Microsoft elevated the discipline of innovation security so that DevSecOps is a top-level element because of how new, important, and transformative this discipline is for many organizations.

- **NIST cybersecurity framework:** For organizations that use the NIST cybersecurity framework, we have highlighted bold text where the framework most closely maps. Modern access control and DevSecOps map broadly to the full spectrum of the framework, so those items aren't noted individually.

Mapping to roles and responsibilities

While security is a highly technical discipline, it's first and foremost a human discipline reflective of the long history of human conflict (but updated for computers and the internet). The following diagram summarizes the roles and responsibilities in a security program.



For more information, see Cloud security functions.

Security transformation

As organizations adopt the cloud, they quickly find that static security processes cannot keep up with the pace of change in cloud platforms, the threat environment, and the evolution of security technologies. Security must shift to a continuously evolving approach to match pace with this change that will transform organizational culture and daily processes throughout the organization.

To guide this transformation, this methodology provides guidance on the integration of security with business processes (top row) and security technical disciplines (bottom row). These collectively enable meaningful and sustainable progress on your security journey to reduce organizational risk. Few organizations can master all of these at once, but all organizations should steadily mature each process and discipline.

Change drivers

Security organizations are experiencing two types of major transformations at the same time

- **Security as business risk:** Security has been propelled into the realm of business risk management from a purely technical quality-oriented discipline. This is driven by dual forces of:
  - **Digital transformation:** Increases in digital footprint are continuously increasing the potential attack surface of the organization
  - **Threat landscape:** Increases in attack volume and sophistication that are fueled by an industrialized

attack economy with specialized skills and continual commoditization of attack tools and techniques.

- **Platform change:** Security is also grappling with a technical platform change to the cloud. This shift is on the scale of factories shifting from running their own electrical generators to plugging into an electrical grid. While security teams often have the right foundational skills, they are becoming overwhelmed by the changes to nearly every process and technology they use everyday.
- **Shift in expectations:** In the past decade, digital innovation has redefined entire industries. Business agility, especially agility related to digital transformation, can quickly unseat an organization as a market leader. Likewise, loss of consumer confidence can have a similar impact on the business. While it was once acceptable for security to start with "no" to block a project and protect the organization, the urgency of embracing digital transformation must change the engagement model to "let's talk about how to stay safe while you do what is needed to stay relevant."

Guiding lasting transformation

Transforming how the business and tech teams view security requires aligning security closely to the priorities, processes, and risk framework. Key areas that drive success are

- **Culture:** The culture of security must be focused on safely meeting the business mission, not impeding it. At the same time, security must become a normalized part of the culture of the organization as the internet upon which the business operates is open, allow adversaries to attempt attacks at any time. This cultural shift requires improved processes, partnerships, and ongoing leadership support at all levels to communicate the change, model the behavior, and reinforce the shift.
- **Risk ownership:** The accountability for security risk should be assigned to the same roles that own all other risks, freeing security up to be a trusted advisor and subject matter expert rather than a scapegoat. Security should be responsible for sound and balanced advice that is communicated in the language of those leaders, but should not be held accountable for decisions they do not own.
- **Security talent:** Security talent is in a chronic shortage and organizations should always be planning how to best develop and distribute security knowledge and skills. In addition to growing security teams directly with technical security skillsets, mature security teams are also diversifying their strategy by focusing on
  - **Growing security skillsets and knowledge** within existing teams in IT and the business. This is especially important for DevOps teams with a DevSecOps approach and can take many forms (such as a security help desk, identifying and training champions within the community, or job swapping programs).
  - **Recruiting diverse skillsets** to security teams to bring fresh perspectives and frameworks to problems (like business, human psychology, or economics) and build better relationships within the organization. To a hammer, all problems look like nails.

Business alignment

Because of these shifts, your cloud adoption program should focus heavily on business alignment in three categories

- **Risk insights:** Align and integrate security insights and risk signals/sources to the business initiatives. Ensure repeatable processes educate all teams on the application of those insights and hold teams accountable for improvements.
- **Security integration:** Integrate security knowledge, skills, and insights deeper into daily operations of the business and IT environment via repeatable processes and deep partnership at all levels of the organization.
- **Operational resiliency:** Focus on ensuring the organization is resilient by being able to continue operations during an attack (even if at a degraded state) and that the organization rapidly bounces back to full operations.

Security disciplines

This transformation will affect each security discipline differently. While each of these disciples is extremely important and requires investment, these are ordered (roughly) by which ones have the most immediate

opportunities for quick wins as you adopt the cloud:

- **Access control:** Application of network and identity create access boundaries and segmentation to reduce the frequency and reach of any security breaches
- **Security operations:** Monitor IT operations to detect, respond, and recover from breach. Use data to continuously reduce risk of breach
- **Asset protection:** Maximize protection of all assets (infrastructure, devices, data, applications, networks, and identities) to minimize risk to the overall environment
- **Security governance:** Delegated decisions accelerate innovation and introduce new risks. Monitor decisions, configurations, and data to govern decisions made across the environment and within all workloads across the portfolio.
- **Innovation security:** As an organization adopts DevOps models to increase the pace of innovation, security must become an integral part of a DevSecOps process and integrate security expertise and resources directly into this high-speed cycle. This involves shifting some decision making from centralized teams to empower workload-focused teams.

Guiding principles

All security activities should be aligned to and shaped by a dual focus on

- **Business enablement:** Align to organization's business objective and risk framework
- **Security assurances:** Focused on applying zero trust principles of
  - **Assume breach:** When designing security for any component or system, reduce risk of attackers expanding access by assuming other resources in the organization are compromised
  - **Explicit verification:** Explicitly validate trust using all available data points, rather than assuming trust. For example, in access control, validate the user identity, location, device health, service or workload, data classification, and anomalies, rather than simply allowing access from an implicitly trusted internal network.
  - **Least-privileged access:** Limit the risk of a compromised user or resource by providing just-in-time and just-enough-access (JIT/JEA), risk-based adaptive policies, and data protection to help secure both data and productivity.

Related resources

The Secure methodology is part of a comprehensive set of security guidance that also includes:

- Azure Well-Architected Framework: Guidance on securing your workloads on Azure.
- Azure security benchmarks: Prescriptive best practices and controls for Azure security.
- Enterprise-scale landing zone: Azure reference architecture and implementation with integrated security.
- Top 10 security best practices for Azure: Top Azure security best practices that Microsoft recommends based on lessons learned across customers and our own environments.

Risk management insights

6/8/2021 • 6 minutes to read • Edit Online

Operating a business comes with risk. A security team's role is to inform and advise decision makers on how security risks fit in their frameworks. The goal of security is to learn about your business, and then use their security expertise to identify risks to business goals and assets. Security then advises your decision makers on each risk and then recommends which ones are acceptable. This information is provided with the understanding that the responsibility for these decisions lies with your asset or process owner.

> **NOTE**
>
> The general rule for risk accountability is:
>
> The person who owns and accepts the risk is the person that explains to the world what went wrong (often in front of TV cameras).

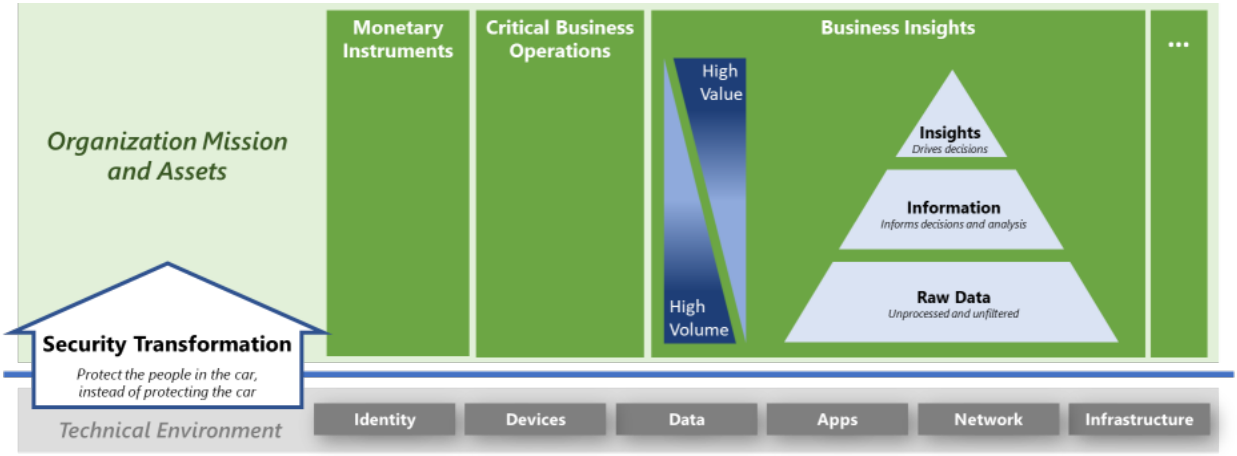
When mature, the goal of security is to expose and mitigate risks, and then empower the business to change with minimal risk. That level of maturity requires risk insights and deep security integration. At any level of maturity for your organization, the top security risks should appear on the risk register. Those risks are then managed down to an acceptable level.

What is cybersecurity risk?

Cybersecurity risk is the potential damage or destruction of business assets, revenue, and reputation. This damage is caused by human attackers attempting to steal money, information, or technology.

While these attacks happen in the technical environment, they often represent a risk to your entire organization. Cybersecurity risk should be aligned to your risk measurement, tracking, and mitigation framework. Many organizations still treat cybersecurity risk as a technical problem to be *solved*. This perception leads to the wrong conclusions that don't mitigate the strategic business impact of risk.

The following diagram shows the shift from a typical technically oriented program to a business framework.



Security leaders must step back from the technical lens and learn what assets and data are important to business leaders. Then prioritize how teams spend their time, attention, and budget relating to business importance. The technical lens is reapplied as the security and IT teams work through solutions. But looking at cybersecurity risk only as a technology problem runs the risk of solving the wrong problems.

Align your security risk management

Continuously work to build a stronger bridge between cybersecurity and your organizational leadership. This concept applies to both human relationships and explicit processes. The nature of security risk and the divergent dynamics of business opportunities are always changing. Security risk sources require ongoing investment into building and improving this relationship.

Key to this relationship is understanding how business value connects to specific technical assets. Without this direction, security can't be sure what's most important to your organization. They only succeed at protecting the most important assets with lucky guesses.

It's important to begin this process immediately. Start by gaining a better understanding of sensitive and business critical assets in your organization.

The typical process of starting this transformation is:

1. **Align the business** in a two-way relationship:
   - **Communicate in their language** to explain security threats using business-friendly terminology. This explanation helps to quantify the risk and impact to the overall business strategy and mission.
   - **Actively listen and learn** by talking to people across the business. Work to understand the impact to important business services and information if they were compromised or breached. This understanding provides clear insight into the importance of investing in policies, standards, training, and security controls.
2. **Translate learnings** about business priorities and risks into concrete and sustainable actions, such as:
   - **Short term** that focuses on dealing with important priorities.
     - Protect critical assets and high-value information with appropriate security controls. These controls increase security while enabling business productivity.
     - Focus on immediate and emerging threats that are most likely to cause business impact.
     - Monitor changes in business strategies and initiatives to stay in alignment.
   - **Long term** sets direction and priorities to make steady progress over time, improving the overall security posture.
     - Use **Zero trust** to create a strategy, plan, and architecture for reducing risks in your organization. Align them with the zero trust principles of assuming breach, least privilege, and explicit verification. Adopting these principles shifts from static controls to more dynamic risk-based decisions. These decisions are based on real-time detections of strange behavior no matter where the threat started.
     - **Pay off technical debt** as a consistent strategy by operating security best practices across the organization. For example, replace password-based authentication with passwordless and multifactor authentication, apply security patches, and retire or isolate legacy systems. Like paying off a mortgage, you must make steady payments to realize the full benefit and value of your investments.
     - **Apply data classifications, sensitivity labels, and role-based access control** to protect data from loss or compromise throughout its lifecycle. These efforts can't completely capture the dynamic nature and richness of business context and insight. But these key enablers are used to guide information protection and governance, limiting the potential impact of an attack.
3. **Establish a healthy security culture** by explicitly practicing, communicating, and publicly modeling the right behavior. The culture should focus on open collaboration between business, IT, and security colleagues. Then apply that focus to a 'growth mindset' of continuous learning. Focus culture changes on removing silos from security, IT, and the larger business organization. These changes achieve greater knowledge sharing and resilience levels.

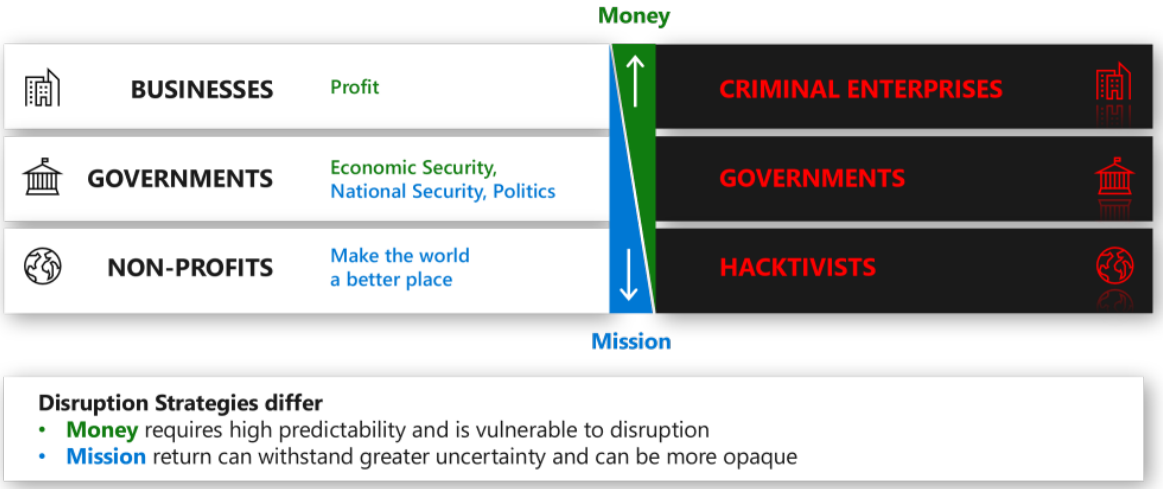
For more information, see Define a security strategy.

Understanding cybersecurity risk

Cybersecurity risk is caused by human attackers trying to steal money, information, or technology. It's important to understand the motivations and behavior patterns of these attackers.

Motivations

The motivations and incentives for different types of attackers mirror those of legitimate organizations.



Understanding the attackers' motivations can help you understand the likelihood and potential impact of different types of attacks. While the security strategies and most important technical controls are similar across organizations, this context can help guide your security investment focus areas.

For more information, see Disrupting attacker return on investment.

Behavior patterns

Organizations face a range of human attacker models that shape their behavior:

- **Commodity:** Most of the threats organizations typically face are for-profit attackers, driven by a financial return on investment (ROI). These attackers typically use the cheapest and most effective available tools and methods. The sophistication of these attacks (for example, stealth and tooling) typically grows as new methods are proven by others and made available for high scale use.
- **Leading edge:** Sophisticated attack groups are driven by long-term mission outcomes and often have funding available. This funding is used to focus on innovation. This innovation might include investing in supply chain attacks or changing tactics within an attack campaign to hinder detection and investigation.

Typically, the attackers are:

- **Flexible:** They use more than one attack vector to gain entry to the network.
- **Objective-driven:** They achieve a defined purpose from accessing your environment. The goals might be specific to your people, data, or applications, but you might also fit a certain class of targets. For example, "A profitable company that's likely to pay to restore access to their data and systems."
- **Stealthy:** They take precautions to remove evidence or hide their tracks, usually at different investment and priority levels.
- **Patient:** They take time to do reconnaissance to understand your infrastructure and business environment.
- **Well-resourced and skilled:** They're educated in the technologies they're targeting, though the depth of skill can vary.
- **Experienced:** They use established techniques and tools to gain elevated privileges to access or control different aspects of the estate.

Next steps

For risk management to be effective, it must be rolled into all aspects of your governance and compliance activities. For risks to be properly assessed, security must always be considered as part of a comprehensive approach.

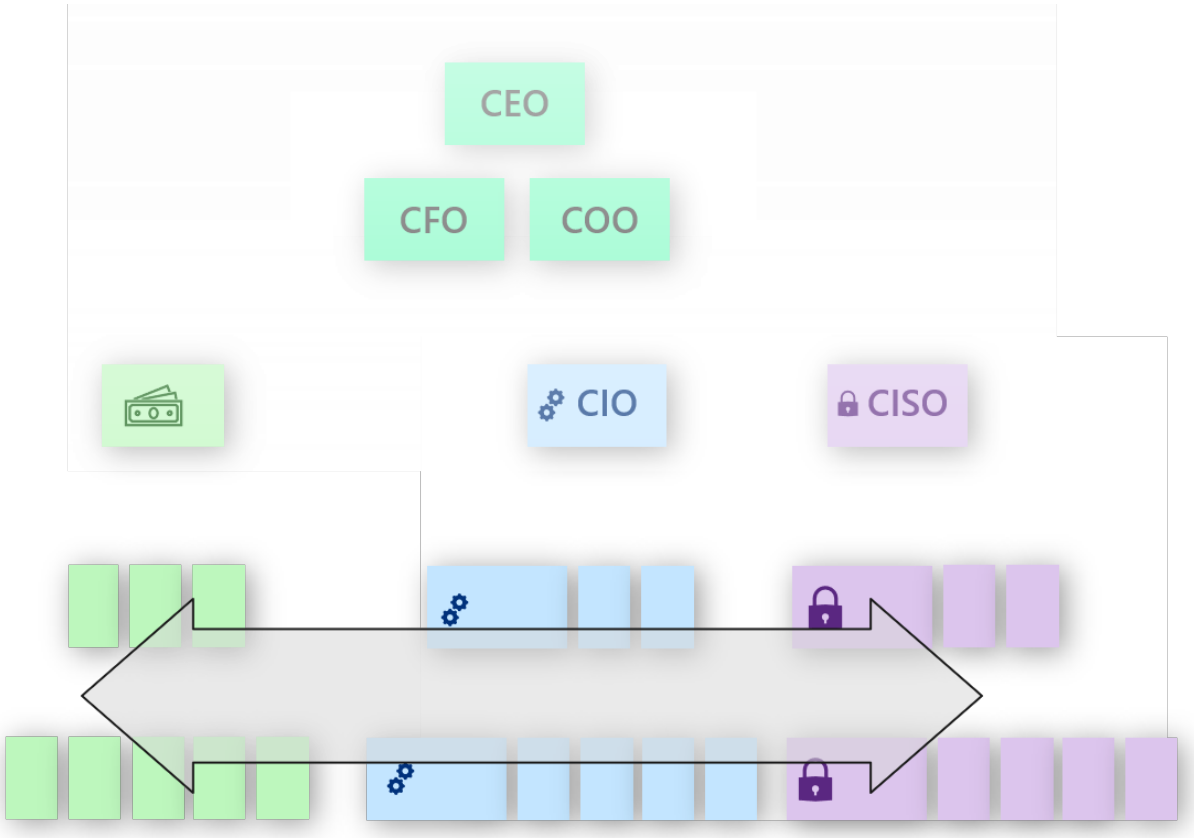
- Security integration is the next focus area of security.

Security integration

6/8/2021 • 11 minutes to read • Edit Online

In your organization, security should be a part of everyone's job, just like business requirements, performance, and reliability. Security, at all levels, should be familiar with overall business priorities, IT initiatives, and risk appetite for your organization. Envision security as a thread that weaves through every aspect of your business. Security should feel like natural part of business. Business should feel like a natural part of security.

Your organization must maintain security assurances while minimizing friction with business processes.



Internal friction and low levels of conflict between teams might occur in organizations. Such conflicts aren't sustainable. It's critical for all of the teams to work together in the age of cloud, digital business, and zero trust security. Teams operating with different goals, culture, and language make an organization inefficient and ineffective.

Be sure that security teams aren't operating in silos. The teams should work closely together to ensure smooth process operations and knowledge sharing.

This guidance describes how you can improve security integration with business and IT teams and integration between security teams.

Normalize relations

Overcoming the silo approach that is prevalent at many organizations can be a challenge, however it can be done. The main elements are to be clear about the end state, provide clarity on the process, and provide ongoing leadership support for tangible goals and for changes in culture and behavior. The following elements are critical to the process:

- Identify shared goals and outcomes.

- Identify the right level of security.

Identify shared goals and outcomes

Ensure there's a shared understanding of your goals that involves all teams. Security teams sometimes define themselves as quality control over the business and IT functions. This approach creates an adversarial dynamic and generates friction. Business productivity, IT goals, and security goals can suffer from this dynamic.

Be sure that security teams are closely integrated with their IT and business counterparts. Security teams are jointly responsible for the business, IT, and security outcomes of any initiative. Share the challenges of designing systems to meet business and IT goals. Share security perspective and expertise at the right time.

As systems are being designed, implemented, operated, and continuously improved, it's critical to set guardrails so that *no voice dominates all decisions* that are related to business, IT, or security.

Identify the right level of security

Some security controls like biometric sign-in with Windows Hello for Business have the dual benefits of making the user experience better and the security stronger. Many security measures add friction to business processes and can slow them down. We strive to first find security measures that are easy and invisible to users and developers. We must accept that sometimes a tradeoff must be made.

These joint teams should always strive for a healthy level of friction in the process that creates value by driving critical thinking at the right time. For example, you might consider what an attacker might do with a new feature or how much business impact would come from altering some data.

Teams should strive for the best balance between two absolute truths:

- Security cannot be skipped. Skipping security often leads to incidents that cost more in the end (productivity, revenue, overall business impact) than integrating security would have.
- Security controls can reach an unhealthy level of friction where the security friction impedes more value than it protects.

It's critical to find a balance as security is integrated into the process. Stakeholders must all work together to ensure business concerns, IT reliability and performance concerns, and security concerns are considered and balanced. Organizations need to also work on solving for the 80 percent and planning for the other 20 percent. By holding off on controls, features, and capabilities for security until there is a 100 percent solution, organizations will put everything they do at risk of exposure. Iterative approaches work well, as do the basics of updating and education.

For more information about healthy security friction, see the right level of security friction in the security strategy guidance.

The next section illustrates how to integrate security stakeholders with IT, end users, and workload owners. It also includes examples for within the security team.

Integrate with IT and business operations

While most security functions operate out of sight, some security considerations appear in daily business and IT workflows. Security thinking must be integrated into the normal experience of planning and operating a business.

Security update process

Security updates are one of the most common and visible points where business processes and security processes interact. It's a common source of friction because it involves the difficult balance of two different forces, that map to separate stakeholders in the organization:

- **Immediate business impact:** Security updates often require testing and system reboots, which consume time and resources for application owners and IT teams, and potentially result in business impact from

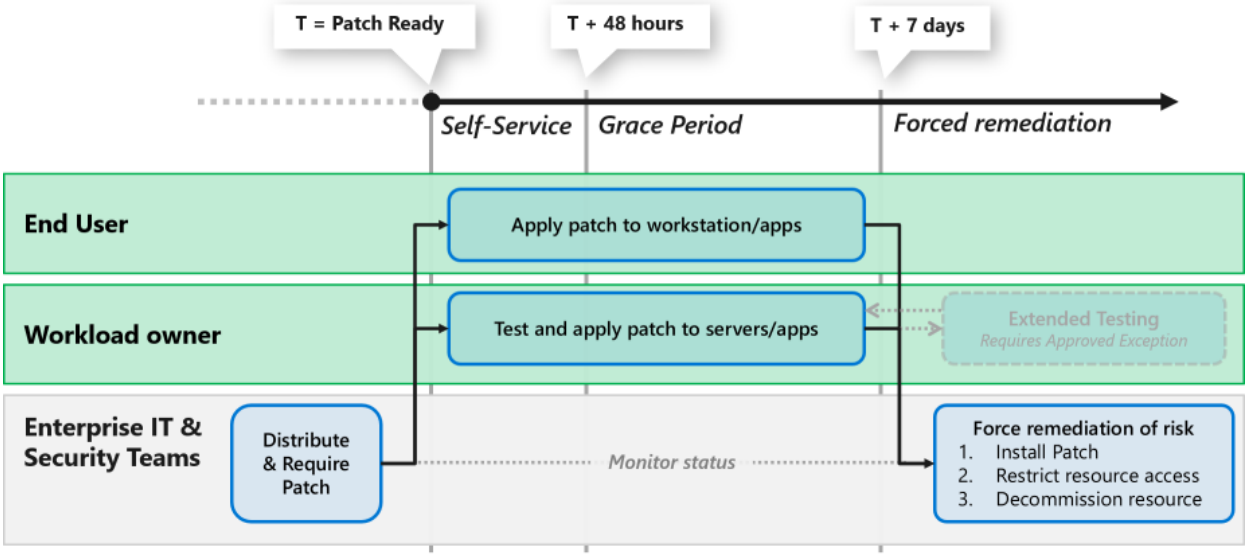
downtime.

- **Potential future business impact as security risk:** If updates aren't applied completely, attackers can exploit the vulnerabilities and cause business impacts.

If teams operate without shared goals and responsibility (for example, IT and the business focused 100 percent on immediate business impact, security accountable for 100 percent of the security risk), then they will be in constant conflict over security updates. This conflict distracts the teams with endless arguments rather than them working together to solve the problem, which allows them to move onto the next problem, risks, and business value creation opportunity. Ongoing communication across the organization and creating a culture where updates are embraced will go a long way to limit push back from end users. If the users know they'll be better protected, able to be more productive, and be able to build the business because security is side by side with them, they will be more likely to embrace updates and ongoing education.

Correctly placing the accountability for all benefits and risk on the asset owners makes it easier for them to consider immediate and potential future impact. Making identifying the solution a joint responsibility of all subject matter experts in security, IT, and business increases the quality of the solution by considering more and diverse perspectives. Make everyone a stakeholder in the security story across the company. While everyone might not have security as their day to day role, they do have security requirements to perform their role.

This example process illustrates how organizations are starting to solve this problem using shared responsibility and flexibility within a limited time frame:



This process is followed on a regular schedule:

- Enterprise IT and security teams start the process by identifying which security updates, or patches, are required and the highest impact. They make these updates available to end users or workload owners through enterprise-wide distribution channels.
- End users have a fixed period to test the updates, apply them, and restart their devices. After that period expires, enterprise IT and security teams either apply the update or block access to corporate resources. They might use a mechanism like Azure AD Conditional Access or a third-party network access control solution.
- Workload owners are given a fixed period of time to test the updates, apply them to production systems, and reboot as required. After that self-service period, and any grace period expires, enterprise IT and security teams will either force the update to apply or isolate them from other corporate resources. Some organizations with stringent requirements might decommission the assets by deleting them from Azure subscriptions or AWS accounts.
- Enterprise IT and security teams monitor the status of the update and conduct any forced remediation that's identified.

This process is not static and it's not set up in one day. It's built iteratively and continuously improved over time.

Start wherever you are and continually improve the process to make incremental progress toward this end state. Use the following dimensions for continuous improvement planning:

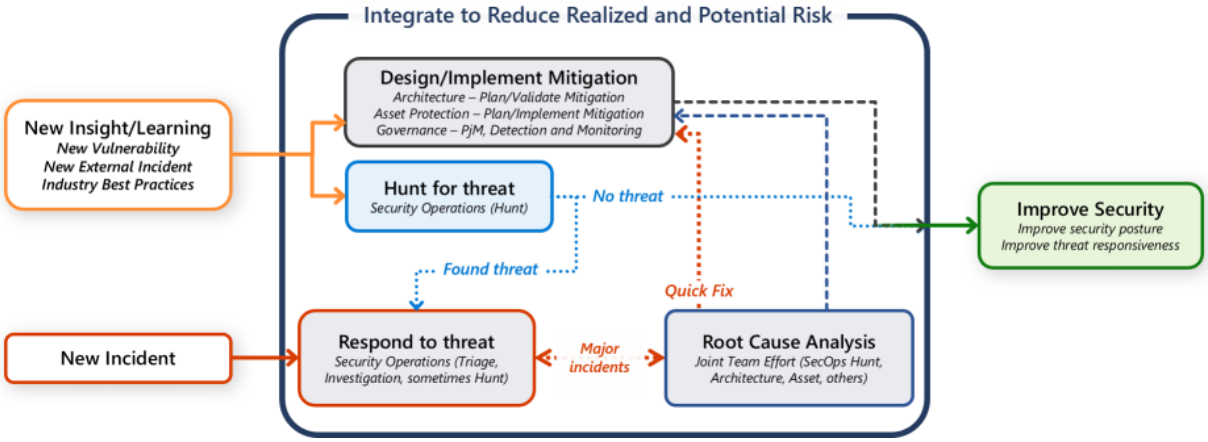
- **Coverage:** Start with a few application teams that have a high chance of success or have a large business impact if compromised. Add more until you cover all workloads in your environment.
- **Time:** Start with deadlines that you know you can accomplish and set a clear roadmap to continually shorten them until you are close to fully updated in a week or less.
- Technology scope. Continuously improve what patches and technologies you cover, including applications, middleware, and open-source components used in application code. You should encourage the use of components that are updated for you to lower the maintenance burden. For example, use Azure SQL Database instead of installing and updating your own SQL Server.
- **Processes:** Continually improve communication channels between teams, prioritization guidance, exception processes, and all other aspects of this process.

Integrate security teams

Security teams should work together and collaborate to avoid increased business risk resulting from operating in silos. When learnings and key insights aren't shared across security teams, the organization might experience greater damage and impact from a future incident that could have been avoided.

Security is a dynamic discipline that must always be responding to active threats and always be learning and continuously improving processes, tools, and technologies. Security must constantly adapt to the shifts in attacker techniques, technology platforms, and business models of their organization. Security teams should work together to rapidly respond to threats and continuously integrate insights and learnings into processes that both improve the security posture of the organization, and the ability to rapidly respond to attacks.

The following workflow diagram illustrates how security disciplines should collaborate to completely integrate learnings and insights to improve security overall.



Security's primary mission is to rapidly respond to the following situations:

- **New incidents:** Active attackers with access to organizational resources pose an immediate risk to the organization that must be rapidly remediated as a first priority. After remediation, these attacks represent the best opportunity to learn what future attacks will look like. Whether they succeeded or failed, attackers are likely to repeat pursuing the same target, technique, or monetization model again.
- **New insights and learnings:** New insights and learnings can come from the following sources:
  - External incidents. Incidents at other organizations can offer insights on attackers. They might attempt the same on your organization. This knowledge informs improvement plans or validates that your investments are on the right track. Discover external incidents through an information sharing and analysis center (ISAC), direct relationships with peer organizations, or other public reporting and analysis on incidents.

    - New technical capabilities. Cloud providers and software vendors continuously innovate. They add capabilities to their products:
        - Business capabilities that require security defenses.
        - Security capabilities that improve the ability of security to defend the assets. These capabilities might be native security capabilities integrated into cloud platforms or other platform technology. They might be traditional stand-alone security capabilities.
        - The visibility and telemetry available from cloud-based security far outweighs what organizations could get from their single on-premises environment. All of this data is gathered by using metadata from all over. The data is put through a rigorous analysis process, including behavioral analytics, detonation chambers, machine learning, and AI.
    - **Industry best practices:** Industry best practices from vendors and organizations like National Institute of Standards and Technology (NIST), Center for Internet Security (CIS), and The Open Group. These organizations have a charter to gather and share learnings and best practices that security teams can learn from.
    - Vulnerabilities are anything that an attacker can exploit to take control of an asset, such as software vulnerabilities. There are also security configuration choices, weaknesses in cryptographic algorithms, insecure practices, and processes for using or managing systems. As you discover vulnerabilities, evaluate them for how they affect your security posture and ability to detect, respond to, and recover from an attack.
- **Respond to threat:** Security operations teams investigate detections. They respond to them by evicting adversaries from their control points in the organization. Depending on the size of the organization and complexity of the incident, this response might involve several security teams.
- **Root cause analysis:** Identifying the key contributing factors that made a major incident higher likelihood or higher impact generates insights learnings that can improve the organizations security posture and ability to respond. These learnings can happen along many dimensions including attack tools and infrastructure, attack techniques, targets, motivation, and monetization models. Root cause analysis can inform preventive controls, detective controls, security operations processes, or any other element of the security program or architecture.
- **Hunt for threat:** Hunting proactively for threats is an ongoing activity. Hunting should always consider new insights or learnings in the hunt planning and hypothesis development. Hunt teams might want to focus on key aspects:
    - Recently widespread or high impact vulnerability.
    - A new attacker group.
    - A new attack technique that was demonstrated at a conference.
- **Design and implement mitigation:** Lessons learned must be integrated into the technical environment and both security and business processes. Teams should work together to integrate lessons into architecture, policy, and standards. For example, the theft of administrative credentials in a recent internal or public incident might spur the organization to adopt the controls in Microsoft's privilege access. For more information, see Security rapid modernization plan.

Next steps

As you plan your cloud adoption, focus on integrating security functions together. Integrate security with your larger organization. Pay close attention to the friction that security generates. Be sure the friction is healthy. Healthy friction reduces risk for the organization without creating slowdowns that erode more value than they protect.

- Review define a security strategy

- Review cloud security functions
- Review business resilience, which is the next focus area of security.

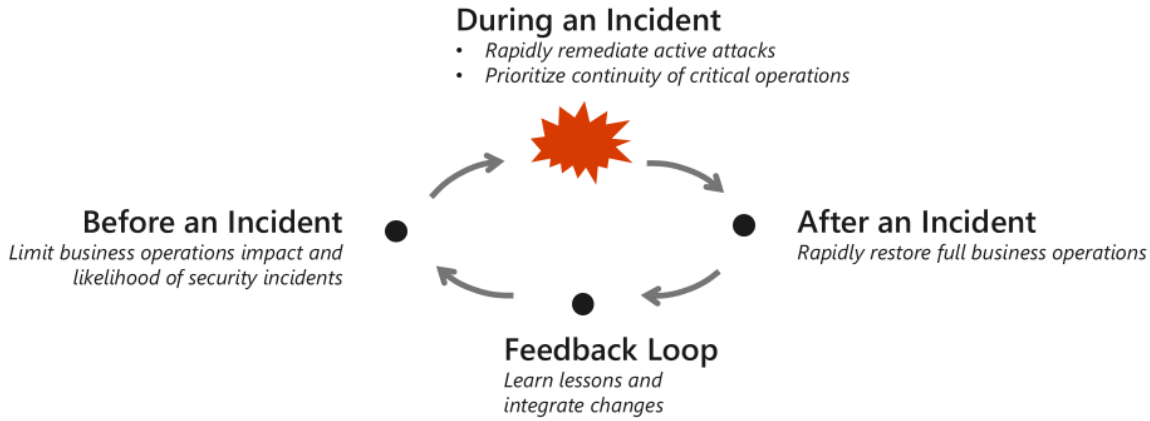
Business resilience

5/24/2021 • 3 minutes to read • Edit Online

An organization can never have perfect security, but it can become resilient to security attacks. Like we are never perfectly immune to all health and safety risks in the physical world, the data and information systems we operate are also never 100 percent safe from all attacks all the time.

Focus your security efforts on increasing the resilience of business operations in the face of security incidents. These efforts can reduce risk and enable continuous improvements in security posture and your ability to respond to incidents.

Resilience requires taking a pragmatic view that assumes a breach. It needs continuous investment across the full lifecycle of security risk.



- **Before an incident:** Continuously improve security posture and the organization's ability to respond to an incident. Continuously improving security posture helps limit the likelihood and potential impact of a security incident on your business operations and assets. There are many techniques covered throughout the security disciplines, but all of them are designed to raise the cost of attack. Make the attackers develop and try new techniques because you've made their old ones stop working. These techniques raise their costs and friction, slowing them down and limiting their success.
- **During an incident:** Business operations must continue during an incident, even if they're degraded, slower, or limited to only critical systems. During an attack, a hospital can't stop caring for patients, ships need to move cargo, and planes must continue to fly safely. The two main priorities during an incident are:
  - **Protect critical operations:** If threatened, all efforts should focus on protecting and sustaining critical business operations over all other functions.
  - **Prevent further damage:** The default priority for security operations is to discover the full extent of attacker access, and then rapidly evict them from the environment. This eviction prevents further damage by the attackers, including retaliation.
- **After an incident:** If business operations are damaged during an attack, repairs must start immediately to restore full business operations. These repairs apply even if that means restoring operations without data that was lost in an attack, for example ransomware or a destructive attack like NotPetya.
- **Feedback loop:** Attackers repeat themselves, or repeat what other attackers have found to work. Attackers learn from attacking your organization, so you must continuously learn from their attacks. Focus on taking the proven and available techniques that were attempted before. Then ensure you can block, detect, rapidly respond, and recover from them. Your efforts raise the cost of attack on your

organization and deter or slow down future attacks.

Resilience goals

Security resiliency is focused on supporting the resiliency of your business.

- **Enable your business** to rapidly innovate and adapt to the ever changing business environment. Security should always be seeking safe ways to say *yes* to business innovation and technology adoption. Your organization can then adapt to unexpected changes in the business environment, like the sudden shift to working from home during COVID-19.
- **Limit the impact** and likelihood of disruptions before, during, and after active attacks to business operations.

Security resilience and assume breach

Good security starts with assuming breach or assuming compromise. Understand that information technology (IT) systems operate over the open internet where criminals are constantly attacking and compromising organizations. This assumption is a key principle of zero trust, which drives healthy and pragmatic security behaviors. Zero trust prepares your organization to prevent attacks, limit their damage, and quickly recover from them.

> **NOTE**
> The *assume breach* principle doesn't assume that every compromise is a reportable event to regulators. The principle assumes that some level of security defenses have been breached or compromised. It then drives a focus on making the organization resilient through improvements to security posture and response processes.

Assuming compromise should drive changes across people, process, and technology in your organization. Education, exercises, and other incident preparation activities are good examples. For more information, see Incident preparation.

Assuming compromise is a significant shift from the classic security mindset. That mindset assumed organizations could establish a perfectly safe network protected by a security perimeter, restricting all valuable business activity to that network. Elements of this strategy provide value in a resilient enterprise. But the assumption of a perfectly safe network didn't in the world of mobile work and evolving business models, technology platforms, and security threats.

Resiliency requires a close partnership between business stakeholders, IT professionals and leaders, and security professionals.

Next steps

The security disciplines in the Secure methodology align to the goal of increasing resilience to security incidents. For more information, see:

- Access control
- Security operations
- Asset protection
- Security governance
- Innovation security

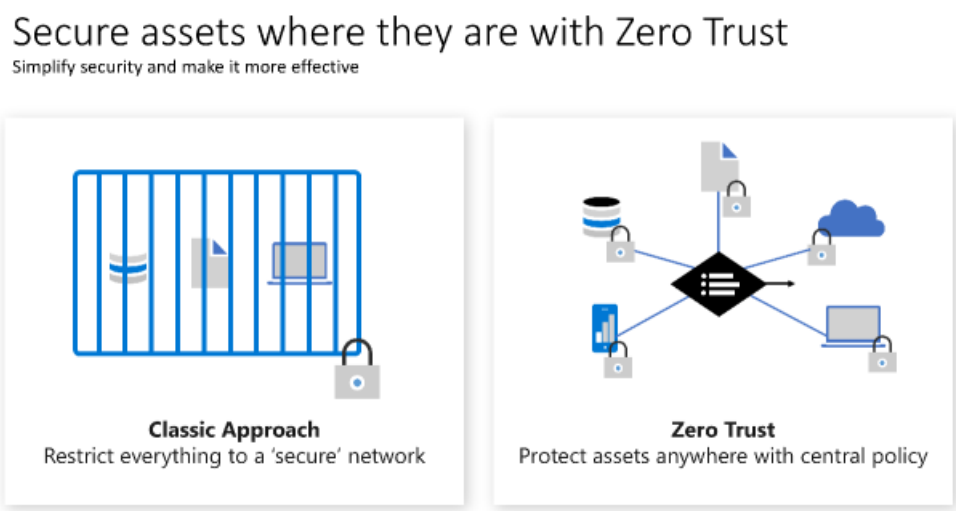
Access control

6/8/2021 • 7 minutes to read • Edit Online

Access control is the part of security that people experience first and most often. They see it when they sign in to their computers and mobile phones, when they share a file or try to access an application, and when they use an ID card key to enter a building or room. While access control isn't everything in security, it's critically important, and it requires proper attention so that both the user experience and the security assurances are right.

From security perimeter to zero trust

The traditional approach of access control for IT has been based on restricting access to a corporate network and then supplementing it with more controls as appropriate. This model restricts all resources to a corporate owned network connection and has become too restrictive to meet the needs of a dynamic enterprise.



Organizations must embrace a zero trust approach to access control as they embrace remote work and use cloud technology to digitally transform their business model, customer engagement model, employee engagement, and empowerment model.

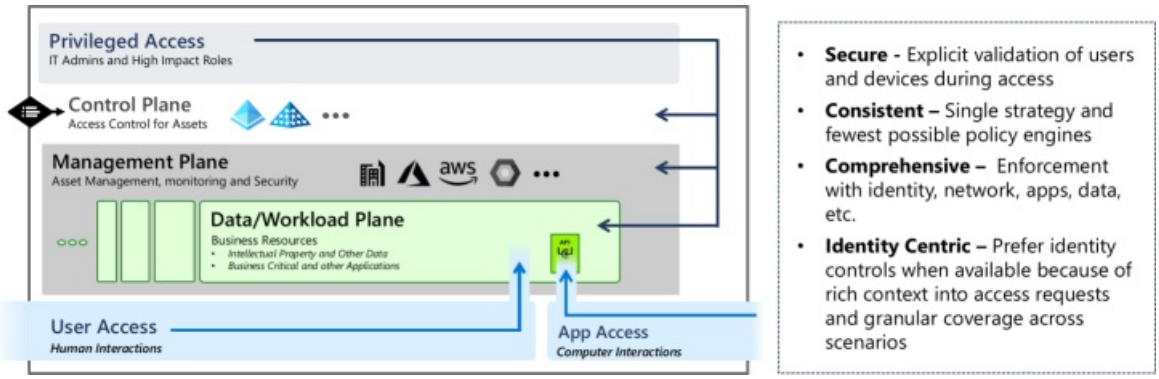
Zero trust principles help establish and continuously improve security assurances, while maintaining flexibility to keep pace with this new world. Most zero trust journeys start with access control and focus on identity as a preferred and primary control while they continue to embrace network security technology as a key element. Network technology and the security perimeter tactic are still present in a modern access control model, but they aren't the dominant and preferred approach in a complete access control strategy.

Modern access control

Organizations should develop an access control strategy that:

- Is comprehensive and consistent.
- Rigorously applies security principles throughout the technology stack.
- Is flexible enough to meet the needs of the organization.

This diagram illustrates all of the different elements that an organization must consider for an access control strategy for multiple workloads, multiple clouds, various business sensitivity levels, and access by both people and devices.



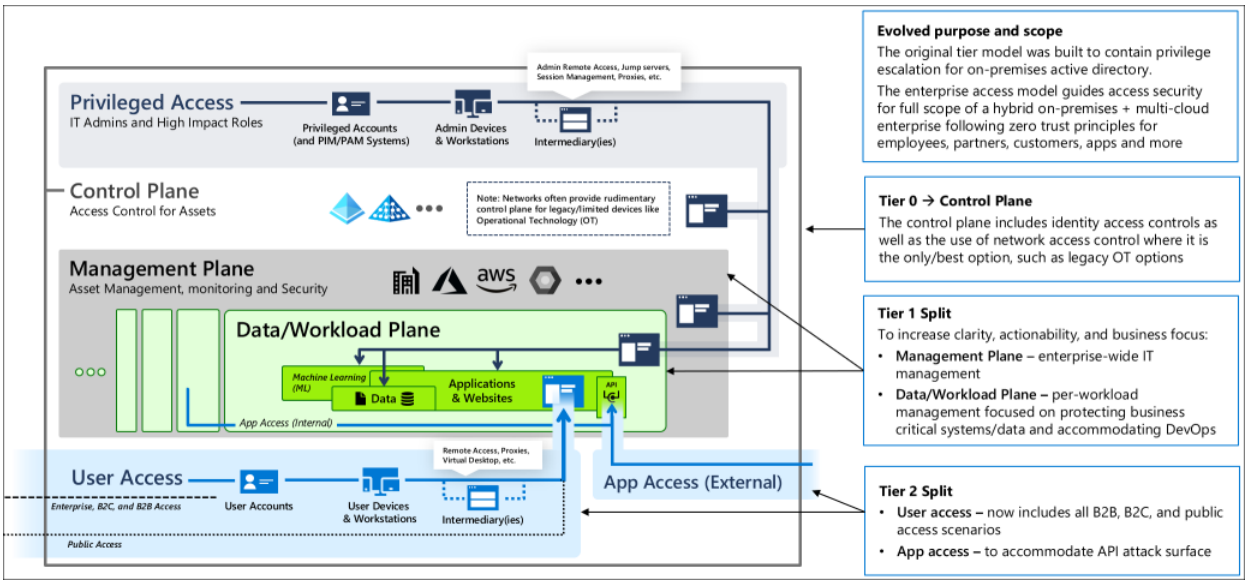
A good access control strategy goes beyond a single tactic or technology. It requires a pragmatic approach that embraces the right technology and tactics for each scenario.

Modern access control must meet the productivity needs of the organization, and also be:

- **Secure:** Explicitly validate the trust of users and devices during access requests, using all available data and telemetry. This configuration makes it more difficult for attackers to impersonate legitimate users without being detected. Also, the access control strategy should focus on eliminating unauthorized escalation of privilege, for example, granting a privilege that can be used to get higher privileges. For more information on protecting privileged access, see Securing privileged access.
- **Consistent:** Ensure that security assurances are applied consistently and seamlessly across the environment. This standard improves the user experience and removes opportunities for attackers to sneak in through weaknesses in a disjointed or highly complex access control implementation. You should have a single access control strategy that uses the fewest number of policy engines to avoid configuration inconsistencies and configuration drift.
- **Comprehensive:** Enforcement of access policy should be done as closely to the resources and access pathways as possible. This configuration improves security coverage, and helps security fit smoothly into scenarios and expectations of users. Take advantage of security controls for data, applications, identity, networks, and databases to drive policy enforcement closer to the business assets of value.
- **Identity-centric:** Prioritize the use of identity and related controls when available. Identity controls provide rich context into access requests, and application context that isn't available from raw network traffic. Networking controls are still important, and sometimes the only available option (such as in operational technology environments), but identity should always be the first choice if available. A failure dialog during application access from the identity layer will be more precise and informative than a network traffic block, making it more likely the user can correct the issue without a costly help desk call.

Enterprise access model

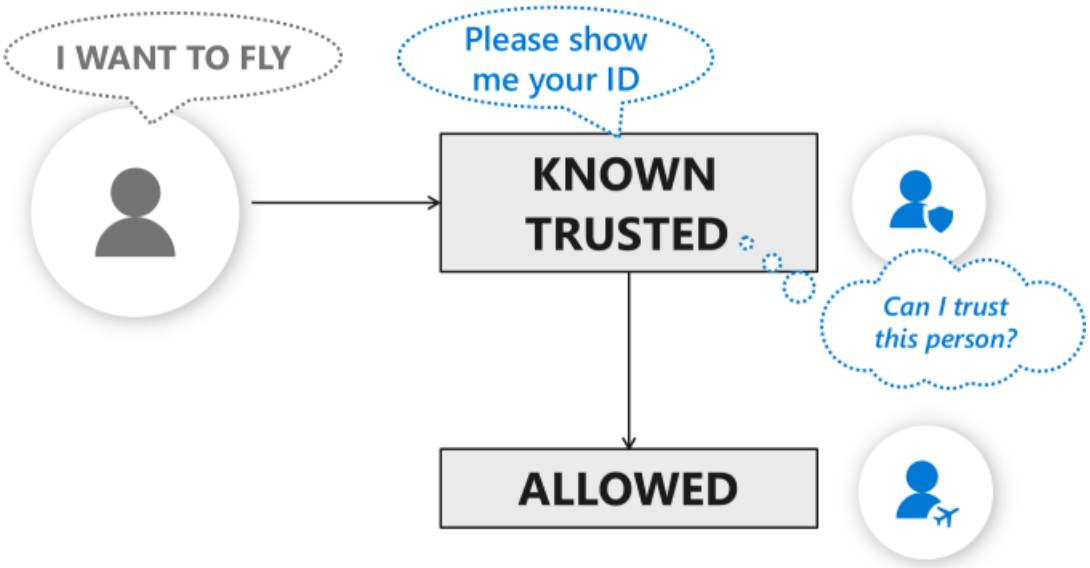
The enterprise access model is a comprehensive access model based on zero trust. This model addresses all types of access by internal and external users, services, applications, and privileged accounts with administrative access to systems.



The enterprise access model is described in detail in Enterprise access model

Known, trusted, allowed

One helpful perspective on the zero trust transformation of access control is that it shifts from a static two-step process of authentication and authorization, to a dynamic three-step process called *known, trusted, allowed*:



1. **Known:** Authentication that ensures you are who you say you are. This process is analogous to the physical process of checking a government-issued photo identification document.
2. **Trusted:** Validation that the user or device is trustworthy enough to access the resource. This process is analogous to security at an airport that screens all passengers for security risks before allowing them to enter the airport.
3. **Allowed:** Granting of specific rights and privileges for the application, service, or data. This process is analogous to an airline that manages where passengers are going, what cabin they sit in (first class, business class, or coach), and whether they have to pay for luggage.

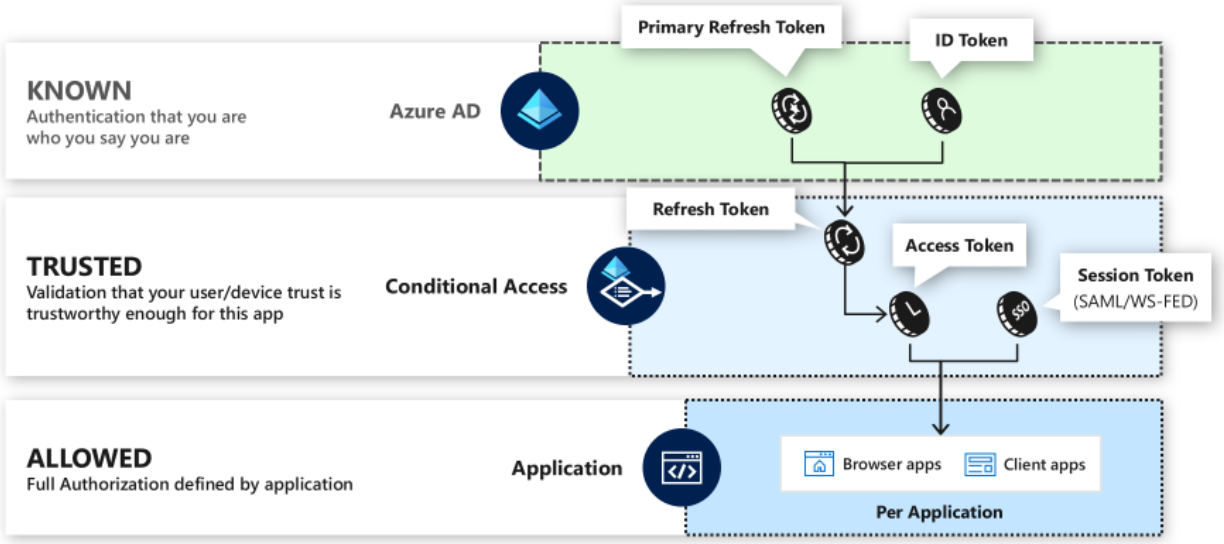
Key access control technologies

The key technical capabilities that enable modern access control are:

- **Policy engine:** The component where organizations configure the technical security policy to meet the organization's productivity and security objectives.
- **Policy enforcement points:** Points that enforce, across the organization's resources, the central policy

decisions by the policy engine. The resources include data, applications, identity, network, and databases.

This diagram depicts how Azure Active Directory provides a policy engine and a policy enforcement point, so that security protocols can implement a *known, trusted, allowed* approach.



The Azure Active Directory policy engine can be extended to other policy enforcement points, including:

- **Modern applications:** Applications that use modern authentication protocols.
- **Legacy applications:** Via Azure AD Application Proxy.
- **VPN and remote access solutions:** Such as Cisco AnyConnect, Palo Alto Networks, F5, Fortinet, Citrix, and Zscaler.
- **Documents, email, and other files:** Via Microsoft Information Protection.
- **SaaS applications:** For more information, see Tutorials for integrating SaaS applications with Azure Active Directory.

Data-driven access decisions

To fulfill the zero trust principle of explicit validation, it's critical to make an informed decision. The zero trust policy engine should have access to diverse data on the users and devices in order to make sound security decisions. This diversity helps to identify these aspects with greater confidence:

- Whether the actual user is in control of the account.
- Whether the device has been compromised by an attacker.
- Whether the user has the appropriate roles and permissions.

Microsoft built a threat intelligence system that integrates security context from many and diverse signal sources. For more information, see Summary of Microsoft threat intelligence.

Segmentation: Separate to protect

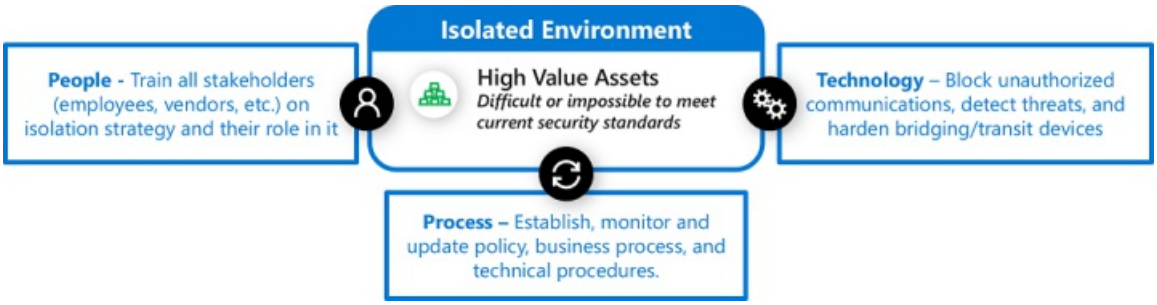
Organizations often choose to create boundaries to divide the internal environment into separate segments, as part of their access control approach. This configuration is intended to contain the damage of a successful attack to the attacked segment. Segmentation is traditionally done with firewalls or other network filtering technology, though the concept can also be applied to identity and other technologies.

For information on applying segmentation to Azure environments, see Azure components and reference model

Isolation: Avoid firewall and forget

Isolation is an extreme form of segmentation that is sometimes required for protecting critically important assets. Isolation is most often used for assets that are both business-critical, and difficult to bring up to current policy and standards. Classes of assets that might require isolation include operational technology (OT) systems like:

- Supervisory control and data acquisition (SCADA)
- Industrial control system (ICS)



Isolation must be designed as a complete people/process/technology system and be integrated with business processes to be successful and sustainable. This approach typically fails over time if it's implemented as a purely technology approach without processes and training to validate and sustain the defenses. It's easy to fall into a *firewall and forget* trap by defining the problem as static and technical.

In most cases, processes are needed to implement isolation, processes that various teams like security, IT, operational technology (OT), and sometimes business operations must follow. Successful isolation usually consists of:

- **People:** Train all employees, vendors, and stakeholders on isolation strategy and their part in it. Include why it's important, for example, threats, risks and potential business impact, what they're expected to do, and how to do it.
- **Process:** Establish clear policy and standards and document processes for business and technical stakeholders for all scenarios such as vendor access, change management process, threat response procedures, including exception management. Monitor to ensure the configuration doesn't drift and that other processes are followed correctly and rigorously.
- **Technology:** Implement technical controls to block unauthorized communications, detect anomalies and potential threats, and harden bridging and transit devices that interact with the isolated environment, for example, operator consoles for operational technology (OT) systems.

For more information, see Asset protection: network isolation.

Next steps

Whatever your requirements for access control, your strategy should be grounded in the proper fundamentals. The guidance provided in these articles, and in the Cloud Adoption Framework, can help organizations find and implement the right approach.

The next discipline is modernizing security operations.

Security operations

6/8/2021 • 8 minutes to read • Edit Online

This article provides strategic guidance for leaders establishing or modernizing a security operation function. For architecture and technology focused best practices, see Best practices for security operations.

Security operations reduce risk by limiting damage from attackers who gain access to your organization's resources. Security operations focus on reducing the time attackers have access to resources by detecting, responding to, and helping recover from active attacks.

Rapid response and recovery protect your organization by damaging the adversary's return on investment (ROI). When adversaries are evicted and forced to start a new attack, their cost to attack your organization goes up.

Security operations (SecOps) is sometimes referred to or structured as a security operations center (SOC). Security posture management of the operational environment is a function of governance discipline. Security of the DevOps process is part of the innovation security discipline.

People and process

Security operations can be highly technical, but more importantly, it's a human discipline. People are the most valuable asset in security operations. Their experience, skill, insight, creativity, and resourcefulness are what make the discipline effective.

Attacks on your organization are also planned and conducted by people like criminals, spies, and hacktivists. While some commodity attacks are fully automated, the most damaging ones are often done by live human attack operators.

Focus on empowering people: your goal shouldn't be to replace people with automation. Empower your people with tools that simplify their daily workflows. These tools enable them to keep up with or get ahead of the human adversaries they face.

Rapidly sorting out signal (real detections) from the noise (false positives) requires investing in both humans and automation. Automation and technology can reduce human work, but attackers are human and human judgment is critical in defeating them.

Diversify your thinking portfolio: security operations can be highly technical, but it's also just another new version of forensic investigation that shows up in many career fields like criminal justice. Don't be afraid to hire people with a strong competency in investigation or deductive or inductive reasons and train them on technology.

Make sure that your people are set up with a healthy culture and are measuring the right outcomes. These practices can increase productivity and employee enjoyment of their jobs.

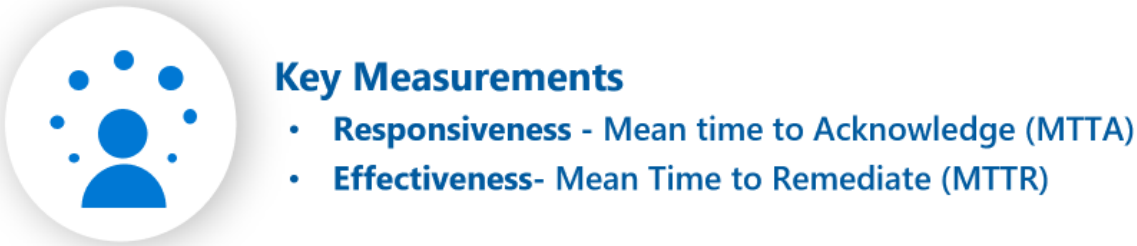
SecOps culture



Key cultural elements to focus on include:

- **Mission alignment:** Because of how challenging this work is, security operations should always have a clear understanding of how their work connects to the overall organization's mission and goals.
- **Continuous learning:** Security operations are highly detailed work and always changing because attackers are creative and persistent. It's critical to continuously learn and work to automate highly repetitive or highly manual tasks. These types of tasks can quickly wear down morale and team effectiveness. Ensure that the culture rewards learning, finding, and fixing these pain points.
- **Teamwork:** We've learned that the "lone hero" doesn't work in security operations. Nobody is as smart as the whole team together. Teamwork also makes a high-pressure working environment more enjoyable and productive. It's important that everyone has each other's back. Share insights, coordinate and check each other's work, and constantly learn from each other.

SecOps metrics



Metrics drive behavior, so measuring success is a critical element to get right. Metrics translate culture into clear measurable goals that drive outcomes.

We've learned that it's critical to consider what you measure, and the ways that you focus on and enforce those metrics. Recognize that security operations must manage significant variables that are out of their direct control, like attacks and attackers. Any deviations from targets should be viewed primarily as a learning opportunity for process or tool improvement, rather than assumed to be a failure by the SOC to meet a goal.

The main metrics to focus on that have a direct influence on organizational risk are:

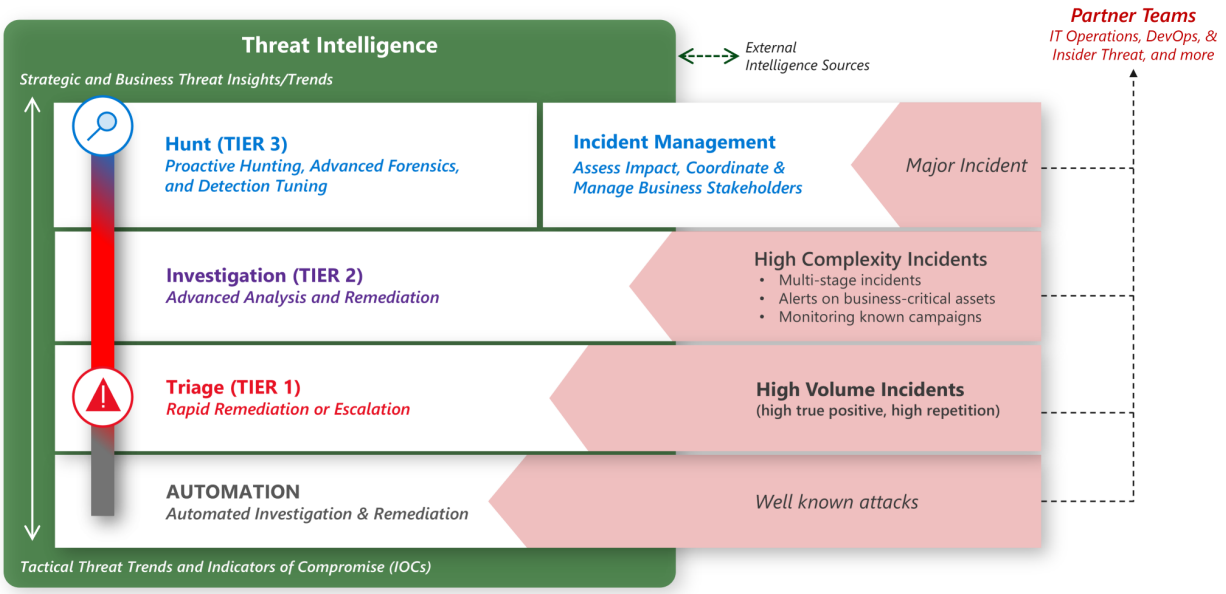
- **Mean time to acknowledge (MTTA):** Responsiveness is one of the few elements SecOps has more direct control over. Measure the time between an alert, like when the *light starts to blink*, and when an analyst sees that alert and begins the investigation. Improving this responsiveness requires that analysts don't waste time investigating false positives. It can be achieved with ruthless prioritization, ensuring that any alert feed that requires an analyst response must have a track record of 90 percent true positive detections.
- **Mean time to remediate (MTTR):** Effectiveness of reducing risk measures the next period of time. That period is the time the analyst begins the investigation to when the incident is remediated. MTTR identifies how long it takes SecOps to remove the attacker's access from the environment. This information helps identify where to invest in processes and tools to help analysts reduce risk.
- **Incidents remediated (manually or with automation):** Measuring how many incidents are remediated

manually and how many are resolved with automation is another key way to inform staffing and tool decisions.

- **Escalations between each tier:** Track how many incidents escalated between tiers. It helps ensure accurate tracking of the workload to inform staffing and other decisions. For example, so that work done on escalated incidents isn't attributed to the wrong team.

Security operations model

Security operations handles a combination of high volume incidents and high complexity incidents.



Security operations teams often focus on three key outcomes:

- **Incident management:** Manage active attacks on the environment, including:
  - **Reactively responding** to detected attacks.
  - **Proactively hunting** for attacks that slipped through traditional threat detections.
  - **Coordinating** the legal, communications, and other business implications of security incidents.
- **Incident preparation:** Help the organization prepare for future attacks. Incident preparation is a broader strategic set of activities that are aimed at building muscle memory and context at all levels of the organization. This strategy prepares people to better handle major attacks and gain insights on security process improvements.
- **Threat intelligence:** Gathering, processing, and disseminating threat intelligence to security operations, security teams, security leadership, and business leadership stakeholders through security leadership.

To deliver against these outcomes, security operations teams should be structured to focus on key outcomes. In larger SecOps teams, the outcomes are often broken out between subteams.

- **Triage (tier 1):** The first line of response for security incidents. Triage is focused on the high-volume processing of alerts, and typically generated by automation and tools. The triage processes solve for most of common incident types and resolve them within the team. More complex incidents, or incidents that haven't been seen and resolved before should be escalated to tier 2.
- **Investigation (tier 2):** Focused on incidents that require further investigation, often requiring correlation of data points from multiple sources. This investigation tier looks to provide repeatable solutions to issues escalated to them. It then enables tier 1 to resolve later recurrences of that issue type. Tier 2 will also respond to alerts against business critical systems, to reflect the severity of the risk and the need to act quickly.
- **Hunt (tier 3):** Focused primarily on proactive hunting for highly sophisticated attack processes and developing guidance to the wider teams for maturing security controls as a result. The tier 3 team also acts as an escalation point for major incidents in to support forensic analysis and response.

SecOps business touchpoints

SecOps has multiple potential interactions with business leadership.



- **Business context to SecOps:** SecOps must understand what is most important to the organization so that the team can apply that context to fluid real-time security situations. What would have the most negative impact on the business? Downtime of critical systems? A loss of reputation and customer trust? Disclosure of sensitive data? Tampering with critical data or systems? We've learned it's critical that key leaders and staff in the SOC understand this context. They'll wade through the continuous flood of information and triage incidents and prioritize their time, attention, and effort.
- **Joint practice exercises with SecOps:** Business leaders should regularly join SecOps in practicing response to major incidents. This training builds the muscle memory and relationships that are critical to fast and effective decision making in the high pressure of real incidents, reducing organizational risk. This practice also reduces risk by exposing gaps and assumptions in the process that can be fixed before a real incident occurs.
- **Major incidents updates from SecOps:** SecOps should provide updates to business stakeholders for major incidents as they happen. This information allows business leaders to understand their risk and take both proactive and reactive steps to manage that risk. For more information on major incidents by the Microsoft Detection and Response Team, see the incident response reference guide.
- **Business intelligence from the SOC:** Sometimes SecOps finds that adversaries are targeting a system or data set that isn't expected. As these discoveries are made, the threat intelligence team should share these signals with business leaders as they might trigger insight for business leaders. For example, someone outside the company is aware of a secret project or unexpected attacker targets highlight the value of an otherwise overlooked dataset.

SecOps modernization

Just like other security disciplines, security operations face the transformative effect of continuously evolving business models, attackers, and technology platforms.

The transformation of security operations is driven primarily by the following trends:

- **Cloud platform coverage:** Security operations must detect and respond to attacks across the enterprise estate including cloud resources. Cloud resources are a new and rapidly evolving platform that is often unfamiliar to SecOps professionals.
- **Shift to identity-centric security:** Traditional SecOps relies heavily on network-based tools, but must now integrate identity, endpoint, application, and other tools and skills. This integration is because:
  - Attackers have incorporated identity attacks, like phishing, credential theft, password spray, and other attack types into their arsenal that reliably evade network-based detections.

    - Assets of value, like bring your own devices (BYOD), spend some or all of their lifecycle outside the network perimeter, limiting the utility of network detections.
- **Internet of Things (IoT) and operational technology (OT) coverage:** Adversaries actively target IoT and OT devices as part of their attack chains. These targets might be the ultimate purpose of an attack or a means to access or traverse the environment.
- **Cloud processing of telemetry:** Security operations modernization is required because of the massive increase in relevant telemetry that comes from the cloud. This telemetry is difficult or impossible to process with on-premises resources and classic techniques. It then drives SecOps to embrace cloud services that provide massive scale analytics, machine learning, and behavior analytics. These technologies help rapidly extract the value to meet the time sensitive needs of security operations.

It's important to invest in updated SecOps tools and training to ensure security operations can meet these challenges. For more information, see Update incident response processes for cloud.

For more information on security operations roles and responsibilities, see Security operations.

For more architecture and technology focused best practices, see Microsoft security best practices for security operations and the videos and slides.

Next steps

The next discipline is asset protection.

Asset protection

5/24/2021 • 9 minutes to read • Edit Online

Assets include physical and virtual items such as laptops, databases, files, and virtual storage accounts. Securing business critical assets often relies on the security of underlying systems, like storage, data, endpoint devices, and application components. The most valuable technical assets are typically data and the availability of applications, such as business websites, production lines, and communications.

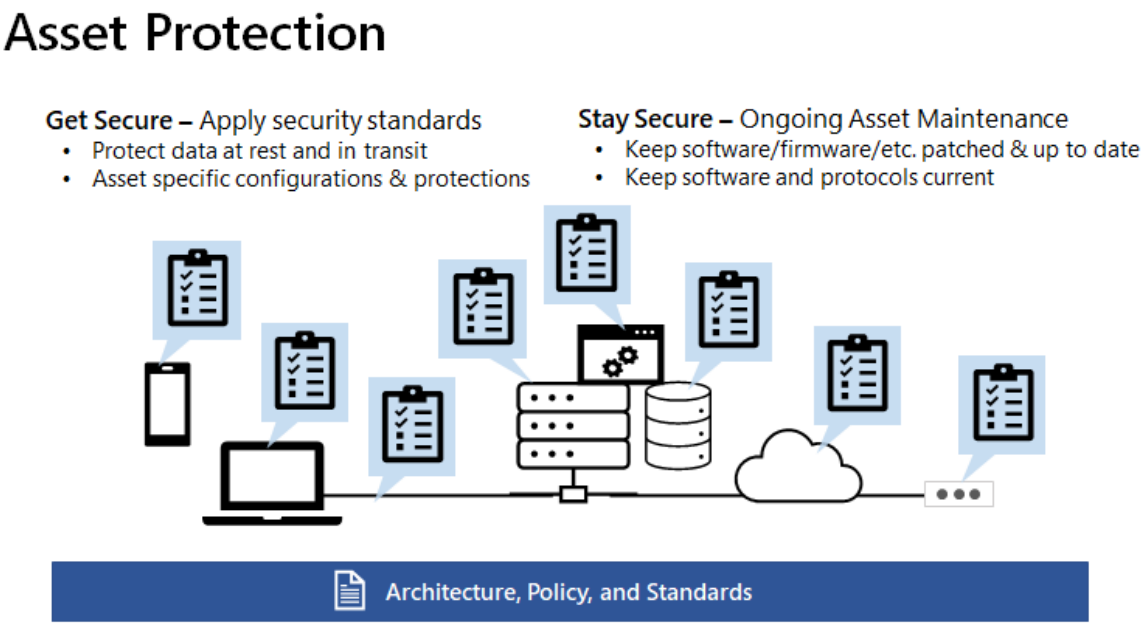
Asset protection implements controls to support security architecture, standards, and policy. Each asset type and security requirement is unique. The security standards for any asset type should be consistently applied to all instances.

Asset protection focuses on consistent execution across all control types. Preventive, detective, and others align to meet the policies, standards, and architecture.

Asset protection acts as the technical subject matter expert for assets. It works with other disciplines, such as governance, architecture, security operations, and workload teams. Asset protection ensures that policy and standards are feasible and it enables implementation of controls to support the policy and standards. Asset protection provides feedback for continuous improvement.

> **NOTE**
> Asset protection is usually implemented by IT operations teams that maintain the assets and are supplemented by expertise in the security team. For more information, see Design controls as a team.

Threat actors are persistent and seek out vulnerabilities that result from gaps in the application of standards and policy. Attackers can directly target the business-critical data or application. They can also target the infrastructure that grants them access to the business-critical data and applications. Access control focuses on managing authorized access to resources. Asset protection addresses on all other potential out of band ways to gain access or control of resources. These two disciplines compliment each other and should be designed together to meet your architecture, policies, and standards. For more information, see Access control.



Get secure

Get secure focuses on bringing resources up to meet your organization's current security standards, policy, and architecture. There are two types of activities:

- **Brownfield:** Retrofit current security standards and controls to existing assets. Organizations might design and operate IT environments with security as low priority. This approach creates a "technical debt": weak security configurations, software that isn't upgraded, unencrypted communication or storage, legacy software and protocols, and more. Bring your security controls up to current approach. This improvement is critical to mitigate risk because attackers continually improve their ability to exploit these opportunities.
- **Greenfield:** Ensure that new assets and new asset types are configured to standards. This process is critical to avoid continuously creating instant legacy or brownfield, or systems that don't meet current standards. This technical debt will have to be addressed later at a greater expense, which results in increased risk exposure until it's complete.

Financially, *get secure* typically maps to capital expenditures (CAPEX) dynamics of a one-time investment. *Greenfield budget* for security should be mapped as closely as feasible to the creation of the asset, with a reserved percentage of the budget for security for each new software project, major software upgrade, or overall cloud adoption initiative. Many organizations reserve around 10 percent of budget for security. *Brownfield budget* is typically a special project funded to bring security controls up to current standards and compliance.

Stay secure

Everything degrades over time. Physical items wear out. The environment changes around virtual items like software, security controls, and security. They might no longer meet changing requirements. These shifts happen fast today because of the following rapid changes:

- **Business requirements**, driven by digital transformation.
- **Technology requirements**, driven by rapid cloud platform evolution and feature releases.
- **Security requirements**, driven by attacker innovation and rapid evolution of native cloud security capabilities.

This dynamic affects all parts of security, including security operations, access control, and particularly DevSecOps in Innovation security.

Staying secure includes many elements. Focus on these two specific areas of asset protection:

- **Continuous cloud improvement:** Embrace the continuous improvement in security capabilities that the cloud brings. For example, many services in Azure like Azure Storage and Azure SQL Database have added security features to defend against attackers over time.
- **Software end-of-life:** Any software, including operating systems, always reach end-of-life, when security updates are no longer provided. This situation can expose business critical data and applications to cheap and easy attacks. While software as a service (SaaS) and cloud infrastructure and platforms are maintained by the cloud provider, enterprises often have a significant amount of software they install, author, and must maintain.

Plan to upgrade or retire end-of-life software. Investing into your security posture reduces the risk of a major security incident. *Stay secure* is part of the operational expenditures (OPEX) dynamics of a regular ongoing investment.

The patch dilemma

It's critical for business leaders to support their IT and security leaders and teams. Running complex software in a hostile environment has inherent risk. Security and IT leaders constantly make difficult decisions about operational risk and security risk.

- **Operational risk:** A change to the software where the system runs could disrupt business processes. Such changes affect the assumptions made when the system was customized for the organization. This fact creates pressure to avoiding changing the system.
- **Security risk:** An attack brings the business risk of downtime. Attackers analyze every major security update on release. They can develop a working exploit in 24-48 hours to attack organizations that haven't applied the security update.

Your organization could have this dilemma often because of the continuing changes to technology and evolution of attack technique. Business leaders must recognize the risk of running a business using complex software. Support updating business processes such as these examples:

- **Integrating software maintenance** into the business operational assumptions, schedule, forecasting, and other business processes.
- **Investing in architectures** that make maintenance easier and reduce impact on business operations. This approach could involve updating existing architectures or shifting to new architectures entirely by migrating to cloud services or a service-oriented architecture.

Without business leadership support, security and IT leaders are distracted from supporting important business goals. They must constantly manage the politics of a no-win situation.

Network isolation

Network isolation might be a valid option for protecting older assets that can no longer be secured, but cannot be immediately retired. This scenario can typically occur for end of life operating systems and applications. It's common in operational technology (OT) environments and legacy systems.

Isolation itself is considered access control, even though the assets that can't be secured are identified as part of asset protection. For more information, see Avoid firewall and forget.

Some systems at end of life are difficult to disconnect and isolate completely. We don't recommend leaving these insecure systems fully connected to a production network. This configuration can allow attackers to compromise the system and gain access to assets in the organization.

It's never cheap or easy to upgrade or replace computer technology that has been working well for a decade or more. There might be limited documentation on its functionality. The potential business impact of losing control of multiple business critical assets often exceeds the cost of upgrade or replacement. For these assets that cannot be isolated, organizations often find that modernizing the workload with cloud technology and analytics can create new business value that can offset or justify the cost of upgrade or replacement.

Staying secure is challenging in a world that is constantly changing. It's critical to constantly decide what assets to modernize and what to secure as best you can. Use business risk and business priorities to evaluate.

Getting started

To get started with asset protection, we recommend that organizations take the following steps.

- **Focus on well-known resources first:** Think about virtual machines, networks, and identities in the cloud that the team is already familiar with. This technique allows you to make immediate progress and are often easier to manage and secure with native cloud tools like Azure Security Center and Azure Defender.
- **Start with vendor/industry baselines:** Start your security configuration with a well-known and proven solution, for example:
    - **Security baselines in the Azure Security Benchmark.** Microsoft provides security configuration guidance tailored to individual Azure services. These baselines apply the Azure security benchmarks to the unique attributes of each service. This approach enables security teams to secure each service and

refine configurations as needed. For more information, see Security baselines for Azure.

- **Microsoft security baselines.** Microsoft provides security configuration guidance for commonly used technologies including Windows, Microsoft Office, and Microsoft Edge. For more information, see Microsoft security baselines.Microsoft-security-baselines) for more information
- **CIS benchmarks.** The Center for Internet Security (CIS) provides specific configuration guidance for many products and vendors. For more information, see CIS benchmarks.

Key information

These key element help guide your asset protection process:

Accountable and responsible teams

Accountability for security should always reside with the ultimate resource owner in the business that owns all other risks and benefits. The security teams and subject matter experts are collectively responsible for advising the accountable owner on the risks, any mitigation, and on doing the actual implementation.

Asset protection responsibilities might be performed by IT operations that manage enterprise-wide assets, DevOps and DevSecOps teams responsible for their workload's assets, or security teams working with the IT or DevOps and DevSecOps teams.

As organizations move to the cloud, many of these responsibilities can transfer to the cloud provider, for example, updating firmware and virtualization solution, or made easier, for example, security configuration scanning and remediation.

For more information on the shared responsibility model, see Shared responsibility in the cloud.

Cloud elasticity

Unlike on-premises resources, cloud resources might exist for only a short time. As needed, workloads can create more instances of servers, Azure Functions, and other resources, to do a job. Azure removes the resources afterward. This scenario can happen within months, but sometimes within minutes or hours. Take this possibility into account for your asset protection processes and measurements.

Cloud elasticity requires adjusting many processes. It improves your visibility, with on-demand inventory instead of static reports. Cloud elasticity also improves your ability to correct issues. For instance, building a new virtual machine for security reasons can happen quickly.

Exception management

Once you identify a best practice for an asset, apply it consistently to all instances of the asset. You might need to make temporary exceptions, but manage the exceptions with specific expiration dates. Be sure that temporary exceptions don't become permanent business risks.

Challenges with measuring value

It can be difficult to measure the business value of asset protection. The impact of a problem isn't obvious until there's a real-world failure. The risk of not updating security for vulnerabilities is silent and invisible.

Prefer automated policy

Favor automated enforcement and remediation mechanisms like Azure Policy for asset protection. This approach helps avoid cost and morale issues from repeatedly doing manual tasks. It also decreases risk from human errors.

Azure Policy allows central teams to specify configurations to use for assets across clouds.

Design controls as a team

All controls should be designed as a partnership with key stakeholders:

- **Asset protection** provides subject matter expertise on the assets, the controls available for them, and

feasibility of implementing the controls.

- **Governance team** provides context of how the controls fit into the security architecture, policies and standards, and regulatory compliance requirements.
- **Security operations** advise on detective controls. They integrate alerts and logs into security operations tools, processes, and training.
- **Vendors and cloud providers** can provide deep subject matter expertise on systems and components to avoid known issues seen across their customer base.

Next steps

The next discipline to review is security governance

Security governance

5/24/2021 • 4 minutes to read • Edit Online

Security governance bridges your business priorities with technical implementation like architecture, standards, and policy. Governance teams provide oversight and monitoring to sustain and improve security posture over time. These teams also report compliance as required by regulating bodies.



Business goals and risk provide the best direction for security. This direction ensures that security focuses their efforts on important matters for the organization. It also informs risk owners using familiar language and processes in the risk management framework.

Compliance and reporting

Compliance and reporting on external security requirements, and sometimes internal policy, are basic required elements of operating in a given industry. The mandatory requirements are like feeding a bear in the zoo. If you don't feed the bear every day, it might eat you.

Architecture and standards

Architecture, standards, and policy provide the critical translation from business requirements and risk into the technical environment. We recommend having a unified view across your enterprise estate instead of splitting up cloud versus on-premises. Attackers don't care about your internal processes and follow the path of least resistance to their goal. That includes moving laterally between cloud and on-premises environments. Most enterprises today are a hybrid environment that spans:

- **On-premises:** Includes multiple generations of technology and often a significant amount of legacy software and hardware. This tech sometimes includes operational technology controlling physical systems with a potential life or safety impact.
- **Clouds:** Typically includes multiple providers for:
  - Software as a service (SaaS) applications

- Infrastructure as a service (IaaS)
- Platform as a service (PaaS)

Security posture management

Hope and reporting problems isn't a plan. Governance in the cloud age must have an active component that continuously engages with other teams. Security posture management is an emerging function. It represents a step forward in the long-term convergence of security functions. These functions answer the question "how secure is the environment?", including vulnerability management and security compliance reporting.

In the on-premises world, security governance followed the cadence of data it could get about the environment. This way of getting data might take time and be constantly out of date. Cloud technology now provides on-demand visibility into the current security posture and asset coverage. This visibility drives a major transformation of governance into a more dynamic organization. This organization provides a closer relationship to other security teams to monitor security standards, provide guidance, and improve processes.

In its ideal state, governance is the heart of continuous improvement. This improvement engages across your organization to constantly improve security posture.

The key tenets of success for governance are:

- **Continuous discovery of assets and asset types:** A static inventory isn't possible in a dynamic cloud environment. Your organization must focus on the continuous discovery of assets and asset types. In the cloud, new types of services are added regularly. Workload owners dynamically spin up and down instances of applications and services as needed, making inventory management a dynamic discipline. Governance teams need to continuously discover asset types and instances to keep up with this pace of change.
- **Continuous improvement of asset security posture:** Governance teams should focus on improving standards, and enforcement of those standards, to keep up with the cloud and attackers. Information technology (IT) organizations must react quickly to new threats and adapt accordingly. Attackers are continuously evolving their techniques, and defenses are continuously improving and might need to be enabled. You can't always get all the security you need into the initial configuration.
- **Policy-driven governance:** This governance provides consistent execution by fixing something once in policy that's automatically applied at scale across resources. This process limits any wasted time and effort on repeated manual tasks. It's often implemented using Azure Policy or third-party policy automation frameworks.

To maintain agility, best practices guidance is often iterative. It digests small pieces of information from multiple sources to create the whole picture and continuously make small adjustments.

Governance and protection disciplines

Protection disciplines include access control, asset protection, and innovation security. The security governance team provides standards and guidance to drive consistent execution of security best practices and controls.

In the ideal state, the protection teams apply these controls and provide feedback on what's working, like challenges in applying the controls. The teams then work together to identify the best solutions.

Governance and security operations

Security governance and security operations work together to provide complete visibility. They ensure that lessons learned from real world incidents are integrated into architecture, standards, and policy.

Governance and security operations provide complementary types of visibility.

- **Security operations** provide insight into the immediate risk of active attacks.

- **Security governance** provides a broad or long view of risk from potential future attacks and attack vectors.

The security architects within the governance function help identify lessons learned from incidents. For example, the root cause of major incidents. They capture the lessons into your organization's standards to ensure consistent application across the enterprise.

For more information, see Security integration.

> **NOTE**
> Some organizations place monitoring of security posture in security operations. We recommend having this monitoring in governance. This placement creates a better relationship with the IT engineering and operations teams that apply the standards. This relationship often results in higher quality communications and better security outcomes. Otherwise, you have a governance team that never sees the real world impact of their standards.

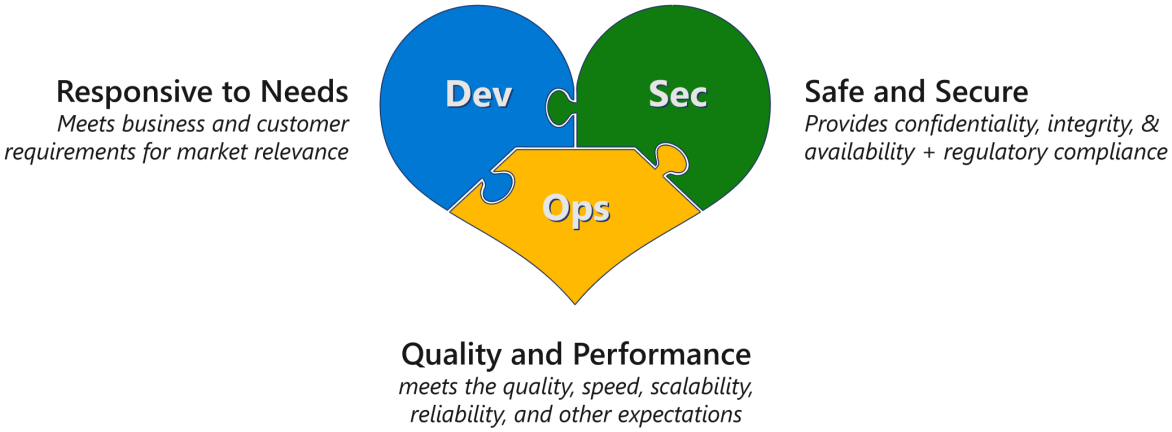
Next steps

The next discipline is innovation security.

Innovation security

5/26/2021 • 12 minutes to read • Edit Online

Innovation is the lifeblood of an organization in the digital age and needs to be both enabled and protected. Innovation security protects the processes and data of innovation against cyberattacks. Innovation in the digital age takes the form of developing applications using the DevOps or DevSecOps method to rapidly innovate without waiting for the traditional waterfall ship schedule that can take months or years between releases.



Developing new capabilities and applications requires successfully meeting three different requirement types:

- **Business development (`Dev`)**: Your application must meet business and user needs, which are often rapidly evolving.
- **Security (`Sec`)**: Your application must be resilient to attacks from rapidly evolving attackers and take advantage of innovations in security defenses.
- **IT operations (`Ops`)**: Your application must be reliable and perform efficiently.

Merging these three requirements together and creating a shared culture is critically important, but often challenging. Leaders of development, IT, and security teams must work together to drive this change. For more information, see the leadership imperative: blend the cultures.

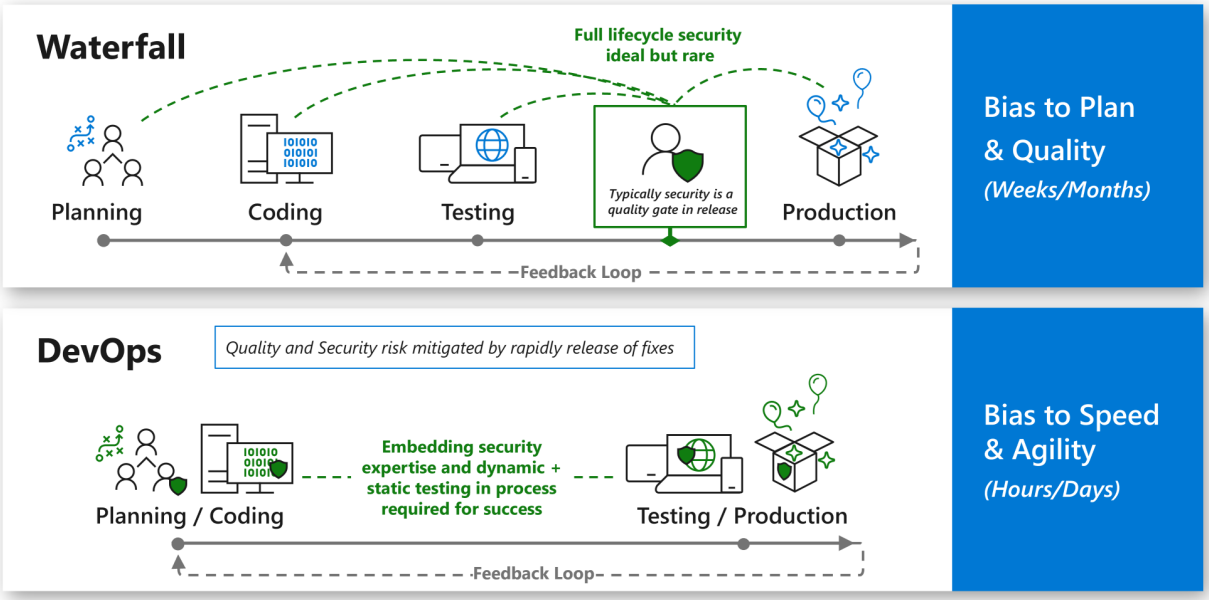
What is DevSecOps?

Technology innovation is frequently developed in the context of a rapid lean and agile development approach that combines development and operations together into a *DevOps* process. We've learned that integrating security into that process is critical to mitigate risks to the innovation process, the organization's growth, and the existing assets in organization. Integrating security into the process creates a *DevSecOps* process.

Secure by design and shifting left

As organizations adopt DevOps and other rapid innovation methodologies, security must be a thread woven throughout the tapestry of the organization and its development processes. Integrating security late in the process is expensive and difficult to fix.

Shift security *left* in the timeline to integrate it into the envisioning, design, implementation, and operation of services and products. As development teams shift to DevOps and adopt cloud technologies, security must be a part of that transformation.



In the waterfall model, security was traditionally a *quality gate* after development finishes.

DevOps expanded the traditional development model (people, process, and technology) to include operations teams. This change reduced the friction that resulted from having the development and operations teams separated. Similarly, DevSecOps expands DevOps to reduce the friction from separate or disparate security teams.

DevSecOps is the integration of security into every stage of the DevOps lifecycle from idea inception through envisioning, architectural design, iterative application development and operations. Teams must align simultaneously to goals of innovation speed, reliability, and security resilience. With mutual understanding and mutual respect for each other's needs, the teams will work on the most important issues first, whatever the source.

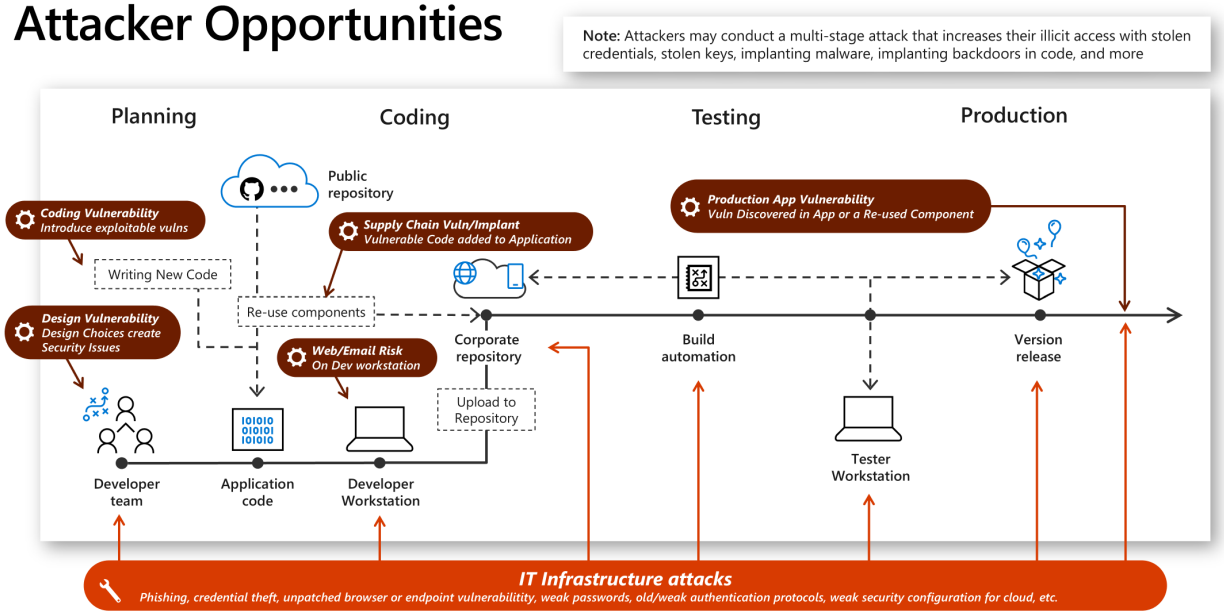
The Cloud Adoption Framework's Organize methodology provides further context on DevSecOps structures in an organization. For more information, see Understand application security and DevSecOps functions.

Why DevSecOps?

DevOps brings agility, DevSecOps brings secure agility.

Nearly every organization on the planet looks to software development to gain a competitive advantage through innovation. Securing the DevOps process is critical to the success of the organization. Attackers have taken notice of this shift to custom applications and are increasingly attacking custom applications during their attacks. These new applications are often rich sources of valuable intellectual property that contain valuable new ideas that aren't yet a commodity in the marketplace.

Protecting this innovation requires that organizations address potential security weaknesses and attacks in both the development process and the infrastructure hosting the applications. This approach applied to both cloud and on-premises.



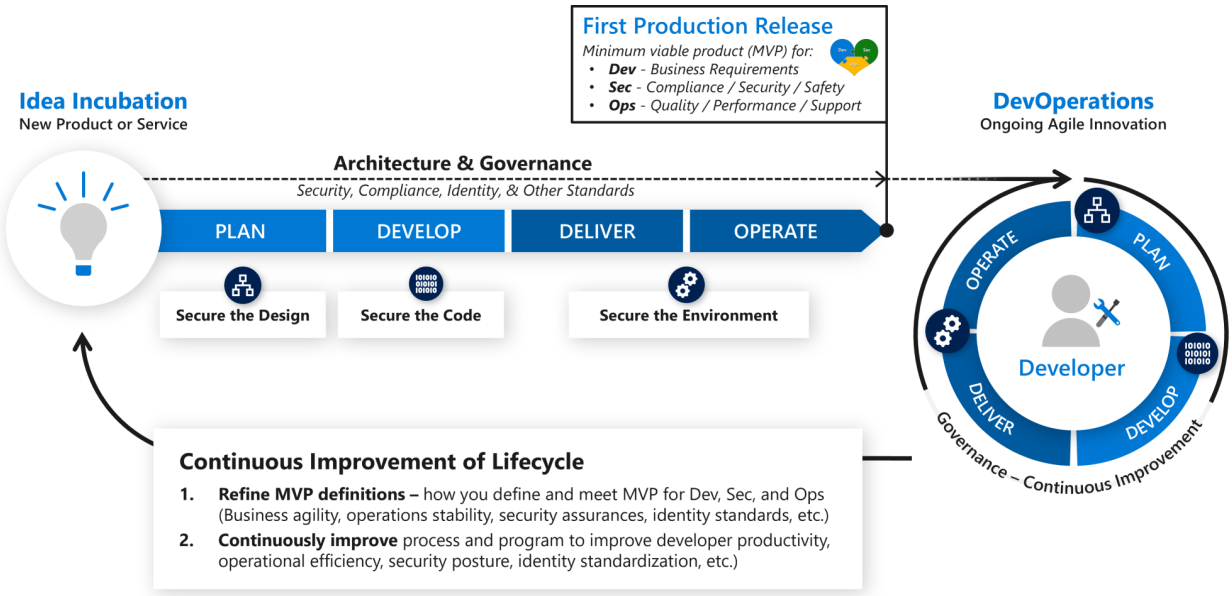
Attackers might exploit weaknesses in:

- **Development process:** Attackers might find weaknesses in the application design process, for example, using weak or no encryption for communications. Or attackers might find weakness in the implementation of the design, for example, code doesn't validate input and allows common attacks like SQL injection. Additionally, attackers might implant back doors in the code that allows them to return later to exploit in your environment or in your customer's environment.
- **IT infrastructure:** Attackers can compromise endpoint and infrastructure elements that the development process is hosted on using standard attacks. Attackers might also conduct a multistage attack that uses stolen credentials or malware to access development infrastructure from other parts of the environment. Additionally, the risk of software supply chain attacks makes it critical to integrate security into your process for both:
- **Protecting your organization:** From malicious code and vulnerabilities in your source code supply chain
- **Protecting your customers:** From any security issues in your applications and systems, which might result in reputation damage, liability, or other negative business impacts on your organization

The DevSecOps journey

Most organizations find that DevOps or DevSecOps for any given workload or application is actually a two-phase process, where ideas first incubate in a safe space and then later released to production and iteratively and continuously updated.

This diagram shows the lifecycle of this kind of innovation factory approach:



Secure innovation is an integrated approach for both of these phases:

- **Idea incubation** where an initial idea is built, validated, and made ready for initial production use. This phase begins with a new idea and ends when the **first production release** meets the minimum viable product (MVP) criteria for:
  - **Development:** Functionality meets the minimum business requirements
  - **Security:** Capabilities meet the regulatory compliance, security, and safety requirements for production use
  - **Operations:** Functionality meets the minimum quality, performance, and supportability requirements to be a production system
- **DevOps:** This phase is the ongoing iterative development process of the application or workload that enables continuous innovation and improvement

The leadership imperative: Blend the cultures

Meeting these three requirements requires merging these three cultures together to ensure that all team members value all types of requirements and work together toward common goals.

Integrating these cultures and goals together into a true DevSecOps approach can be challenging, but it's worth the investment. Many organizations experience a high level of unhealthy friction from development, IT operations, and security teams who work independently, creating issues with:

- Slow value delivery and low agility
- Quality and performance issues
- Security issues

While having a few issues is normal and expected with new development, conflicts between teams often dramatically increase the number and severity of these issues. The conflicts occur, often because one or two teams have a political advantage, and repeatedly override requirements of other teams. Over time, the neglected issues grow in volume and seriousness. Left unsolved, this dynamic might get worse with DevOps as the speed of making decisions increases to meet the rapid evolution of business needs and customer preferences.

Solving these problems requires creating a shared culture that values dev, sec, and ops requirements that are supported by leadership. This approach will allow your teams to work better together and help solve the most urgent issues on any given sprint, whether they're improving security, operational stability, or adding critical business features.

Leadership techniques

These key techniques can help leadership build a shared culture:

1. **No one wins all the arguments:** Leaders must ensure that no single mindset dominates all decisions that might cause an imbalance that negatively impacts the business.

2. **Expect continuous improvement, not perfection:** Leaders should set an expectation of continuous improvement and continuous learning. Building a successful DevSecOps program doesn't happen overnight. It's a continuous journey with incremental progress.

3. **Celebrate both common interests and unique individual values:** Ensure the teams can see that they're working toward common outcomes and each individual provides something the others can't. All of the requirement types are about creating and protecting the same business value. Development is trying to create new value, while ops and security are trying to protect and preserve that value, against different risk scenarios. Leaders at all levels throughout the organization should communicate this commonality and how important it is to meet all types of requirements for both immediate and long-term success.

4. **Develop shared understanding:** Everyone on the team should have a basic understanding of:

   - **Business urgency:** The team should have a clear picture of revenue at stake. This view should include current revenue (if the service is offline), and potential future revenue that will be affected by a delay in delivery of applications and features. This should be directly based on signals from leadership stakeholders.
   - **Likely risks and threats:** Based on threat intelligence team input, if present, the team should establish a sense of the likely threats that the application portfolio will face.
   - **Availability requirements:** The team should have a shared sense of the operational requirements such as required uptime, expected lifetime of the application, and troubleshooting and maintenance requirements, for example, patching while service online.

5. **Demonstrate and model the desired behavior:** Leaders should publicly model the behavior that they want from their teams. For example, show humility, focus on learning and value the other disciplines. Another example is development managers discuss the value of security and high-quality applications or security managers discuss the value of rapid innovation and application performance.

6. **Monitor the level of security friction:** Security naturally creates friction that slows down processes. It's critical for leaders to monitor the level and type of friction that security generates:

   - **Healthy friction:** Similar to how exercise makes a muscle stronger, integrating the right level of security friction in the DevOps process strengthens the application by forcing critical thinking at the right time. If teams are learning and using those learnings to improve security, for example, considering how why, and how an attacker might try to compromise an application, and finding and fixing important security bugs, then they are on track.
   - **Unhealthy friction:** Look out for friction that impedes more value than it protects. This often happens when security bugs generated by tools have a high false positive rate or false alarms, or when the security effort to fix something exceeds the potential impact of an attack.

7. **Integrate security into budget planning:** Ensure that security budget is allocated proportionally to other investments into security. This is analogous to a physical event like a concert where the event budget includes physical security as a norm. Some organizations allocate 10 percent of total cost for security as a general rule to ensure consistent application of security best practices.

8. **Establish shared goals:** Ensure performance and success metrics for application workloads reflect development, security, and operations goals.

> **NOTE**
>
> Ideally these teams should collectively create these shared goals to maximize buy in, whether for the whole organization or for a particular project or application.

Identify the DevSecOps MVP

During the transition from an idea to production, it's critical to ensure that the capability meets the minimum requirements, or the minimum viable product (MVP), for each requirement type:

- **Developers (dev)** focus on representing the business needs for rapid delivery of capabilities that meet the expectations of users, customers, and business leaders. Identify the minimum requirements to ensure that the capability helps make the organization successful.
- **Security (sec)** brings focus to meeting compliance obligations and defending against the attackers that are continuously seeking illicit gain from the organization's resources. Identify the minimum requirements to meet regulatory compliance requirements, sustain security posture, and ensure security operations can rapidly detect and respond to an active attack.
- **Operations (ops)** focus on performance, quality, and efficiency, ensuring that the workload can continue to deliver value over the long term. Identify the minimum requirements to ensure that the workload can perform and be supported without requiring massive architectural or design changes in the foreseeable future.

The definitions for MVP can change over time, and with different workload types, as the team learns together from their own experience and from other organizations.

Integrate security natively in the process

Security requirements must focus on natively integrating with the existing process and tools. For example:

- Design activities like threat modeling should be integrated into design phase
- Security scanning tools should be integrated into the continuous integration and continuous delivery (CI/CD) systems like Azure DevOps, GitHub, and Jenkins
- Security issues should be reported using the same bug tracking systems and processes, for example, prioritization scheme, as other bugs.

The way that security is integrated into the process should be continuously improved as the teams learn and processes mature. Security reviews and risk assessments should ensure mitigations are integrated into the end-to-end development processes, the final production service, and the underlying infrastructure.

For more information on DevSecOps, see DevSecOps technical controls.

Tips on navigating the journey

Transformation requires building toward this ideal state incrementally on a journey. Many organizations will have to navigate complexity and challenges on this journey. This section outlines some of the common ones that organizations face.

- **Education and culture changes are critical early steps:** *You go to war with the army you have*. The team you have will often need to develop new skills and adopt new perspectives to understand the other parts of the DevSecOps model. This education and culture change takes time, focus, executive sponsorship, and regular follow up to help individuals fully understand and see the value of the change. Changing cultures and skills drastically can sometimes tap into the professional identity of individuals, creating potential for strong resistance. It's critical to understand and express the why, what, and how of the change for each individual and their situation.
- **Change takes time:** You can only move as fast as your team can adapt to the implications of doing things in new ways. Teams will always have to do their existing jobs while they transform. It's critical to carefully prioritize what is most important and to manage expectations of how fast this change can happen. Focusing on a crawl, walk, run strategy, where the most important and foundational elements come first, will serve your organization well.
- **Limited resources:** A challenge organizations usually face early on is to find talent and skills in both security and application development. As organizations begin to collaborate more effectively, they might find

hidden talent, such as developers with a security mindset or security professionals with a development background.

- **Shifting nature of applications, code, and infrastructure:** The technical definition and composition of an application is fundamentally changing with the introduction of technologies such as serverless, cloud services, cloud APIs, and codeless applications, such as Power Apps. This shift is changing development practices, application security, and even empowers non-developers to create applications.

> **NOTE**
>
> Some implementations combine operations and security responsibilities into a **site reliability engineer (SRE)** role.
>
> While fusing these responsibilities into a single role might be the ideal end-state for some organizations, this is often an extreme change from current enterprise practices, culture, tooling, and skill sets.
>
> Even if you are targeting an SRE model, we recommend starting by embedding security into DevOps using practical quick wins and incremental progress outlined in this guidance to ensure you are getting good return on investment (ROI) and meeting immediate needs. This will incrementally add security responsibilities to your operations and development personnel, which gets your people closer to the end-state of an SRE (if your organization plans to adopt that model later).

Next steps

Review the DevSecOps technical controls for more detailed guidance on DevSecOps.

For information on how GitHub advanced security integrates security into your continuous integration and continuous delivery (CI/CD) pipelines, see About GitHub advanced security.

For more information and tooling on how Microsoft's IT organization implemented DevSecOps, see Secure DevOps toolkit.

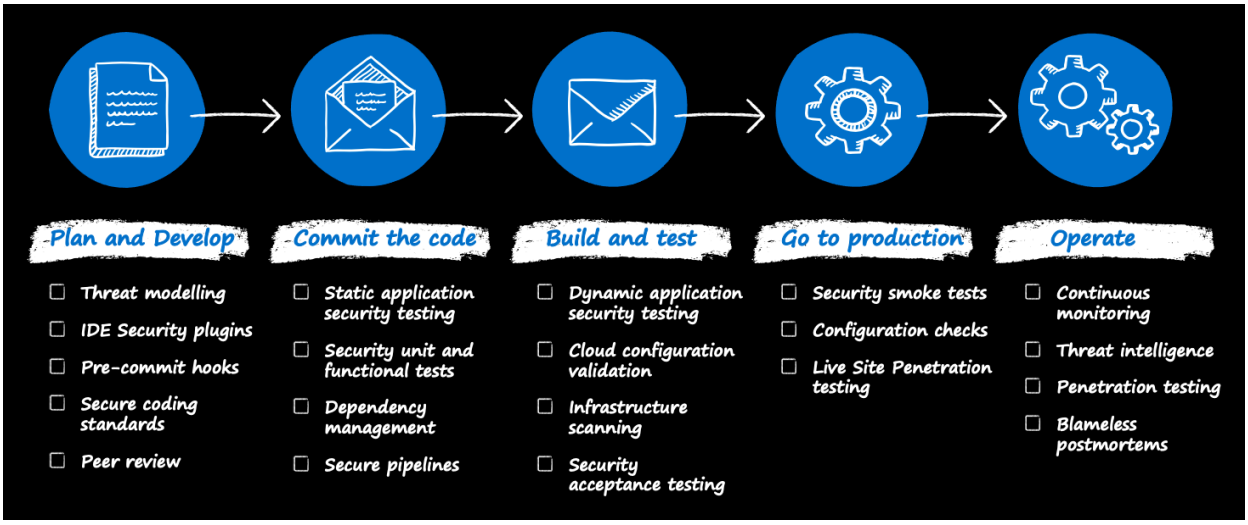
DevSecOps controls

5/24/2021 • 13 minutes to read • Edit Online

DevSecOps is the application of innovation security by integrating security processes and tools into the DevOps development process.

Because DevOps itself is an emerging discipline with a high degree of process variations, successful DevSecOps is best achieved by understanding and thoughtfully integrating security into development process. Adding security should start with low friction changes to the code, the development processes, and the infrastructure that hosts the workload. Focus first on changes that have the highest positive impact on security while placing a low burden on DevOps processes and skills.

This documentation reviews each stage of a continuous integration and continuous delivery (CI/CD) DevOps process and what security controls we recommend integrating first.



Plan and develop

Typically, modern development follows an agile development methodology. Scrum is one implementation of agile methodology that has every sprint start with a planning activity. Introducing security into this part of the development process should focus on:

- **Threat modeling** to view the application through the lens of a potential attacker
- **IDE security plug-ins and pre-commit hooks** for lightweight static analysis checking within an integrated development environment (IDE).
- **Peer reviews and secure coding standards** to identify effective security coding standards, peer review processes, and pre-commit hooks.

It's not mandatory to add all of these steps, however, each one helps discover security issues early, when they're much cheaper and easier to fix.

Threat modeling

Threat modeling is arguably the most important security practice. It delivers immediate results and helps develop a security mindset in developers that can improve security in all of their future projects.

Threat modeling is a simple concept, though it can become detailed and technical if it needs to be. Threat modeling develops and documents a realistic security view of your application that includes:

- How attackers can abuse the application's design
- How to fix these issues
- How important it is to fix them

Threat modeling effectively puts you in the mindset of an attacker. It allows you to see the application through their eyes and can block their attack before they get a chance to do anything about it. If your team uses user personas in design, you can treat the attacker as a hostile user persona.

There are several differently published approaches for threat modeling that range from simple question and answer methods to detailed tool-based analysis. The can be based on methodologies like the STRIDE model, the DREAD model, or OWASP threat modeling.

Threat modeling: Start simple

Because some approaches to threat modeling can be a time consuming and skill intensive process, we recommend starting with a simpler approach based on basic questions. These simpler methods aren't as thorough, but start the critical thinking process and quickly identify major security issues

These simple questions for threat modeling are ideal to get your started:

- Simple questions method from Microsoft: This method asks specific technical questions designed to surface common security design mistakes.
- OWASP threat modeling: This method focuses on asking simple, non-technical questions to get the process started.

You can use either or both of these approaches, depending on what works better for the team.

As the team gets more comfortable with the process, they can apply more advanced techniques from Microsoft security development lifecycle and integrate threat modeling tools like Microsoft threat modeling tool to get deeper insights and help automate the process.

Another helpful resource is a guide to threat modeling for developers.

IDE security plug-ins and pre-commit hooks

Developers focus on the speed of delivery, and security controls might slow down the process. Typically, the slow-down occurs if the security checks start at the pipeline. A developer will find out about the potential vulnerability after pushing the code to the repository. To speed up this process and give immediate feedback, it's worth adding steps such as IDE security plug-ins and pre-commit hooks.

IDE security plug-ins identify different security issues during the development process in the developer's comfort zone, which is their IDE environment. Plug-ins can provide immediate feedback if there's a potential security risk in the developer's written code, or the third-party library or package included. Depending on the chosen IDE, many open-source or commercial plug-ins are provided by security companies available on the market.

Another step worth considering is to introduce a pre-commit framework if the version control system allows it. A pre-commit framework provides Git hook scripts that enable identification issues before submitting code for code review. One example is pre-commit and can be implemented in GitHub.

Peer review and secure coding standards

Pull requests are standard in the development process. Part of the pull request process is peer reviews that allow detecting uncovered defects, bugs, or issues related to human mistakes. It's a good practice to have a security champion or knowledgeable security teammate who can incorporate and guide the developer during the peer review before creating a pull request.

Secure coding practice guidelines help developers learn essential secure coding principles and how they should be applied. There are secure coding practices such as OWASP secure coding practices available and incorporated with the general coding practices.

Commit the code

Typically, developers create, manage, and share their code in repositories such as GitHub or Azure Repos. This approach provides a central, version-controlled library of code that can be collaborated on easily. However, enabling many collaborators on a single codebase can also introduce the risk of changes being introduced. That risk can lead to vulnerabilities or the unintentional inclusion of credentials or tokens in commits.

To address this risk, development teams should evaluate and implement a repository scanning capability. Repository scanning tools perform static code analysis on source code within repositories and look for vulnerabilities or credentials and flag items found for remediation. This capability acts to protect against human error and is a useful safeguard for distributed teams where many people are collaborating in the same repository.

Dependency management

It's known that up 90 percent of the code in current applications contains and is based on external packages and libraries. With the adoption of dependencies in the source code, it's essential to address potential risks. Many third-party libraries have serious security problems themselves. Additionally, developers not consistently implement the proper lifecycle and keep them up to date.

Developer teams should ensure that they know what components are included in their applications, be sure that secure and up-to-date versions are downloaded from the known sources, and there's a process for keeping them up to date. This can be done with tools like the OWASP Dependency-Check project, WhiteSource, and others.

To focus on the dependency vulnerabilities or their lifecycle only is not enough. It's also important to address package feeds security. There are known attack vectors toward package management systems: typosquatting, compromising existing packages, substitution attacks, and others. Therefore responsible for the package management administration must address those risks. For more information, see the white paper Three Ways to Mitigate Risk When Using Private Package Feeds.

Static application security testing

Once third-party libraries and package management have been addressed, it's essential to shift the focus and improve one's written code security state. There are different ways to improve code security, like using IDE security plug-ins and wiring incremental static analysis pre-commit and commit checks discussed before. It's also possible to do the complete source code scanning to catch some mistakes missed by previous steps. It's needed but might take hours or days to run on a large block of code. This approach can slow down development and introduce burden.

There's a need to start from somewhere when you start to implement static code scanning practices. One way is to introduce static code analysis inside of continuous integration to verify security as soon as changes are made. One of these tools is SonarCloud, which wraps multiple static application security testing (SAST) tools for different languages. SonarCloud assesses and tracks technical debt with a focus on maintainability, such as code quality and style, and has security-specific checkers. There are many others commercial and open-source tools available on a market.

To ensure that the feedback loop is effective, it's crucial to tune the tool to minimize false positives and provide clear, actionable feedback on problems to fix. Additionally, it's good to implement the workflow, which prevents code commit to the default branch if there are findings. For both quality and security findings. In this way, security becomes a part of the unit testing experience.

Secure pipelines

DevOps brings automation on another level where everything goes through the pipelines. Continuous integration and continuous delivery is a large part of modern development. Infrastructure is why pipelines are a central part of the development and have keys to the kingdom. Pipelines introduce unique security challenges. They can be compromised to run malicious code, credentials might be stolen from pipelines, or an attacker

without access to the production might modify the pipeline to achieve their goals.

DevOps teams must ensure the proper security controls are implemented for the pipeline. Depending on the chosen platform, there are different guidelines on how to address those risks. For more information, see Securing Azure Pipelines.

Build and test

Many organizations use build and release pipelines to automate and standardize the processes for building and deploying code. This use of pipelines allows development teams to make iterative changes to sections of code quickly and at scale, without the need to spend large amounts of time redeploying or upgrading existing environments. Using pipelines also enables teams to promote code from development environments, through testing environments and ultimately into production. As part of this automation, development teams should include security tools that run scripted automated tests when they deploy code into testing environments. The test should include testing such as unit testing the applications features to check for vulnerabilities or checking for public endpoints to ensure they're intentionally accessible.

Dynamic application security testing

In a classical waterfall development model, security was typically introduced on the last stop, right before going to production. One of the most popular security approaches is penetration testing or pen test. The penetration test is an important step, which allows looking at the application from the black-box security perspective, closest to the attacker mindset. A penetration test consists of several action points, one of them is known as dynamic application security testing (DAST). DAST is a web application security test that finds security issues in the running application by seeing how the application responds to specially crafted requests. DAST tools are also known as web application vulnerability scanners. One of them is an open-source OWASP Zed Attack Proxy (ZAP), which finds vulnerabilities in the running web application. There are several ways how OWASP ZAP does the scan: passive baseline scan or full scan depending on the configuration.

The downside of a pen test is that it takes time. The proper pen test might take up to several weeks, and with DevOps development speed, it becomes unsustainable. However, it's still worth adding a *lighter* version of pen test during the development process to uncover what might be missed by SAST and previous steps. DAST tools like OWASP ZAP might help. Developers can integrate OWASP ZAP in the pipeline as a task. During the execution, the OWASP ZAP scanner is spun up in the container and does the scanning process after it publishes results. This approach might not be a perfect, as it's not complete penetration testing, however, it's one more quality gate in the development cycle for improving the security posture.

Cloud configuration validation and infrastructure scanning

Alongside scanning and securing the code for applications, it's important to ensure that the environments that applications are deployed into are also secure. This is key for organizations who want to move at pace, innovate, and use new technologies or create environments quickly for experimentation. Azure has capabilities that enable organizations to create security standards from environments, such as Azure Policy, that can be used to create policy sets that prevent the creation of certain workload types or configuration items such as public IP addresses. These *guardrails* enable teams to experiment within a safe and controlled environment, balancing innovation and governance.

One of the DevOps aspects when bringing closer developers and operations in cooperation is transferring infrastructure into an infrastructure-as-code approach.

> Infrastructure as code (IaC) is the management of infrastructure (networks, virtual machines, load balancers, and connection topology) in a descriptive model, using the same versioning as the DevOps team uses for source code. Like the principle that the same source code generates the same binary, an IaC model generates the same environment every time it's applied. IaC is a key DevOps practice and is used in conjunction with continuous delivery.

DevSecOps is shifting security left and it's not only about application security but about infrastructure security as well. One of the steps is to introduce security scanning of infrastructure before it's deployed in the cloud. As infrastructure became code, it became possible to apply the same security actions as those for application security shown in the previous challenges. There are security tools that can do the security scanning based on chosen IaC strategy.

With the adoption of the cloud, containerization is a popular approach in application architecture decisions. Some of the container repositories scan images to catch packages with the known vulnerabilities. There's still a risk that a container might contain out-of-date software. Because of this risk, it's vital to scan the container for security risks. There are plenty of open-source and commercial security tools targeting this area and supporting tight integration in the continuous delivery process. Those tools are helping to adopt DevSecOps for infrastructure as code and specifically containers part.

Go to production and operate

When the solution is delivered to the production, it's vital to continue overseeing and managing the security state. At this stage, it's time to focus on the cloud infrastructure and application overall.

Configuration and infrastructure scanning

To obtain visibility for cloud subscriptions and resource configuration across multiple subscriptions, the Azure tenant security solution from the AzSK team can be used.

Azure includes monitoring and security capabilities that are designed to detect and alert anomalous events or configurations that require investigation and potential remediation. Technologies such as Azure Defender, Azure Security Center, and Azure Sentinel are first-party tools that natively integrate into the Azure environments. These tools complement the environment and code security tools to provide a broad-reaching set of security monitoring to enable organizations to experiment and innovate at a pace in a secure way.

Penetration testing

Penetration testing is a recommended practice for environments to check for any vulnerabilities in the infrastructure or application configuration that might create weaknesses that attackers could exploit.

There are many products and partners that provide penetration testing services. Microsoft provides guidance for penetration testing in Azure.

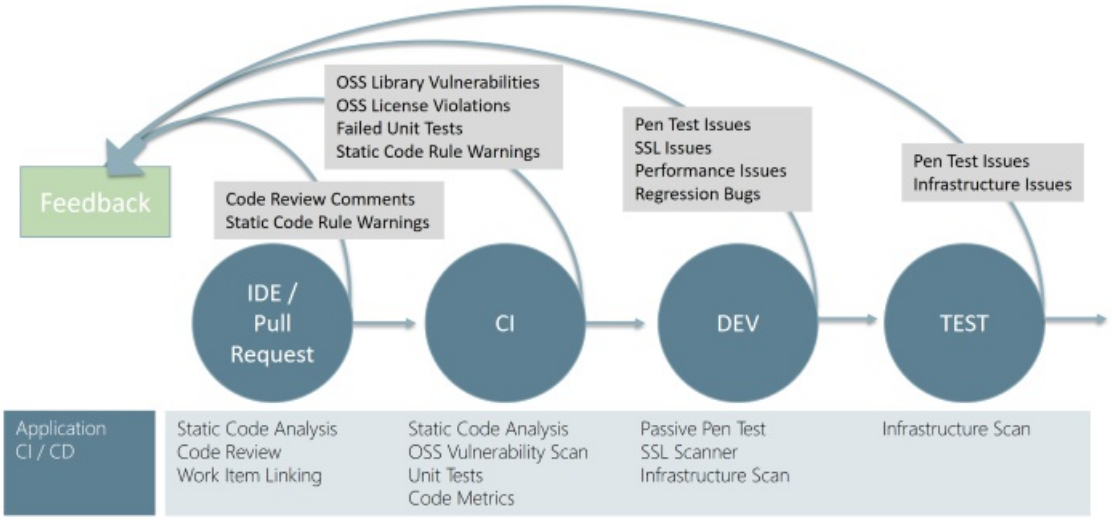
Testing typical covers the following test types:

- Tests on your endpoints to uncover vulnerabilities.
- Fuzz testing (finding program errors by supplying malformed input data) of your endpoints.
- Port scanning of your endpoints.

Actionable intelligence

The tools and techniques in this guidance can significantly contribute toward a holistic security model for organizations who want to move at pace and experiment with new technologies that aim to drive innovation. A key element of DevSecOps is data-driven, event-driven processes that enable the three functions to operate effectively in identifying, evaluating, and responding to potential risks. Many organizations choose to integrate these alerts and telemetry into their IT service management (ITSM) platform in order to bring the same structured workflow to security events that they use for other incidents are requests.

Feedback loops



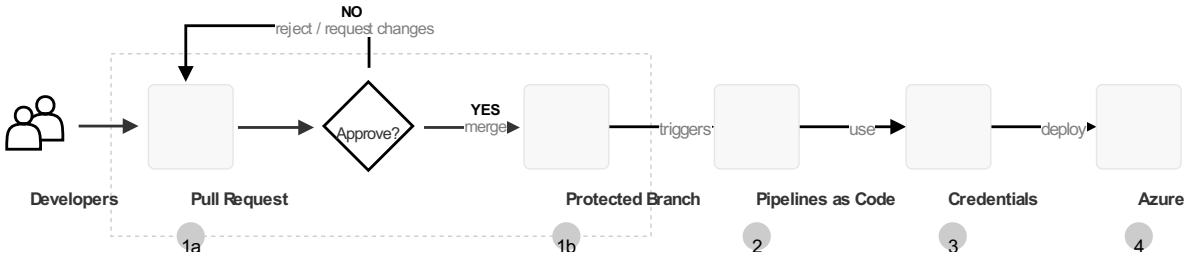
All of these techniques and tools enable teams to find and flag risks and vulnerabilities that require investigation and potential resolution. Operations teams who receive an alert, or discover a potential issue when they investigate a support ticket, need a route back to the development team to flag items for review. A smooth, collaborative, feedback loop is vital to address issues quickly and to minimize the risk of a vulnerability as much as possible. A common pattern for this feedback is to integrate it into the organization's developer work management system, such as Azure DevOps or GitHub to link alerts or incidents to work items for developers to plan and action. This process provides an effective way for developers to resolve issues within their standard workflow, including development, testing, and release.

Secure the pipeline and CI/CD workflow

5/24/2021 • 5 minutes to read • Edit Online

Automation and the agile methodology enables teams to deliver faster, but also adds complexity to security because the workflow extends to the developer teams themselves.

The following diagram illustrates a baseline CI/CD workflow. The red configuration icon indicates security permissions which must be configured by the customer. This follows the shared responsibility model, where Azure and other vendors provide permissions, which must be configured by the customer according to their governance model and business requirements.



Let's examine each stage of this typical workflow to help you understand how the configurations often depend on one another. Your workflow might have more stages. The following concepts will help you understand CI/CD and help you design your workflow for security.

Stage 1: Git workflow

Code changes, not just to software, but also to *pipeline as code* and infrastructure as code, are saved and managed in Git. Git is a distributed source code management software. When code is pushed from local computers to the centralized Git server, business rules can be applied before it's accepted.

Pull requests and collaboration

The industry standard workflow, regardless of your software configuration management (SCM) software as a service (SaaS) vendor, is to use pull requests, which can act both as an automated quality gatekeeper and a manual approval step before source code is accepted.

The pull request workflow is designed to introduce healthy friction, which is why it should only be applied to secure **specific Git branches**. Especially the branches that will trigger automated workflows that can deploy, configure, or in any other way affect your cloud resources. These branches are called protected branches and typically follow naming conventions such as `production` or `releases/*`.

It's common for pull requests to require:

- Peer reviews
- Passing continuous integration (CI) builds
- Manual approval

If the requirements are met, the code changes are accepted and can be merged.

Restrict access to protected branches

The pull request workflow is used together with restricted access controls. The pull request workflow can't be enforced however, unless the server is configured to reject direct changes to protected branches.

A developer can't push directly to the `production` branch, but instead must create a pull request that targets the

protected branch. Each SCM vendor has a different flavor for achieving restricted access to protected branches. For example, with GitHub this feature is only available for organizations using GitHub team or GitHub Enterprise cloud.

Document your Git access model

Because the collaboration model is complex and has many moving parts, it's helpful to create a table that documents all possible ways code changes can trigger deployments for example:

BRANCH NAME	REQUIRES PR?	DEPLOYS?	DEVELOPER ACCESS	ADMIN ACCESS
`feat/*`	No	No	Read/write	Read/write
`main`	Yes	Staging	Read	Read/write
`production`	Yes, from `main` only	Production	Read	Read/write

This Git access table example is oversimplified to illustrate its purpose. In practice, there are often more actors, more deployment targets, and multiple pipelines that run for different use cases. Your table structure might be different depending on the requirements of your organization and your workloads.

The table should help you answer questions such as:

- If a developer pushes code changes to branch X, will it deploy? If so, to which environment?
- At what point in the application code lifecycle is a vulnerability scan done?
- If a security vulnerability is found, how many code pushes and approvals are required before it lands in production?

Not only is this table useful for debugging and static documentation, but also for team collaboration. It's transparent to developers where healthy friction has been introduced into the workflow to prioritize code quality and security. More importantly, it shows the developer the expected path for code changes to reach production.

Because DevOps is a journey, your Git access model isn't static. It will change and evolve as the teams find their own rhythms and mature. That's why it's important to store this documentation as close as possible to the code, for example, in the Git repositories.

To learn more about pull requests and protected branches, see:

- Set branch permissions
- Create, view, and manage pull requests
- Improve code quality with branch policies
- About pull requests
- About protected branches

Stage 2: Pipelines as code

The pipeline-as-code movement accelerated automation adoption and deployments by moving pipeline definitions and configurations from the CI vendor to the developers, bringing the build and deployment logic closer to the corresponding application logic. The greater flexibility here also comes with greater responsibility.

RBAC controls in a UI-driven pipeline can prevent individual users from making destructive changes. Pipelines as code, however, often run with privileged identities and can destroy your workloads if instructed to do so.

Stage 3: Secure your deployment credentials

Pipelines and code repositories should not include hard-coded credentials and secrets. Credentials and secrets should be stored elsewhere and use CI vendor features for security. Because pipelines run as headless agents, they should never use an individual's password. Pipelines should run using headless security principals instead, such as service principals or managed identities. Access to this security principal's credentials, database connection strings, and third-party API keys should also be securely managed in the CI platform.

How a credential is secured, gates, and approvals are vendor-specific features. When choosing a CI platform, make sure it supports all the features you require.

Azure Pipelines is an enterprise-scale continuous integration solution where credentials are stored as service connections, upon which you can configure approvals and checks. This configuration includes manual approval and specific branch or pipeline authorizations.

Azure Key Vault

If your CI platform supports it, consider storing credentials in a dedicated secret store, for example Azure Key Vault. Credentials are fetched at runtime by the build agent and your attack surface is reduced.

Stage 4: Securing your Azure resources

Your Azure resources should be secured according to the principle of least privilege, applied to both permissions and scope.

For more information, see:

- Best practices for Azure RBAC
- What is Azure role-based access control (Azure RBAC)?

Create custom roles for build agents

CI/CD automation applies not just to applications, but also to infrastructure. Infrastructure as code (IaC) templates ensure consistent deployments and help centralized cloud platform teams scale.

It's important to understand that IaC automation can go wrong. It can misconfigure, and in worst case scenarios, permanently delete infrastructure. When you plan your cloud journey, identify in advance which operations are business critical and require human intervention.

For example, cannot-delete management locks, might be applied to business critical resources like a data. To prevent this from happening, you might *remove* `Microsoft.Authorization/*/Delete` permissions from a service principal used in CI automation. In this example and use-case, the service principal can *create* the management lock, but not delete it.

It's recommended to create a custom role for your CI agents. Remember, build agents run headless, and headless agents are brainless. Choose your permissions carefully.

To learn more, see:

- Azure AD built-in roles
- Azure resource provider operations

Resources

- Platform automation and DevOps
- Pipelines security walkthrough
- Security via templates
- DevSecOps in GitHub

Next steps

Now that you understand how to secure DevOps, learn more about end-to-end governance from DevOps to Azure.

End-to-end governance from DevOps to Azure

End-to-end governance from DevOps to Azure

5/26/2021 • 3 minutes to read • Edit Online

It's not sufficient to plan and implement an Azure role-based access control (RBAC) model for Azure Resource Manager templates (ARM templates), which restricts access via Azure portal and Azure CLI.

If this model isn't mirrored for DevOps automation, your organization might leave a **security back-door** open. Consider an example where a developer doesn't have access via ARM templates, but still has sufficient permissions to change application code or infrastructure as code and trigger an automation workflow. The developer, indirectly via DevOps, can access and make destructive changes to your ARM templates.

This article explains best practices to help plan for and avoid a scenario where destructive changes can occur.

Single identity management plane with Azure AD groups

The first step is to integrate Azure Active Directory to use single sign-on per identity management best practice.

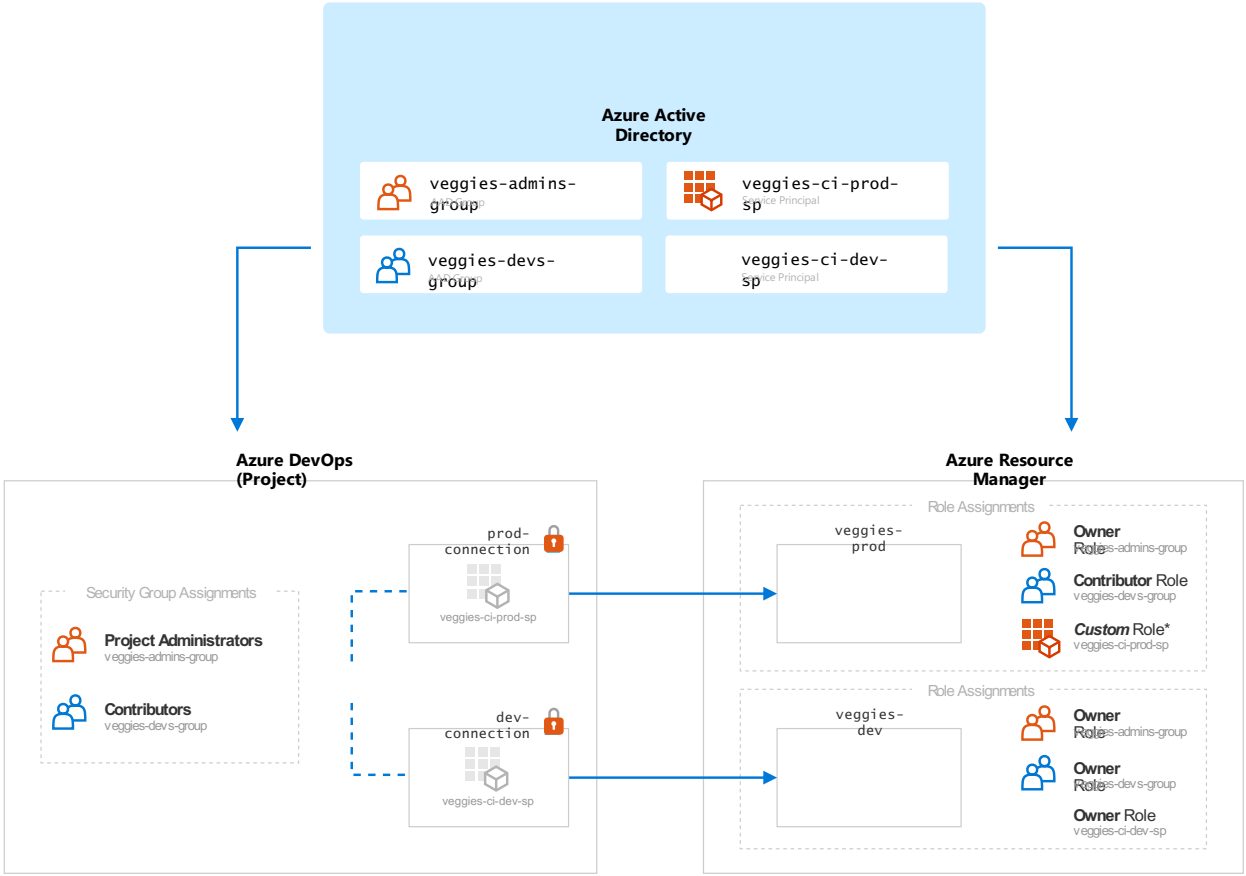
If you aren't using an Azure first party CI product like Azure DevOps, check if your vendor offers Azure AD integration.

The second step is to use Azure AD groups, the same groups you're already using for your ARM templates RBAC model. It's a best practice to assign roles to Azure AD groups, not to individuals. To create an end-to-end governance model, you will need to do this step for ARM templates and DevOps.

Azure DevOps has tight integration with Azure Active Directory including Azure AD groups membership, making it easy to apply role assignments to the same Azure AD group.

> **NOTE**
> If you are using another CI vendor, you might have an intermediary logical container for managing group memberships, which you also need to maintain if Azure AD group membership is not synchronized.

The following diagram illustrates how Azure AD is used as the single identity management plane. In ARM templates and in our DevOps tooling (Azure DevOps in this example), we only need to manage role assignments, not memberships, which should be managed in Azure AD. Note the resource names follow recommended naming conventions and abbreviations for Azure resources.



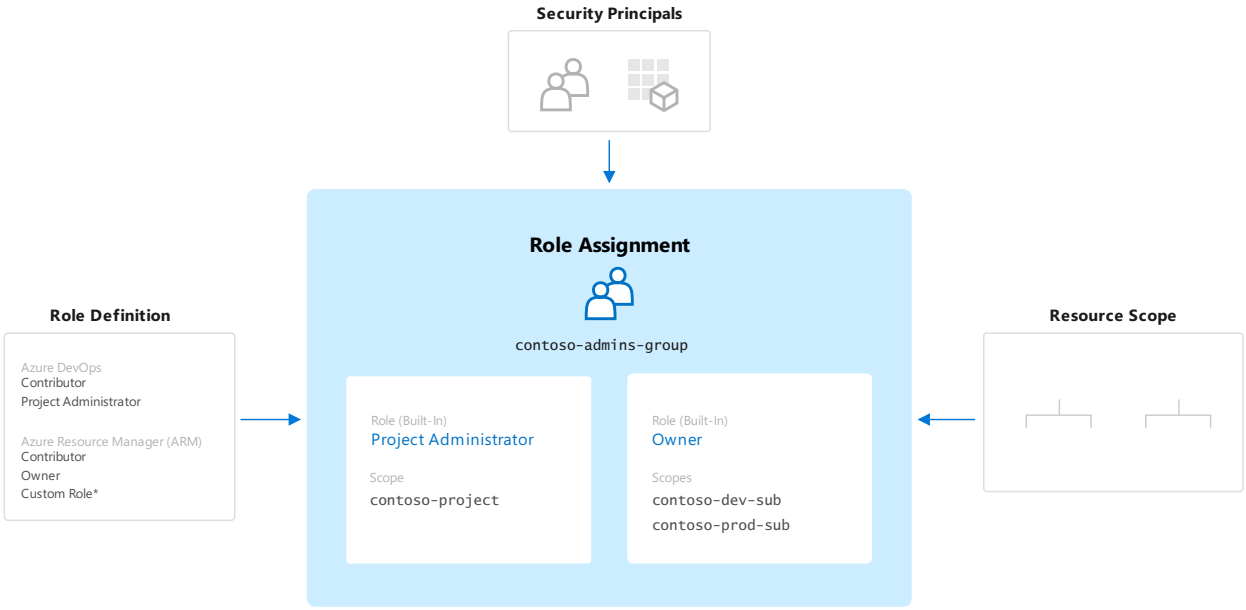
Example scenario: Remove contractor access with a single step, Azure AD membership

To make end-to-end governance concrete, let's examine the benefits with an example scenario.

If you use Azure AD as your single identity management plane, you can remove a developer's access to your Azure resources in one action, by adjusting their *Azure AD group memberships*. For example, if a contractor's access should be revoked upon project completion, when you remove the contractor's membership from the relevant Azure AD groups, access to ARM templates and Azure DevOps is removed.

Mirror RBAC model with role assignments

When planned well, the RBAC model in your CI tooling will closely mirror your Azure RBAC model. Although the Azure AD group name examples in the diagram above imply security rules, membership alone doesn't enforce security. Remember that RBAC is a combination of principals, definitions, and scopes, which does not go into effect **until the role assignment is created**.



- The diagram illustrates role assignment for a single Azure AD group, `contoso-admins-group`.
- This Azure AD group is assigned the Owner role for Azure ARM templates at **multiple** subscription scopes:
  - `contoso-dev-sub` subscription
  - `contoso-prod-sub` subscription
- The `contoso-admins-group` is added to the Project Administrators group for Azure DevOps at a **single** project scope.

The Azure AD group has similarly privileged roles for both ARM templates and DevOps. Following this logic, if we have a developer group assigned the Contributor role for ARM templates, we would not expect them to belong in the Project Administrators group for DevOps.

Now that you understand the need to secure ARM templates and DevOps workflows, you should:

- Review your RBAC model and think about how the roles and responsibilities you have defined for ARM templates match to your CI/CD workflow.
- Review your CI platform's identity management solutions and integrate Azure Active Directory. Ideally, you want to include the Azure AD group membership.
- Review the roles and access levels offered by your CI tooling and compare them with your Azure RBAC model. The roles and access levels will not map one to one. Check your configuration. If a developer doesn't have access for ARM templates, they shouldn't have access for DevOps. In the simplest example, a developer who doesn't have *write* permissions to production resources shouldn't have direct access to trigger production pipeline runs.

To learn more about governance design and permissions, see:

- Governance design for multiple teams
- Recommended method for granting and restricting permissions
- Default permissions and access for Azure DevOps
- Managing people's access to your organization with roles

Next steps

Now that you understand the need to secure ARM templates and DevOps workflows, learn how to manage secrets in a secure way.

Manage secrets in a secure way

Governance baseline considerations for Azure Virtual Desktop

8/16/2021 • 8 minutes to read • Edit Online

This article covers key design considerations and recommendations for security, governance, and compliance in a Cloud Adoption Framework enterprise-scale landing zone architecture for Azure Virtual Desktop.

As with any IT service, it's important to build the environment to scale, secure it, and be able to operate your environment simply and efficiently. While the Azure Virtual Desktop service does most of the front-end work, you still need to have the right control mechanisms to keep your systems and data safe. You also need processes to continually review those controls, report changes and, if necessary, remediate. At the end of this article, you'll understand the critical design areas for security, governance, and compliance, and you'll have clear guidance on Microsoft recommendations in each area.

In most cases, Azure Virtual Desktop is deployed into a landing zone as part of the Microsoft Cloud Adoption Framework for Azure. Microsoft recommends reviewing the Cloud Adoption Framework to ensure that your environment has the right foundation for security, compliance, governance, and cost management.

Design considerations

- **Identity:** Decide on a tool for multifactor authentication and conditional access for user identities. For Azure Virtual Desktop, as for most workloads in Azure, identity is a security boundary. User identity is the central mechanism of user access to desktops, applications, and company data. It's best to protect user credentials during sign-in with multifactor authentication and conditional access.
- **Audit logs:** Audit logs and Azure Virtual Machines are critical to troubleshooting when issues arise, but also serve as a security tool for an Azure Virtual Desktop environment. What tools do you use to capture security or performance logs within your virtual machine (VM)? Are audit logs for Azure Virtual Desktop stored in a central Azure Monitor Logs workspace, or in an isolated Azure Monitor Logs workspace dedicated for Azure Virtual Desktop? Also consider whether to use a partner tool to analyze the logs for security patterns or other reporting needs. What tools do you use to capture security or performance logs within your VM?
- **Compliance:** Nearly all corporations are required to comply with government or industry regulatory policies. It's important to review those policies with your compliance team and have the correct controls for your Azure Virtual Desktop landing zone. You may need controls for specific policies like the Payment Card Industry Data Security Standard (PCI DSS) or the Health Insurance Portability and Accountability Act of 1996 (HIPAA).
- **Defined roles:** Defined administrative, operations, and engineering roles within your organization plays a large part in defining the day-to-day operations in the Azure Virtual Desktop environment. Knowing which team is responsible for what area will help determine Azure role-based access control (RBAC) roles and configuration. Be sure to review the identity and access management section for more information. Consider creating a RACI matrix to map who owns each responsibility, then build controls into the Cloud Adoption Framework management group structure.
- **Security audit tools:** What tools and methods do you use to continually scan, and evaluate your environment for security audits, and vulnerabilities?
- **Software updates:** Define a strategy for continuous operations to keep Windows and applications current.
- **Disk encryption:** Do you have regulatory or internal security requirements to manage and maintain your own keys for encrypting VMs at rest? Are Azure Key Vault keys acceptable for encryption? Do you need advanced hardware encryption or in-guest OS encryption like BitLocker? How will data at rest or data in transit be encrypted?

- **Data protection:** How will data in the VMs be protected? You can use a tool like Azure Information Protection to protect data. Consider using antimalware tools for protection.
- **Service tags:** A service tag represents a group of IP address prefixes for an Azure service. Microsoft manages the address prefixes and automatically updates the tags as addresses change, simplifying frequent updates to network security rules. Sometimes it's necessary to have additional tags in an Azure Virtual Desktop environment for areas like chargeback, security audits, reporting, and alerts.
- **Policies:** Policies for managing your Azure Virtual Desktop environment should be defined in your Cloud Adoption Framework platform design. Include policies pertaining to security, RBAC controls, regulatory governance, and types of resources that can be deployed.
- **Resource group organization:** Organize your resource groups to facilitate good management and prevent accidental deletions, and define who can manage your environment.

Design recommendations

- **Multifactor authentication:** Multifactor authentication for all users is essential to securing desktops and company data. Use multifactor authentication in Azure Active Directory or a partner multifactor authentication tool.
- **Conditional access:** Conditional access helps you to manage risks when granting access to users in your Azure Virtual Desktop environment. Before deciding to grant access to a user, consider who the user is, how they sign in, and which device they use. See What is Azure AD Conditional Access? for an overview of conditional access and advice on best practices.
- **Enable logging:** Enable Azure Virtual Desktop service logging, host pool logging, and workspace logging for all Azure Virtual Desktop objects. For more information, see Use Log Analytics for the diagnostics feature. Enable Azure Virtual Desktop host logging and performance logging as outlined in the management and monitoring section of the Azure Virtual Desktop landing zone architecture.
- **Endpoint protection:** Microsoft strongly advises enabling a next-generation antivirus to create a protection layer and response mechanism to threats. An example is Microsoft Defender for Endpoint. It's integrated with Azure Security Center to provide a data analytics and AI approach to proactively maintain security. Other security needs like network protection, web content filtering, attack surface reduction, security baselines for VM hosts, and threat vulnerability management should be part of your Azure Virtual Desktop design. See the following section for links to Azure Virtual Desktop host security best practices.
- **Microsoft Information Protection:** Enable and configure Microsoft Information Protection to discover, classify, and protect sensitive information wherever it is.
- **Control device redirection:** Only enable what your end users need. Common devices to disable include local hard drive access and USB or port restrictions. Limiting camera redirection and remote printing can help protect company data. Disable clipboard redirection to prevent copying remote content to endpoints.
- **Policy tools:** Use group policy and a device management tools like Intune and Microsoft Endpoint Configuration Manager to maintain a thorough security and compliance practice for your desktops.
- **Patch management:** Patch management is a vital part of the overall security strategy for your environment. You need a consistent practice and deployment policy to maintain secure systems. Tools like Microsoft Endpoint Configuration Manager and partner applications can help manage patches and keep your systems up to date.
- **Screen capture:** The screen capture feature, when enabled, prevents screen information from capture on the client endpoints. Remote content is blocked or hidden in screenshots and screen shares, and from software that captures screen content. For more information, see Enable screen capture protection.
- **Security baseline:** Use a security baseline as a starting point for securing the Windows operating system. For more information, see Windows security baselines.
- **Application control:** Implement Windows Defender Application Control and AppLocker, which allows organizations to control drivers and applications that can run on Windows 10 clients.
- **Azure Security Center:** Enable Security Center to help maintain security compliance and alerting within

your environment.

- **Microsoft Secure Score:** Microsoft Secure Score provides recommendations and best practice advice for increasing your security posture and securing surrounding infrastructure with documented best practices.
- **Disk encryption:** Enable Azure Disk Encryption for your VMs. This option is configured by default with Azure-provided keys. In many cases, this configuration is acceptable to security teams and auditors. However, if you have a security practice or regulatory requirement that requires you to maintain your own keys, you can implement that practice for Azure Virtual Desktop VMs.
- **Key Vault:** Enable Key Vault to protect security principal accounts and encryption keys.
- **Security best practices:** Review security best practices for Azure Virtual Desktop as a starting point to security within your environment, and implement as appropriate.
- **Azure Virtual Desktop service and internet traffic routing and inspection:** By using reverse connect, built into the Azure Virtual Desktop platform, VMs do not need a public IP. VMs communicate outbound securely to Azure Virtual Desktop service URLs over port 443. It's good practice to enable Azure Firewall or a partner firewall appliance for traffic logging, routing, or inspection. Having a web proxy filter to monitor and log internet traffic is also recommended.
- **Azure Virtual Desktop metadata:** A good resource group design for Azure Virtual Desktop can help protect against accidental deletion of workspace and host pool objects, can separate VM machine types, and can allow for administrators from different departments. Outside the Cloud Adoption Framework best practice for RBAC, security controls and landing zone design, here is a sample resource group structure for Azure Virtual Desktop.

> **NOTE**
> This structure should be duplicated for each region you deploy into.

```
    - Networking:  Generally created as part of the Cloud Adoption Framework Landing zone
    - Azure Virtual Desktop Service Objects:  Separate Azure Virtual Desktop Service Objects from Host Pool
VMs.  Service objects include Workspaces, Host Pools and RemoteApp/Desktops App groups. Create a resource
group for these objects.
    - Storage:  If not already created as part of Cloud Adoption Framework, create a resource group for
storage accounts
    - Images:  Create a resource group for custom VM images
    - Host Pools:  Create a resource group for each host pool
    - Basic Structure
        - Subscription
            - rg-wu2-network-services
            - rg-wu2-wvd-storage
            - rg-wu2-wvd-service-objects
            - rg-wu2-wvd-images
            - rg-wu2-wvd-hostpool1
            - rg-wu2-wvd-hostpool2
            - rg-wu2-wvd-hostpool3
```

Azure Virtual Desktop host operating system security

In addition to service level logging for Azure Virtual Desktop, administrators need to have a security strategy inside the guest operating system. Microsoft recommends security tools such as:

- Microsoft Defender for Endpoint: OS-level antivirus and antimalware
- Threat protection (Windows 10): Threat protection overview and details
- Azure Monitor agents and Azure Monitor deployment at scale with Azure Policy: Capture guest logs and performance metrics
- Guest configuration extension for local machines: Monitor, alert, and track guest changes and audit reports
- Azure Monitor dependency extension: Troubleshoot guest connections, logs traffic flows, and configuration

- Enable screen capture protection: Protects from remote capture of data
- Azure Security Center: Security audits, regulatory compliance scanning, policy compliance
- Windows security baselines

For more information on Azure Virtual Desktop best practices, see Session host security best practices. For a detailed list of best practices for Azure VMs, see Security recommendations for virtual machines in Azure.

Manage secrets in a secure way

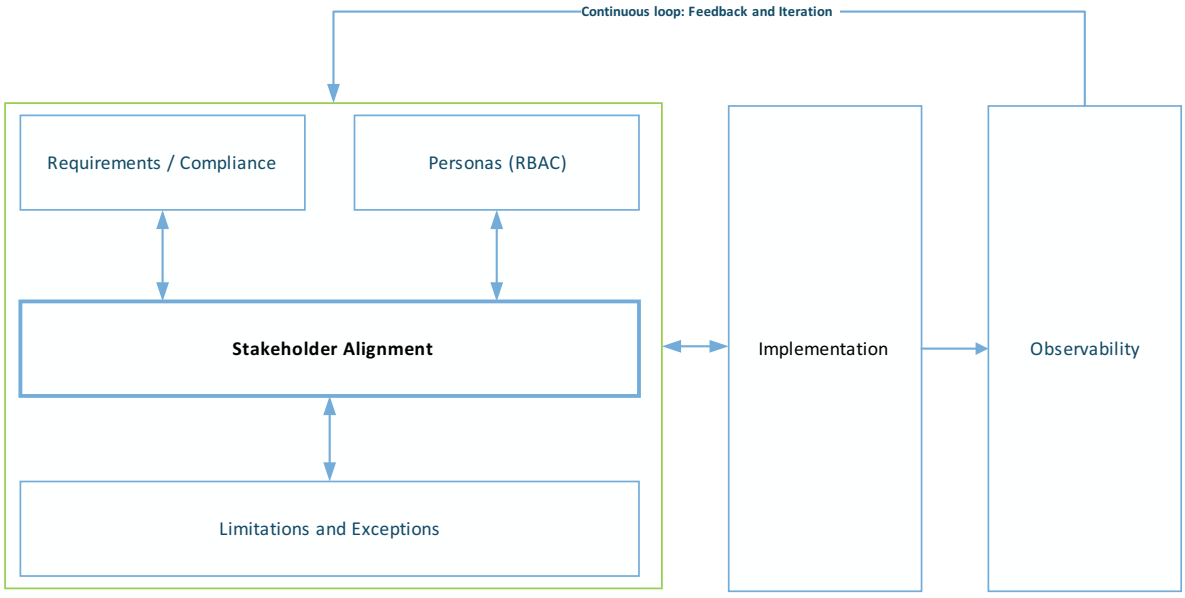
5/24/2021 • 10 minutes to read • Edit Online

The management of company sensitive secrets, keys, and certificates (and associated private keys) aligns with a critical process to improve your security posture. This guidance offers several security best practices on how to manage each of these sensitive assets. Some of these best practices might also align with your compliance or attestation requirements.

Before you continue, it's highly recommended to document the following items within your organization so your alignment with these best practices can be put into action:

1. Limitations and exceptions: understand and document how various security configurations have a direct effect to performance and cost. Ensure your documentation aligns with your overall business requirements. Next, review each business requirement to ensure all stakeholders agree and accept the overhead and complexities various security requirements will place on your secrets management implementation.
2. Identify your compliance and attestation requirements. For example: Center for Internet Security (CIS), National Institute of Standards and Technology (NIST), Health Insurance Portability and Accountability Act (HIPAA), HITRUST, cybersecurity maturity model certification (CMMC), protected B, medium integrity, and medium availability (pbmm), and federal information processing standard (FIPS). With respect to implementation, several of these compliance requirements might be implemented through built-in governance policy-based rule sets within Azure services. For example, there are various security controls in Azure Policy that can be applied to Azure Key Vault.
3. Identify the personas and their access requirements to the various Azure services. For example, developers and applications might only require read access to the secrets stored in Azure Key Vault within a development environment. On the other hand, operators or administrators will require full management access to all key vaults within each environment.
4. The overall *observability methodology* used by your organization to ensure your logging, monitoring, auditing, and alerting configurations are aligned with business requirements.

The following illustration is a high-level logical representation of these processes:



Anticipate limitations and exceptions

To anticipate exceptions to best practices will be difficult and these exceptions might occur within various phases of your cloud journey. Ensure that exceptions and their risks to the business are documented and accepted across all stakeholders. Each Azure service will have limitations or quotas documented that might affect your Azure workload. Review each of the service limits that applies to your workload to ensure they are documented and align with stakeholder expectations.

There might be other limitations and exceptions that will require documentation and require alignment with your business requirements. For example, when you perform a backup of keys, secrets, and certifications and their number of object versions allowed within Azure Key Vault.

> **IMPORTANT**
>
> The best practice to apply for each exception encountered is: *mitigate as much as possible to ensure your risks are minimized*. If each risk can't be mitigated, ensure your *observability methodology* will enable you to identify any nefarious activities as early as possible with the information that will support your:
>
> - Time to be notified and time to acknowledge, also known as MTTA
> - Time to detection, also known as MTTD
> - Root cause analysis and compromise recovery, also known as MTTR processes.

External configuration store pattern

Most of these best practices across your enterprise and workloads deployed to the cloud will use a secret store. To understand the external configuration store cloud design pattern will not only provide context, it will provide insights to various issues and considerations that might affect your workload architectural design.

For example, there are rare exceptions to the security best practice of using Azure Key Vault to store secrets, such as connection strings. These exceptions might be revealed during the development implementation phase when conducting performance and load testing. Ensure your business stakeholders state that performance is critical to the workload.

Understanding alternatives to using Azure Key Vault and adhering to the external configuration store design pattern will ensure alignment with best practices and compliance requirements. An alternative might be to have secrets stored closer to the application's compute layer, for example, Azure App Service application settings that will only be accessed by designated and approved personas for the connections that require lower latency.

> **IMPORTANT**
>
> The overall best practice on how to implement the external configuration store cloud design pattern is to ensure:
>
> - Alignment with your business and compliance requirements
> - Ensure at a minimum that the appropriate personas, their scope, and their permissions adhere to the least-privilege model
> - Implement observability, for example: logging, monitoring, and alerting
> - If you use Azure Key Vault:
>   - Implement the security baseline for Azure Key Vault
>   - Enable Azure Defender for Azure Key Vault
>   - Enable logging for Azure Key Vault

To learn more, see:

- Azure subscription and service limits
- External configuration store pattern

- Understanding separation of duties within NIST
- Applying Azure Policy's NIST SP 800-53 R4 controls, including separation of duties
- Start with the Azure Key Vault security baseline
- Review and validate against the Azure Key Vault authentication fundamentals
- Azure Policy regulatory compliance controls for Azure Key Vault
- Observability using Azure Key Vault logging

Manage keys in a secure way

There are different types of keys that can be used within various Azure services:

- Keys that are cryptographic, such as encryption-at-rest or encryption-in-transit.
- Keys used for connectivity, also known as a token for authentication (AuthN) and authorization (AuthZ) to various Azure services.

Keys used for cryptographic keys: Encryption-at-rest

Read the encryption security fundamentals document for an encryption overview in Azure.

The two types of keys for encryption-at-rest are:

- Platform-managed keys (PMK)
- Customer-managed keys (CMK), also known as bring-your-own-key (BYOK)

As mentioned above, document your various business requirements. For example, what are your encryption-at-rest requirements with respect to key management? Ask yourself: *Are there any limitations or exceptions that I need to be made aware of?*

You might find limitations or exceptions with various Azure services that might not support customer-managed keys (CMK). Validate and document these items with your business stakeholders for each Azure service within your workload.

> **IMPORTANT**
> When CMKs are not required, use PMKs due to operational simplicity as they are managed (including rotation) by the platform.

If your business requires customer-managed keys for encryption-at-rest, ensure you document the various Azure services that support this and their limitations or their release status, such as preview or GA. For example, Azure Monitor supports customer-managed keys and the documents any limitations.

> **IMPORTANT**
> - When you use customer-managed keys, rotation of each key should be conducted on a frequent interval for each of the Azure services in your workloads.
> - If backups are encrypted with your CMKs, ensure your key rotation allows these backups to be restored successfully.
> - If your compliance requirements need backups to be stored that span several years, ensure that the keys used to encrypt said backups are available and can successfully restore these backups.

To learn more about keys used for encryption, see:

- Azure Monitor customer-managed key
- Customer-managed keys for Azure Storage encryption
- Replicate machines with CMK enabled disks

- Configure customer-managed keys for your Azure Cosmos DB account with Azure Key Vault

Other security-based cloud design patterns

For the following sections on keys (used as tokens) and secrets, there are several cloud design patterns that address security best practices and should be considered when designing your workloads. These patterns are:

- **Valet key pattern:** Use a token that provides clients with restricted direct access to a specific resource, in order to offload data transfer from the application.
- **Command and query responsibility segregation (CQRS) pattern:** The command and query responsibility segregation (CQRS) pattern separates read and update operations for a data store. Implementing CQRS in your application can maximize its performance, scalability, and security. The flexibility created by migrating to CQRS allows a system to better evolve over time and prevents update commands from causing merge conflicts at the domain level.

> **IMPORTANT**
> For each of your workloads, identify various established cloud design patterns that address each of the problem domains to ensure alignment with your business security requirements.

To learn more, see:

- Valet pattern
- CQRS pattern

Keys used as connectivity or tokens: Invocation or within connection strings

The other type of keys are used for various Azure services for connectivity or tokens, for example, Azure Storage or Azure Service Bus access keys.

There are limitations or exceptions for the keys of various Azure services. Understanding and aligning with your business requirements will validate the best choice or practice below. For example, Azure Storage has support for shared access policies, shared access signatures, Azure AD, and so on.

Understanding each of these options and documenting them to your business requirements will guide you through this decision process.

Additionally, many Azure services support managed identities for authentication. What are managed identities?

> **IMPORTANT**
> - Use Azure AD managed identities when your use cases support the capabilities of managed identities for the various Azure services your solution will be composed of.
> - If Azure AD managed identities is not suitable for your use cases nor available to an Azure service, following a "least-privilege model" by using stored access policies or shared-access signatures (SAS) with explicit access and time-bound values.

Several Azure services might not have the features to use a stored access signature; however, these services might have the following policy features: manage, send, listen. For example, Azure Service Bus, Azure Event Hubs, Azure IoT Hub have these types of security access policies in addition to using Azure Active Directory.

> **IMPORTANT**
> Following a "least-privilege model", when defining and configuring a policy will only provide at the specified scope the necessary operations needed.

Several Azure services use a key to allow an invocation to be performed. For example, non-anonymous Azure Functions have an access code (also known as a key).

> **IMPORTANT**
> Ensure the keys used for invocation have a defined purpose. Create several keys for their intended purpose and renew or regenerate these keys on a frequent interval or when the key is compromised.

Lastly, various Azure services have features that support key rotation. Understanding shared responsibility in the cloud will support your documentation for various workloads.

> **IMPORTANT**
> Within each of your workloads, periodically rotate your connection string keys for each of the Azure services. Use automation and an external configuration store, such as Azure Key Vault so that there is minimized disruption. When not using an external configuration store, each of these Azure services have two keys so that they can be used to minimize disruption during your maintenance window.

To learn more about keys used to invocate or within connection strings, see:

- Define a stored access policy on Azure Storage
- Manage storage account keys with Key Vault and the Azure CLI
- Grant limited access to Azure Storage resources using shared access signatures
- Azure Policy regulatory compliance controls for Azure Service Bus
- Samples: `Azure.Security.KeyVault.Administration` using .NET
- Managing CMKs with Azure Service Bus
- Automate the rotation of secret for resources that use one set of augmentation credentials
- Automate the rotation of secret for resources that use two sets of authentication credentials

Manage your secrets in a secure way

Secrets store sensitive information and are typically passwords, credentials, or fully qualified connection strings.

Passwords and secrets are managed outside of application artifacts, using tools like Azure Key Vault. API keys, database connection strings, and passwords need to be stored in a secure store and not within the application code or configuration. Using a centrally managed external configuration store simplifies operational tasks like key rotation in addition to improving overall security.

Applying automation will improve operational efficiencies and using credentials for various automated runbooks will need to be aligned with your business requirements. For example, a runbook within an Azure Automation account that purges expired data from storage accounts needs to have credentials to perform that action.

> **IMPORTANT**
> Use an external configuration store, such as Azure Key Vault, to store sensitive information. For example, fully qualified connection strings or passwords. Where possible, automate rotation of secrets and store them in Azure Key Vault so that there is minimized disruption.

To learn more, see:

- Azure Service Bus authentication and authorization
- Azure architecture blog: managing and rotating secrets with Azure Key Vault

- Automate the rotation of secret for resources that use one set of authentication credentials
- Automate the rotation of secret for resources that use two sets of authentication credentials
- Using credentials in Azure Automation runbooks

Renew certificates

Certificates are used for encryption, authorize automation actions, device management, and many other functionalities. Certificates are time bound and when they expire without a robust operational model, these expired certificates cause disruption to your workloads.

There are limitations and exceptions, such as authorizing automation actions within Azure Automation accounts when using Desired State Configuration (DSC), and where these certificates are stored.

> **IMPORTANT**
> Align with the authorized personas, use central storage of these certificates, such as Azure Key Vault, and ensure the expiry dates of SSL/TLS certificates are monitored and there are renewal processes in place that align with your business requirements.

To learn more, see:

- About Azure Key Vault certificates
- Renew your Azure Key Vault certificates
- Manage certificates in Azure Automation

Next steps

Secure DevOps

Security toolchain

5/24/2021 • 2 minutes to read • Edit Online

Commonly used tools in Azure security:

- Azure Active Directory
- Azure role-based access control (RBAC)
- Azure Defender
- Azure DDoS protection
- Azure Firewall
- Azure Monitor
- Azure Policy
- Azure Policy samples
- Azure Security Center
- Azure Security Benchmark
- Azure Sentinel
- Confidential computing on Azure

Additional resources:

- Microsoft zero-trust blog
- Implementing a zero-trust security model at Microsoft
- Microsoft security blog
- Microsoft security response center (MSRC) blog

Define a security strategy

5/24/2021 • 23 minutes to read • Edit Online

The ultimate objectives for a security organization don't change with adoption of cloud services, but how those objectives are achieved will change. Security teams must still focus on reducing business risk from attacks and work to get confidentiality, integrity, and availability assurances built into all information systems and data.

Modernize your security strategy

Security teams need to modernize strategies, architectures, and technology as the organization adopts cloud and operates it over time. While the size and number of changes can initially seem daunting, the modernization of the security program allows security to shed some painful burdens associated with legacy approaches. An organization may temporarily operate with legacy strategy and tooling, but this approach is difficult to sustain with the pace of change in cloud and the threat environment:

- Security teams are likely to be left out of cloud adoption decision making if they take a legacy mindset of "arms-length" security where the answer always starts with "no" (instead of working together with IT and business teams to reduce risk while enabling the business).
- Security teams will have a difficult time detecting and defending against cloud attacks if they use only legacy on-premises tooling and exclusively adhere to network perimeter only doctrine for all defenses and monitoring. Defending at cloud scale mandates the use of cloud native detection and automation capabilities and the introduction of an identity perimeter to help monitor and protect cloud and mobile assets. For more information, consult these references:
  - The Microsoft identity platform helps you incorporate modern authentication and authorization mechanisms into your applications.
  - Azure Sentinel provides cloud native security analytics and threat intelligence across your organization.

Because this transformation can be significant, we recommend that security teams take an agile approach to modernizing security that rapidly modernizes the most critical aspects of the strategy and then continuously improve incrementally afterwards.

Security of the cloud and from the cloud

As your organization adopts cloud services, security teams will work toward two main objectives:

- **Security *of* the cloud (securing cloud resources):** Security should be integrated into the planning and operation of cloud services to ensure that those core security assurances are consistently applied across all resources.
- **Security *from* the cloud (using the cloud to transform security):** Security should immediately start planning and thinking about how to use cloud technologies to modernize security tools and processes, particularly natively integrated security tools. Increasingly, security tools are being hosted in the cloud and providing capabilities that are difficult or impossible to do in an on-premises environment.

Many organizations start by treating cloud resources as another *virtual datacenter*, an effective starting point for security of the cloud. As organizations modernize using security from the cloud, most will find themselves quickly outgrowing this model of thinking. Securing a software-defined datacenter using cloud-hosted tools enables capabilities beyond what on-premises models can offer:

- Rapid enablement and scaling of security capabilities.
- Highly effective asset inventory and security configuration hygiene discovery.

- Continuous assessment of the organization's security posture and controls, by deploying for example Azure Security Center.
- Vastly improved threat detection that uses vast repositories of threat intelligence and the nearly unlimited processing and storage capabilities of the cloud, such as that enabled by Azure Sentinel.

The right level of security friction

Security naturally creates friction that slows down processes, it's critical to identifying which elements are healthy in your DevOps and IT processes and which are not:

- **Healthy friction:** Much like the resistance in exercise makes a muscle stronger, integrating the right level of security friction strengthens the system or application by forcing critical thinking at the right time. This typically takes the form of considering how and why an attacker may try to compromise an application or system during design (known as threat modeling), and reviewing, identifying, and ideally fixing potential vulnerabilities an attacker can exploit in software code, configurations, or operational practices.
- **Unhealthy friction:** Impedes more value than it protects. This often happens when security bugs generated by tools have a high false positive rate (such as false alarms) or when the effort to discover or fix security issues far exceeds the potential impact of an attack.

Standalone and integrated responsibilities

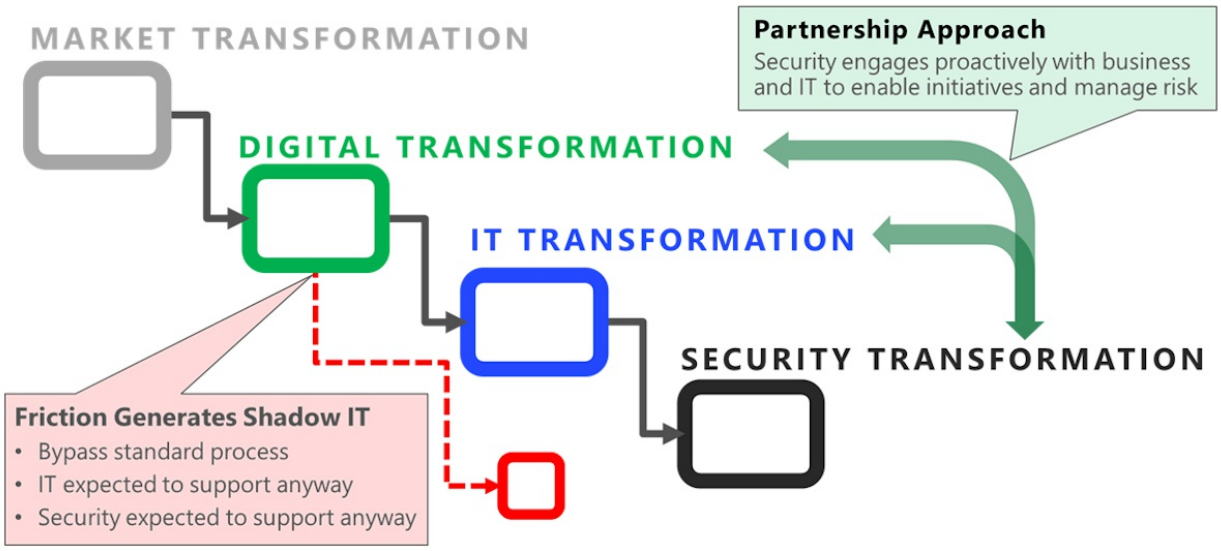
Providing confidentiality, integrity, and availability assurances requires security experts to operate dedicated security functions and work closely with other teams in the organization:

- **Unique security functions:** Security teams perform independent functions that are not found elsewhere in the organization, such as security operations, vulnerability management (such as for virtual machines, containers), and other functions.
- **Integrating security into other functions:** Security teams also serve as subject matter experts to other teams and functions in the organization who are driving business initiatives, assessing risk, designing or developing applications, and operating IT systems. Security teams advise these teams with expertise and context on attackers, attack methods and trends, vulnerabilities that could allow unauthorized access, and options for mitigation steps or workarounds and their potential benefits or pitfalls. This function of security resembles that of a quality function as it will be woven into many places large and small in support of a single outcome.

Executing on these responsibilities while keeping up with the rapid pace of change in the cloud and the transformation of business requires security teams to modernize their tools, technologies, and processes.

Transformations, mindsets, and expectations

Many organizations are managing a chain of multiple simultaneous transformations in the organization. These internal transformations typically start because nearly all external markets are transforming to meet new customer preferences for mobile and cloud technologies. Organizations often face the competitive threat of new startups and the digital transformation of traditional competitors who can disrupt the market.



The internal transformation process typically includes:

- **Digital transformation** of the business to capture new opportunities and stay competitive against digital native startups.
- **Technology transformation** of the IT organization to support the initiative with cloud services, modernized development practices, and related changes.
- **Security transformation** to both adapt to the cloud and simultaneously address an increasingly sophisticated threat environment.

Internal conflict can be costly

Change creates stress and conflict, which can grind decision making to a halt. This is especially true in security where accountability for security risk is often misplaced on the subject matter experts (security teams), rather than on the owners of the assets (business owners) that are accountable for business outcomes and all other risk types. This misplaced accountability often happens because all stakeholders incorrectly view security as a technical or absolute problem to be solved, rather than a dynamic ongoing risk like corporate espionage and other traditional criminal activities.

During this time of transformation, leadership of all teams must work actively to reduce conflict that can both derail critical projects and incentivize teams to bypass security risk mitigation. Internecine conflict between teams can result in:

- **Increased security risk** such as avoidable security incidents or increased business damage from attacks (particularly when teams get frustrated by security and bypass normal processes or when outdated security approaches are easily bypassed by attackers).
- **Negative impact on the business or mission** such as when business processes aren't enabled or updated fast enough to meet market needs (often when security processes hold up key business initiatives).

It's critical to stay aware of relationship health within and between teams to help them navigate the shifting landscape that could leave valuable team members insecure and unsettled. Patience, empathy, and education on these mindsets and the positive potential of the future will help your teams better navigate this period, driving good security outcomes for the organization.

Leaders can help drive culture changes with concrete proactive steps like:

- Publicly modeling the behavior they expect of their teams.
- Being transparent about the challenges of the changes, including highlighting their own struggles to adapt.
- Regularly reminding teams of the urgency and importance of modernizing and integrating security.

Cybersecurity resilience

Many classic security strategies have been focused solely on preventing attacks, an approach that is insufficient

for modern threats. Security teams must ensure their strategy goes beyond this and also enables rapid attack detection, response, and recovery to increase resilience. Organizations must assume that attackers will compromise some resources (sometimes called *assume breach*) and work to ensure that resources and technical designs are balanced between attack prevention and attack management (rather than the typical default approach of only attempting to prevent attacks).

Many organizations are already on this journey because they have been managing the steady rise in volume and sophistication of attacks in recent years. This journey often starts with the first major incident, which can be an emotional event where people lose their prior sense of invulnerability and safety. While not as severe as a loss of life, this event can trigger similar emotions starting with denial and ultimately ending in acceptance. This assumption of "failure" may be difficult for some to accept at first, but it has strong parallels to the well-established "fail-safe" engineering principle and the assumption allows your teams to focus on a better definition of success: resilience.

The functions of the NIST cybersecurity framework serve as a useful guide on how to balance investments between the complementary activities of identify, protect, detect, respond, and recover in a resilient strategy.

More on cybersecurity resilience and the ultimate goals of cybersecurity controls is discussed in How do you keep your organization's risk down.

How the cloud is changing security

Shifting to the cloud for security is more than a simple technology change, it's a generational shift in technology akin to moving from mainframes to desktops and onto enterprise servers. Successfully navigating this change requires fundamental shifts in expectations and mindset by security teams. Adopting the right mindsets and expectations reduces conflict within your organization and increases the effectiveness of security teams.

While these could be part of any security modernization plan, the rapid pace of change in the cloud makes adopting them an urgent priority.

- **Partnership with shared goals.** In this age of fast paced decisions and constant process evolution, security can no longer adopt an "arms-length" approach to approving or denying changes to the environment. Security teams must partner closely with business and IT teams to establish shared goals around productivity, reliability, and security and work collectively with those partners to achieve them.

  This partnership is the ultimate form of "shift left"—the principle of integrating security earlier in the processes to make fixing security issues easier and more effective. This requires a culture change by all involved (security, business, and IT), requiring each to learn the culture and norms of other groups while simultaneously teaching others about their own.

  Security teams must:

  - **Learn** the business and IT objectives and why each is important and how they are thinking about achieving them as they transform.
  - **Share** why security is important in the context of those business goals and risks, what other teams can do to meet security goals, and how they should do it.

  While not an easy task, it's essential for sustainably securing the organization and its assets. This partnership will likely result in healthy compromises where only the minimum security, business, and reliability goals may be met initially, but incrementally improve steadily over time.

- **Security is an ongoing risk, not a problem.** You can't "solve" crime. At its core, security is just a risk management discipline, which happens to be focused on malicious actions by humans rather than natural events. Like all risks, security is not a problem that can be fixed by a solution, it's a combination of the likelihood and impact of damage from a negative event, an attack. It's most comparable to traditional corporate espionage and criminal activities where organizations face motivated human attackers who have financial incentive to successfully attack the organization.

- **Success in either productivity or security requires both.** An organization must focus on both security and productivity in today's "innovation or become irrelevant" environment. If the organization is not productive and driving new innovation, it may lose competitiveness in the marketplace that causes it to weaken financially or eventually fail. If the organization is not secure and loses control of assets to attackers, it may lose competitiveness in the marketplace that causes it to weaken financially and eventually fail.

- **Nobody's perfect.** No organization is perfect at adopting the cloud, not even Microsoft. Microsoft's IT and security teams grapple with many of the same challenges that our customers do such as figuring out how to structure programs well, balancing supporting legacy software with supporting cutting edge innovation, and even technology gaps in cloud services. As these teams learn how to better operate and secure the cloud, they are actively sharing their lessons learned via documents like this along with others on the IT showcase site, while continuously providing feedback to our engineering teams and third-party vendors to improve their offerings.

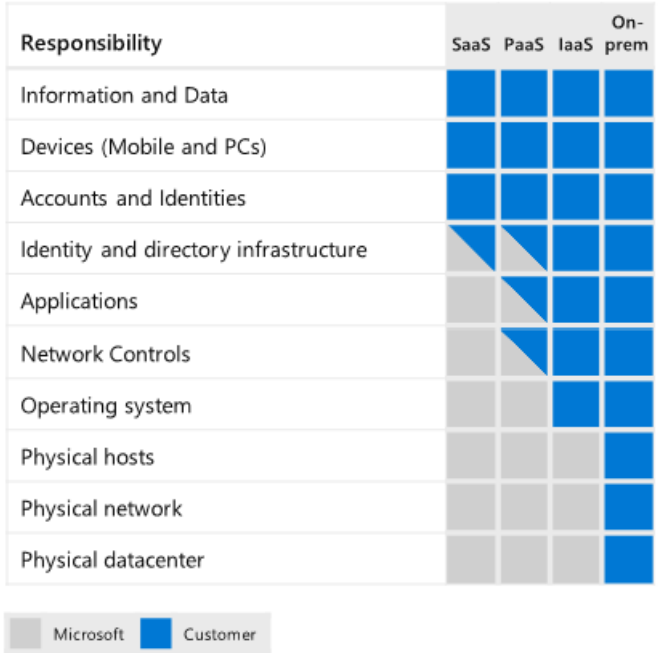
  Based on our experience, we recommend that teams are held to a standard of continuous learning and improvement rather than a standard of perfection.

- **Opportunity in transformation.** It's important to view digital transformation as a positive opportunity for security. While it's easy to see the potential downsides and risk of this change, it's easy to miss the massive opportunity to reinvent the role of security and earn a seat at the table where decisions are made. Partnering with the business can result in increased security funding, reduce wasteful repetitive efforts in security, and make working in security more enjoyable as they will be more connected to the organization's mission.

Adopting the shared responsibility model

Hosting IT services in the cloud splits the operational and security responsibilities for workloads between the cloud provider and the customer tenant, creating a de facto partnership with shared responsibilities. All security teams must study and understand this shared responsibility model to adapt their processes, tools, and skill sets to the new world. This will help avoid inadvertently creating gaps or overlaps in your security posture resulting in security risks or wasted resources.

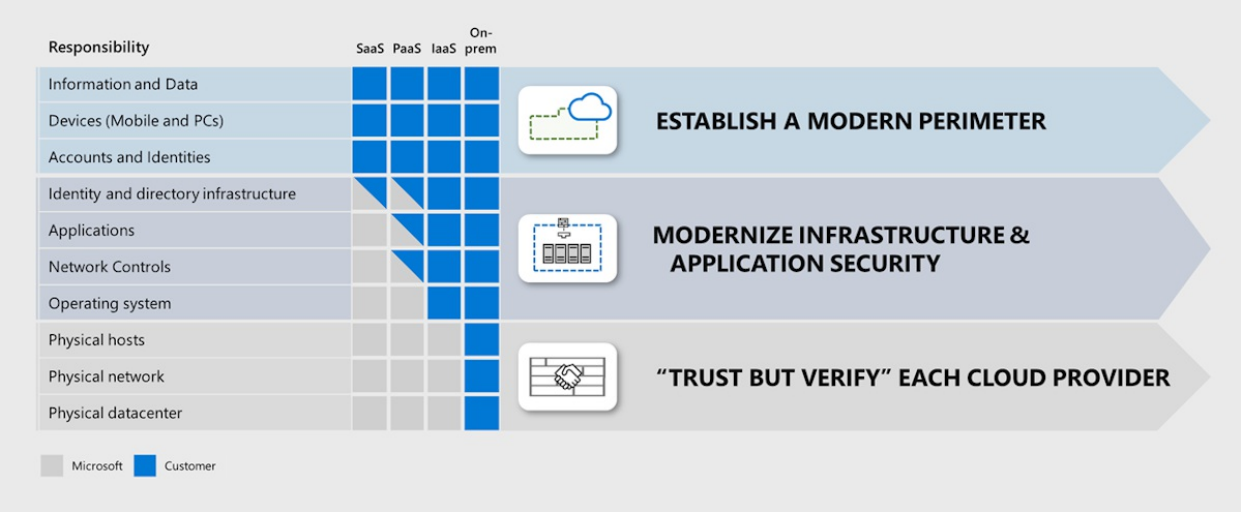
This diagram illustrates how security responsibilities will be distributed between cloud vendors and cloud customer organizations in a de facto partnership:



As there are different models of cloud services, the responsibilities for each workload will vary depending on whether it's hosted on software as a service (SaaS), platform as a service (PaaS), infrastructure as a service (IaaS), or in an on-premises datacenter.

Building security initiatives

This diagram illustrates the three primary security initiatives that most security programs should follow to adjust their security strategy and security program goals for the cloud:



Building a resilient security posture in the cloud requires several parallel complementary approaches:

- **Trust but verify:** For responsibilities performed by the cloud provider, organizations should take a "trust but verify" approach. Organizations should evaluate the security practices of their cloud providers and the security controls they offer to ensure the cloud provider meets the security needs of the organization.
- **Modernize infrastructure and application security:** For technical elements under the organization's control, prioritize modernizing security tooling and associated skill sets to minimize coverage gaps for securing resources in the cloud. This is composed of two different complementary efforts:
  - **Infrastructure security:** Organizations should use the cloud to modernize their approach to protecting and monitoring the common components used by many applications, such as operating systems, networks, and container infrastructure. These cloud capabilities can often including managing infrastructure components across both IaaS and on-premises environments. Optimizing this strategy is important because this infrastructure is a dependency of the applications and data that run on it, which often enable critical business processes and store critical business data.
  - **Application security:** Organizations should also modernize the way they secure the unique applications and technology that is developed by or for their organization. This discipline is changing rapidly with the adoption of agile DevOps processes, the increasing use of open-source components, and introduction of cloud APIs and cloud services to replace application components or interconnect applications.

    Getting this right is critical because these applications often enable critical business processes and store critical business data.
  - **Modern perimeter:** Organizations should have a comprehensive approach for protecting data across all workloads, organizations should establish a modern perimeter of consistent, centrally managed identity controls to protect their data, devices, and accounts. This is heavily influenced by a zero trust strategy discussed in detail in Module 3 of the CISO workshop.

Security and trust

The use of the word *trust* in security can be confusing. This documentation refers to it in two ways that illustrate useful applications of this concept:

- Zero trust is a common industry term for a strategic approach to security that assumes a corporate or intranet network is hostile (worthy of *zero trust*) and designs security accordingly.
- Trust but verify is an expression that captures the essence of two different organizations working together toward a common goal despite having some other potentially divergent interests. This concisely captures many of the nuances of the early stages of partnering with a commercial cloud provider for organizations.

A cloud provider and its practices and processes can be accountable to meet contractual and regulatory requirements and could earn or lose trust. A network is a nonliving connection that cannot face consequences if it's used by attackers (much like you cannot hold a road or a car accountable for criminals using them).

How cloud is changing security relationships and responsibilities

As with previous transitions to a new generation of technology like desktop computing and enterprise server computing, the shift to cloud computing is disrupting long-established relationships, roles, responsibilities, and skill sets. The job descriptions we have become accustomed to over the last few decades do not cleanly map to an enterprise that now includes cloud capabilities. As the industry collectively works to normalize a new model, organizations will have to focus on providing as much clarify as possible to help manage the uncertainty of ambiguity during this period of change.

Security teams are affected by these changes in the business and technology they support as well as their own internal modernization efforts to better orient to threat actors. Attackers are actively evolving to constantly search for the easiest weak points to exploit in the people, process, and technology of the organization and security must develop capabilities and skills to address these angles.

This section describes the key relationships that frequently change on the journey to the cloud, including lessons learned on minimizing risk and embracing the opportunities to improve:

- **Between security and business stakeholders:** Security leadership will need to increasingly partner with business leaders to enable organizations to reduce risk. Security leaders should support business decision making as security subject matter expert (SMEs) and should strive to grow into trusted advisors to these business leaders. This relationship will help ensure business leaders consider security risks while making business decisions, inform security of business priorities, and help ensure security investments are prioritized appropriately alongside other investments.

- **Between security leadership and team members:** Security leadership should take these insights from business leadership back to their teams to guide their investment priorities.

  By setting a tone of cooperation with business leaders and their teams rather than a classic "arms-length" relationship, security leaders can avoid an adversarial dynamic that impedes both security and productivity goals.

  Security leaders should strive to provide clarity to their team on how to manage their daily decisions on productivity and security tradeoffs as this may be new to many on their teams.

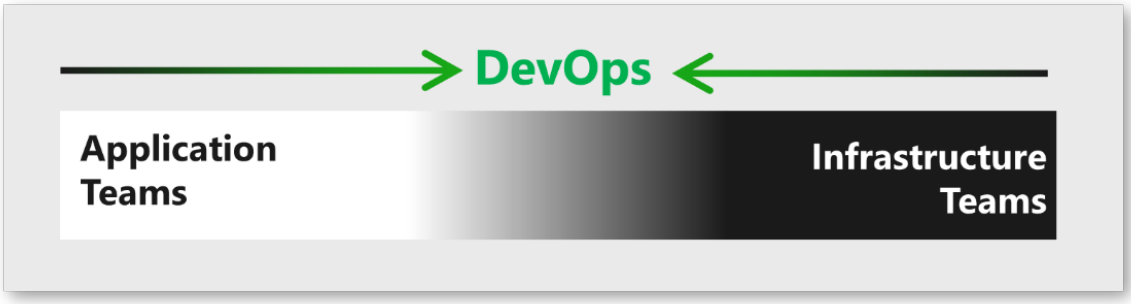
- **Between application and infrastructure teams (and cloud providers):** This relationship is undergoing significant changes because of multiple trends in the IT and security industry aimed at increasing innovation speed and developer productivity.

  The old norms and organizational functions have been disrupted, but new norms and functions are still emerging, so we recommend accepting the ambiguity, keeping up with current thinking, and experiment with what works for your organizations until it does. We don't recommend adopting a wait-and-see approach in this space because it might put your organization at a major competitive disadvantage.

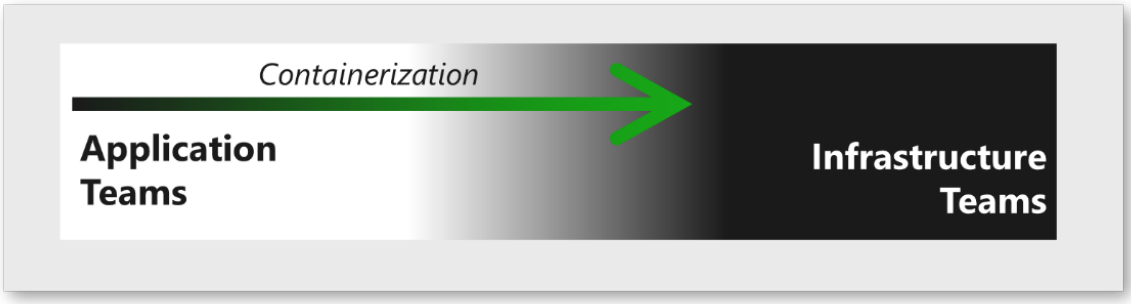
  These trends are challenging the traditional norms for roles and relationships of applications and

infrastructure:

- **DevOps-fusing disciplines:** In its ideal state, this effectively creates a single highly functional team that combines both sets of subject matter expertise together to rapidly innovate, release updates, and resolve issues (security and otherwise). While this ideal state will take some time to achieve and the responsibilities in the middle are still ambiguous, organizations are already reaping some benefits of rapid releases because of this cooperative approach. Microsoft recommends integrating security into this cycle to help learn those cultures, share security learnings, and work toward a common goal of rapidly releasing secure and reliable applications.



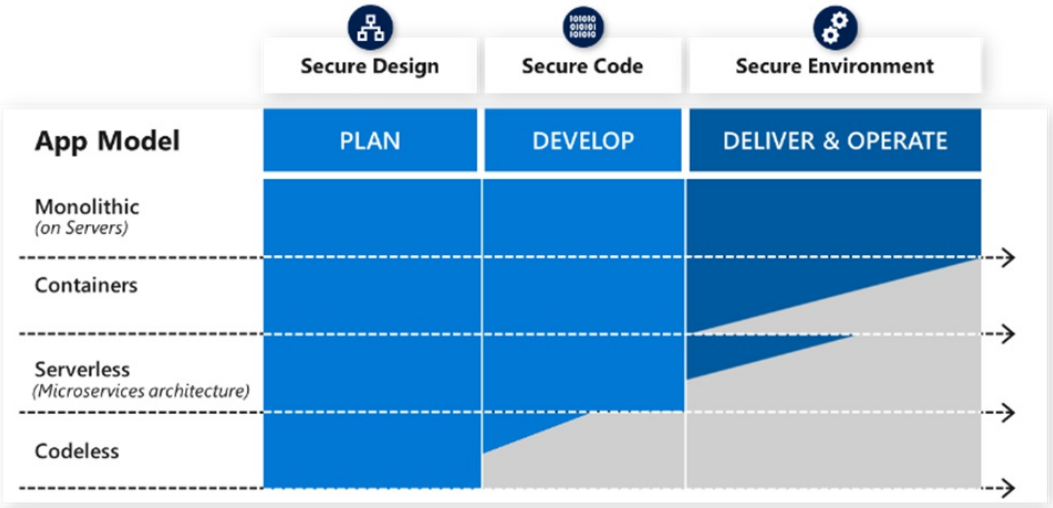
- **Containerization becoming a common infrastructure component:** Applications are increasingly being hosted and orchestrated by technologies like Docker, Kubernetes, and similar technologies. These technologies simplify development and release by abstracting many elements of the setup and configuration of the underlying operating system.



While containers began as an application development technology managed by development teams, it's becoming a common infrastructure component that is increasingly shifting to infrastructure teams. This transition is still in progress at many organizations, but it's a natural and positive direction many of the current challenges will be best solved with traditional infrastructure skill sets like networking, storage, and capacity management.

Infrastructure teams and security team members that support them should be provided with training, processes, and tooling to help manage, monitor, and secure this technology.

- **Serverless and cloud application services:** One of the dominant trends in industry right now is reducing the amount of time and development work required to build or update applications.



Developers are also increasingly using cloud services to:

- Run code instead of hosting applications on virtual machines (VMs) and servers.
- Provide application functions instead of developing their own components. This has led to a *serverless* model that uses existing cloud services for common functions. The number and variety of cloud services (and their pace of innovation) has also exceeded the ability of security teams to evaluate and approve the use of those services, leaving them to choose between allowing developers to use any service, attempting to prevent the development teams from using unapproved services, or trying to find a better way.
- **Codeless applications and Power Apps:** Another emerging trend is the use of codeless technologies like Microsoft Power Apps. This technology enables people without coding skills to create applications that achieve business outcomes. Because of this low friction and high value potential, this trend has the potential to rise in popularity quickly and security professionals would be wise to rapidly understand its implications. Security efforts should be focused on the areas where a human could make a mistake in the application, namely the design of the application and asset permissions via threat modeling the application components, interactions/relationships, and role permissions.

- **Between developers and open-source component authors:** Developers are also increasing efficiency by using open-source components and libraries instead of developing their own components. This brings value through efficiency, but also introduces security risks by creating an external dependency and a requirement to properly maintain and patch those components. Developers are effectively assuming the risk of security and other bugs when they use these components and have to ensure there is a plan to mitigate them at the same standards as code they would develop.

- **Between applications and data:** The line between security of data and applications is becoming blurred in places and new regulations are creating a need for closer cooperation between data/privacy teams and security teams:

  - **Machine learning algorithms:** Machine learning algorithms are similar to applications in that they are designed to process data to create an outcome. The key differences are:

    - **High-value machine learning:** Machine learning often confers a significant competitive advantage and is often considered sensitive intellectual property and a trade secret.

    - **Sensitivity imprint:** Supervised machine learning is tuned using data sets, which imprints characteristics of the dataset on the algorithm. Because of this, the tuned algorithm may be considered sensitive because of the dataset used to train it. For example, training a machine learning algorithm to find secret army bases on a map using a dataset of secret army bases would make it a sensitive asset.

> **NOTE**
>
> Not all examples are obvious, so it's critical to bring a team together with the right stakeholders from data science teams, business stakeholders, security teams, privacy teams, and others. These teams should have a responsibility to meet common goals of innovation and responsibility. They should address common issues such as how and where to store copies of data in insecure configurations, how to classify algorithms, as well as any concerns of your organizations.
>
> Microsoft has published our principles of responsible AI to guide our own teams and our customers.

    - **Data ownership and privacy:** Regulations like GDPR have increased the visibility of data issues and applications. Application teams now have the ability to control, protect, and track sensitive data at a level comparable to tracking financial data by banks and financial institutions. Data owners and applications teams need to build a rich understanding of what data applications store and what controls are required.

- **Between organizations and cloud providers:** As organizations host workloads in the cloud, they are entering into a business relationship with each of those cloud providers. The use of cloud services often brings business value such as:

    - **Accelerating digital transformation initiatives** by reducing time to market for new capabilities.

    - **Increasing value of IT and security activities** by freeing teams to focus on higher value (business-aligned) activities rather than lower-level commodity tasks that are provided more efficiently by cloud services on their behalf.

    - **Increased reliability and responsiveness:** Most modern clouds also have high uptime compared to traditional on-premises datacenters and have shown they can scale rapidly (such as during the COVID-19 pandemic) and provide resiliency following natural events like lightning strikes (which would have kept many on-premises equivalents down for much longer).

      While beneficial, this shift to the cloud is not without risk. As organizations adopt cloud services, they should consider potential risk areas including:

    - **Business continuity and disaster recovery:** Is the cloud provider financially healthy with a business model that's likely to survive and thrive during your organization's use of the service? Has the cloud provider made provisions to allow customer continuity if the provider experiences financial or other failure, such as providing their source code to customers or open-sourcing it?

      For more information and documents regarding Microsoft's financial health, see Microsoft investor relations.

    - **Security:** Does the cloud provider follow industry best practices for security? Has this been validated by independent regulatory bodies?

        - Microsoft Cloud App Security allows you to discover usage of over 16,000 cloud applications, which are ranked and scored based on more than 70 risk factors to provide you with ongoing visibility into cloud use, shadow IT, and the risk that shadow IT poses to your organization.
        - The Microsoft Service Trust Portal makes regulatory compliance certifications, audit reports, pen tests, and more available to customers. These documents include many details of internal security practices (notably the SOC 2 Type 2 report and FedRAMP Moderate system security plan). Azure Security Center allows the management of security policies and can indicate the level of compliance with predefined industry and regulatory standards.

    - **Business competitor:** Is the cloud provider a significant business competitor in your industry? Do you have sufficient protections in the cloud services contract or other means to protect your

business against potentially hostile actions?

Review this article for commentary on how Microsoft avoids competing with cloud customers.

- **Multicloud:** Many organizations have a de facto or intentional multicloud strategy. This could be an intentional objective to reduce reliance on a single supplier or to access unique best of breed capabilities, but can also happen because developers chose preferred or familiar cloud services, or your organization acquired another business. Regardless of the reason, this strategy can introduce potential risks and costs that have to be managed including:
  - **Downtime from multiple dependencies:** Systems architected to rely on multiple clouds are exposed to more sources of downtime risk as disruptions in the cloud providers (or your team's use of them) could cause an outage/disruption of your business. This increased system complexity would also increase the likelihood of disruption events as team members are less likely to fully understand a more complex system.
  - **Negotiating power:** Larger organizations also should consider whether a single-cloud (mutual commitment/partnership) or multicloud strategy (ability to shift business) will achieve greater influence over their cloud providers to get their organization's feature requests prioritized.
  - **Increased maintenance overhead:** IT and security resources already are overburdened from their existing workloads and keeping up with the changes of a single cloud platform. Each additional platform further increases this overhead and takes team members away from higher value activities like streamlining technical process to speed business innovation, consulting with business groups on more effective use of technologies, and so on.
  - **Staffing and training:** Organizations often do not consider the staffing requirements necessary to support multiple platforms and the training required to maintain knowledge and currency of new features which are released in a rapid pace.

Cloud security functions

6/8/2021 • 4 minutes to read • Edit Online

This article provides a summary of the organizational functions required to manage information security risk in an enterprise. These roles and responsibilities form the human portion of an overall cybersecurity system.

Security is a team sport

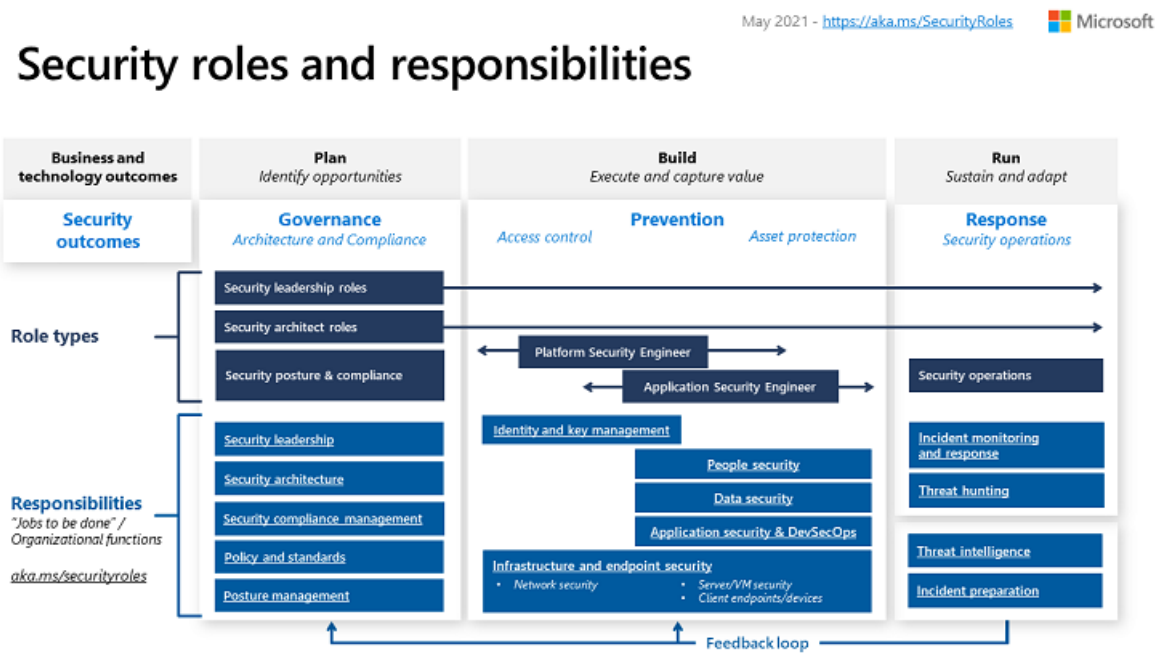
It's critical that individuals on the security team see each other as part of a whole security team, part of the whole organization, and part of a larger security community defending against the same adversaries.

This holistic worldview enables the team to work well in general. It's especially important as the teams work through any unplanned gaps and overlaps discovered during the evolution of roles and responsibilities.

Security responsibilities (functions)

This diagram depicts the specific organizational functions within security, often called responsibilities or "jobs to be done".

These diagrams and documentation represent an ideal view of a complete enterprise security team. This may be an aspirational view for security teams with limited resources that may not have formal responsibilities defined around all of these functions. Each function may be performed by one or more people, and each person may perform one or more functions depending on various factors such as culture, budget, and available resources.



Each of the following articles provides information about each function including a summary of objectives, how the function can evolve because of the threat environment or cloud technology changes, and the relationships and dependencies that are critical to its success.

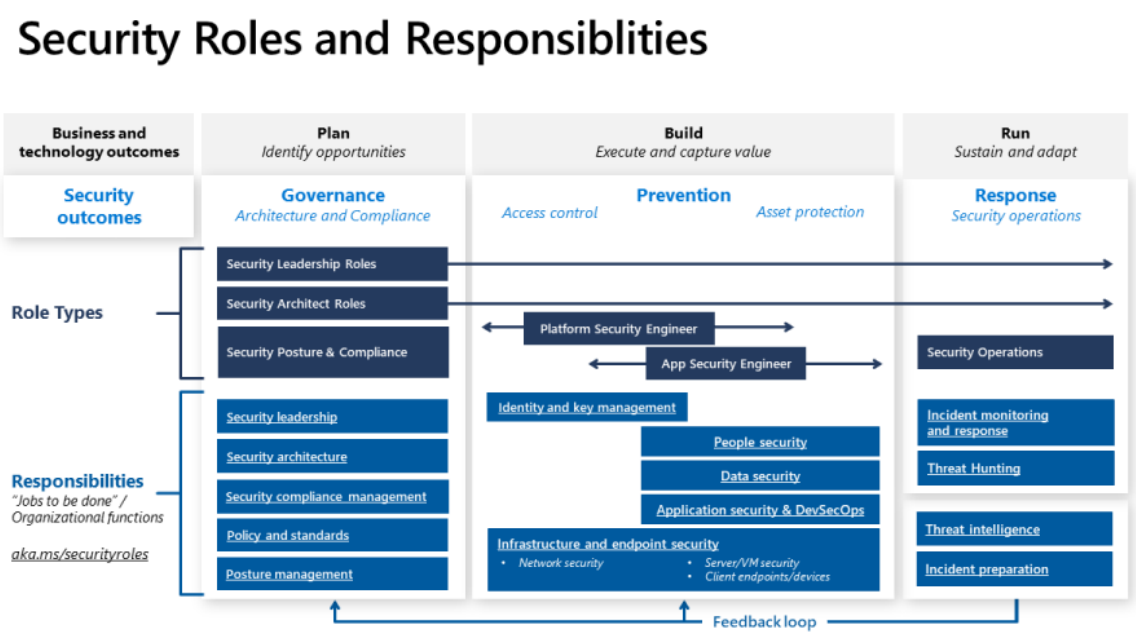
- Policy and standards
- Security operations
- Security architecture
- Security compliance management

- People security
- Application security and DevSecOps
- Data security
- Infrastructure and endpoint security
- Identity and key management
- Threat intelligence
- Posture management
- Incident preparation

These responsibilities are referenced throughout Microsoft documentation including the Azure Security Benchmark, securing privileged access: rapid modernization plan, and the Azure security top 10,

Roles and responsibilities

The following diagram depicts how these functions map to role types within an organization:



Mapping security to business outcomes

At the organizational level, the security disciplines map to standard plan-build-run phases seen widely across industries and organizations. While this cycle is accelerating into a continuous change cycle with the digital age and the advent of DevOps, this illustrates how security maps to normal business processes.

Security is both discipline with its own unique functions *and* a critical element to integrate into normal business operations.

Role types

The middle (dark blue) section groups these responsibilities into typical roles that have common skill sets and career profiles. These groupings also help provide clarity on how industry trends are affecting security professionals:

- **Security leadership:** These roles frequently span across functions, ensuring that teams coordinate with each other, providing prioritization and setting cultural norms, policies, and standards for security.
- **Security architect:** These roles span across functions and provide a key governance capability to ensure all of the technical functions work harmoniously within a consistent architecture

- **Security posture and compliance:** This is a newer role type that represents the increasing convergence of compliance reporting with traditional security disciplines like vulnerability management and configuration baselines. While the scope and audience are different for security and compliance reporting, they are answering different versions of the question of "how secure is the organization?". The way that question is answered is growing more similar via tools like Microsoft Secure Score and Azure Security Center:
  - The use of on-demand data feeds from cloud services is reducing the time required to report compliance.
  - The increased scope of data available is enabling security governance to look beyond traditional software updates/patches and discover/track "vulnerabilities" from security configurations and operational practices
- **Platform security engineer:** These are technology roles focused on platforms that host multiple workloads, focused on both access control and asset protection. These roles are often grouped into teams with specialized technical skill sets including network security, infrastructure and endpoints, identity and key management, and others. These teams work on both preventive controls and detective controls, with detective controls being a partnership with SecOps and preventive controls being primarily a partnership with IT operations. For more information, see Security integration.
- **Application security engineer:** These technology roles focused on security controls for specific workloads, supporting both classic development models and modern DevOps/DevSecOps model. This is a blend of application/development security skills for unique code and infrastructure skills for common technical components like VMs, databases, and containers. These roles may be located in central IT or security organizations or within business and development teams, depending on organizational factors.

> **NOTE**
>
> As both DevOps and infrastructure as code trends progress, we expect to see some security talent migrate from platform security engineering teams to application security teams and posture management roles. This is because the DevOps model requires infrastructure security skills (such as securing the *ops* in DevOps) and governance teams will also require these skills and experience to effectively monitor technical security posture in real time. Additionally, infrastructure as code will automate repetitive manual technical tasks, reducing the volume of time required for these skills in the platform security engineer roles (though increasing the need for broad technical skill sets and automation or scripting skills).

Next steps

Review the Secure methodology

Get started: Implement security across the enterprise environment

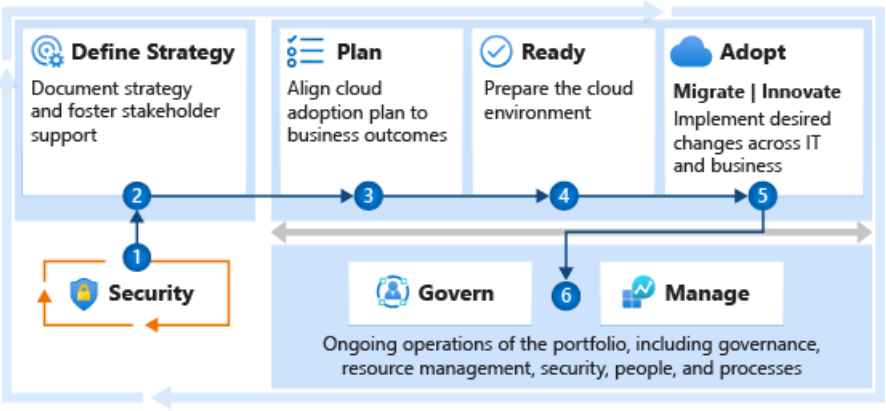
7/21/2021 • 13 minutes to read • Edit Online

Security helps create assurances of confidentiality, integrity, and availability for a business. Security efforts have a critical focus on protecting against the potential impact to operations caused by both internal and external malicious and unintentional acts.

This getting started guide outlines the key steps that will mitigate or avoid the business risk from cybersecurity attacks. It can help you rapidly establish essential security practices in the cloud and integrate security into your cloud adoption process.

The steps in this guide are intended for all roles that support security assurances for cloud environments and landing zones. Tasks include immediate risk mitigation priorities, guidance on building a modern security strategy, operationalizing the approach, and executing on that strategy.

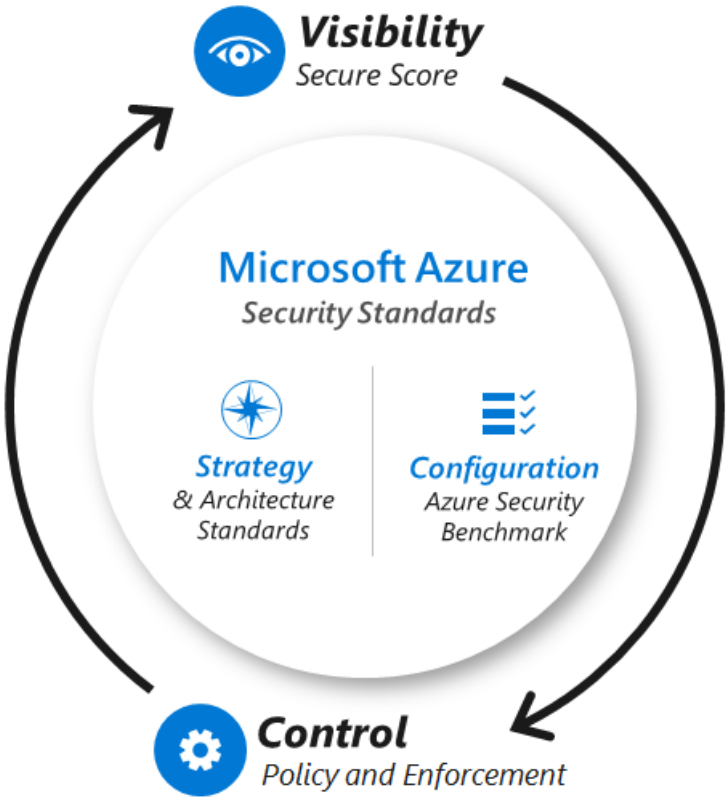
This guide includes elements from across the Microsoft Cloud Adoption Framework for Azure:



Adhering to the steps in this guide will help you integrate security at critical points in the process. The goal is to avoid obstacles in cloud adoption and reduce unnecessary business or operational disruption.

Microsoft has built capabilities and resources to help accelerate your implementation of this security guidance on Microsoft Azure. You'll see these resources referenced throughout this guide. They're designed to help you establish, monitor, and enforce security, and they're frequently updated and reviewed.

The following diagram shows a holistic approach for using security guidance and platform tooling to establish security visibility and control over your cloud assets in Azure. We recommend this approach.



Use these steps to plan and execute your strategy for securing your cloud assets and using the cloud to modernize security operations.

Step 1: Establish essential security practices

Security in the cloud starts with applying the most important security practices to the people, process, and technology elements of your system. Additionally, some architectural decisions are foundational and are very difficult to change later so should be carefully applied.

Whether you're already operating in the cloud or you're planning for future adoption, we recommend that you follow these 11 essential security practices (in addition to meeting any explicit regulatory compliance requirements).

People:

1. Educate teams about the cloud security journey
2. Educate teams on cloud security technology

Process:

3. Assign accountability for cloud security decisions
4. Update incident response processes for cloud
5. Establish security posture management

Technology:

6. Require passwordless or multifactor authentication
7. Integrate native firewall and network security
8. Integrate native threat detection

Foundational architecture decisions:

9. Standardize on a single directory and identity
10. Use identity-based access control (instead of keys)
11. Establish a single unified security strategy

> **NOTE**
> Each organization should define its own minimum standards. Risk posture and subsequent tolerance to that risk can vary widely based on industry, culture, and other factors. For example, a bank might not tolerate any potential damage to its reputation from even a minor attack on a test system. Some organizations would gladly accept that same risk if it accelerated their digital transformation by three to six months.

Step 2: Modernize the security strategy

Effective security in the cloud requires a strategy that reflects the current threat environment and the nature of the cloud platform that's hosting the enterprise assets. A clear strategy improves the effort of all teams to provide a secure and sustainable enterprise cloud environment. The security strategy must enable defined business outcomes, reduce risk to an acceptable level, and enable employees to be productive.

A cloud security strategy provides guidance to all teams working on the technology, processes, and people readiness for this adoption. The strategy should inform the cloud architecture and technical capabilities, guide the security architecture and capabilities, and influence the training and education of teams.

Deliverables:

The strategy step should result in a document that can easily be communicated to many stakeholders within the organization. The stakeholders can potentially include executives on the organization's leadership team.

We recommended capturing the strategy in a presentation to facilitate easy discussion and updating. This presentation can be supported with a document, depending on the culture and preferences.

- **Strategy presentation:** You might have a single strategy presentation, or you might choose to also create summary versions for leadership audiences.
  - **Full presentation:** This should include the full set of elements for the security strategy in the main presentation or in optional reference slides.
  - **Executive summaries:** Versions to use with senior executives and board members might contain only critical elements relevant to their role, such as risk appetite, top priorities, or accepted risks.
- You can also record motivations, outcomes, and business justifications in the strategy and plan template.

Best practices for building security strategy:

Successful programs incorporate these elements into their security strategy process:

- **Align closely to business strategy:** Security's charter is to protect business value. It's critical to align all security efforts to that purpose and minimize internal conflict.
  - **Build a shared understanding** of business, IT, and security requirements.
  - **Integrate security early into cloud adoption** to avoid last-minute crises from avoidable risks.
  - **Use an agile approach** to immediately establish minimum security requirements and continuously improve security assurances over time.
  - **Encourage security culture change** through intentional proactive leadership actions.

  For more information, see Transformations, mindsets, and expectations.
- **Modernize security strategy:** The security strategy should include considerations for all aspects of modern technology environment, current threat landscape, and security community resources.

- Adapt to the shared responsibility model of the cloud.
- Include all cloud types and multicloud deployments.
- Prefer native cloud controls to avoid unnecessary and harmful friction.
- Integrate the security community to keep up with the pace of attacker evolution.

Related resources for additional context:

- Evolution of threat environment, roles, and digital strategies
- Transformation of security, strategies, tools, and threats
- Strategy considerations for the Cloud Adoption Framework:
  - Modernize your security strategy
  - Cybersecurity resilience
  - How cloud is changing security relationships and responsibilities

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Security leadership team (chief information security officer (CISO) or equivalent)	• Cloud strategy team • Cloud security team • Cloud adoption team • Cloud center of excellence or central IT team

Strategy approval:

Executives and business leaders with accountability for outcomes or risks of business lines within the organization should approve this strategy. This group might include the board of directors, depending on the organization.

Step 3: Develop a security plan

Planning puts the security strategy into action by defining outcomes, milestones, timelines, and task owners. This plan also outlines the roles and responsibilities of the teams.

Security planning and cloud adoption planning should not be done in isolation. It's critical to invite the cloud security team into the planning cycles early, to avoid work stoppage or increased risk from security issues being discovered too late. Security planning works best with in-depth knowledge and awareness of the digital estate and existing IT portfolio that comes from being fully integrated into the cloud planning process.

Deliverables:

- **Security plan:** A security plan should be part of the main planning documentation for the cloud. It might be a document that uses the strategy and plan template, a detailed slide deck, or a project file. Or it might be a combination of these formats, depending on the organization's size, culture, and standard practices.

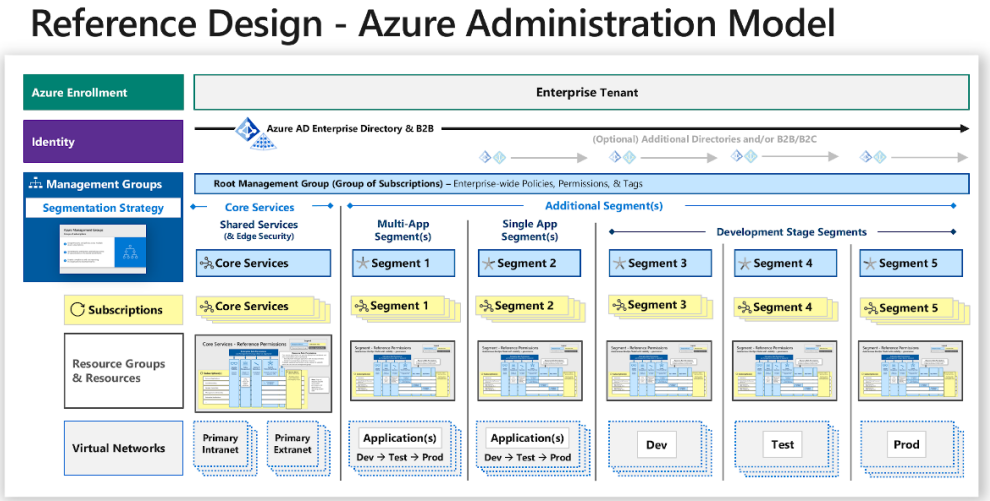
  The security plan should include all of these elements:

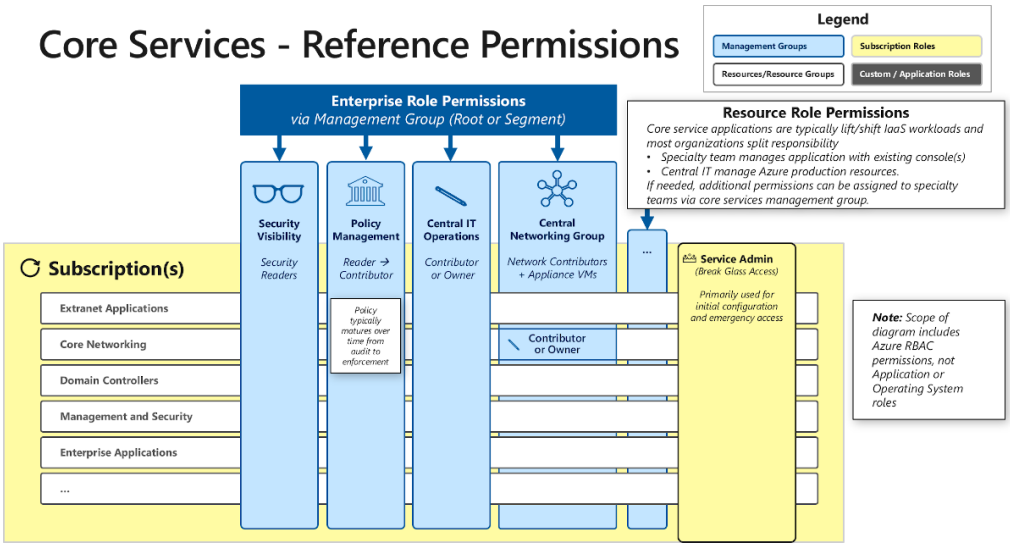
  - **Organizational functions plan**, so teams know how current security roles and responsibilities will change with the move to the cloud.
  - **Security skills plan** to support team members as they navigate the significant changes in technology, roles, and responsibilities.
  - **Technical security architecture and capabilities roadmap** to guide technical teams.

    Microsoft provides reference architectures and technology capabilities to help you as you build

your architecture and roadmap, including:

- Azure components and reference model to accelerate planning and design of Azure security roles.





- Microsoft cybersecurity reference architecture to build a cybersecurity architecture for a hybrid enterprise that spans on-premises and cloud resources.
- Security operations center (SOC) reference architecture to modernize security detection, response, and recovery.
- Zero-trust user access reference architecture to modernize access control architecture for cloud generation.
- Azure Security Center and Microsoft cloud application security to help secure cloud assets.

- **Security awareness and education plan**, so all teams have basic critical security knowledge.
- **Asset sensitivity marking** to designate sensitive assets by using a taxonomy aligned to business impact. The taxonomy is built jointly by business stakeholders, security teams, and other interested parties.
- **Security changes to the cloud plan:** Update other sections of the cloud adoption plan to reflect changes triggered by the security plan.

Best practices for security planning:

Your security plan is likely to be more successful if your planning takes the approach of:

- **Assume a hybrid environment:** That includes software as a service (SaaS) applications and on-premises environments. It also includes multiple cloud infrastructure as a service (IaaS) and platform as a service (PaaS) providers, if applicable.
- **Adopt agile security:** Establish minimum security requirements first and move all noncritical items to a prioritized list of next steps. This should not be a traditional, detailed plan of 3-5 years. The cloud and threat environment change too fast to make that type of plan useful. Your plan should focus on developing the beginning steps and end state:
  - **Quick wins** for the immediate future that will deliver a high impact before longer-term initiatives begin. The time frame can be 3-12 months, depending on organizational culture, standard practices, and other factors.
  - **Clear vision** of the desired end state to guide each team's planning process (which might take multiple years to achieve).
- **Share the plan broadly:** Increase awareness of, feedback from, and buy-in by stakeholders.
- **Meet the strategic outcomes:** Ensure that your plan aligns to and accomplishes the strategic outcomes described in the security strategy.
- **Set ownership, accountability, and deadlines:** Ensure that the owners for each task are identified and are committed to completing that task in a specific time frame.
- **Connect with the human side of security:** Engage people during this period of transformation and new expectations by:
  - **Actively supporting team member transformation** with clear communication and coaching on:
    - What skills they need to learn.
    - Why they need to learn the skills (and the benefits of doing so).
    - How to get this knowledge (and provide resources to help them learn).

    You can document the plan by using the strategy and plan template. And you can use online Microsoft security training to help with education of your team members.
  - **Making security awareness engaging** to help people genuinely connect with their part of keeping the organization safe.
- **Review Microsoft learnings and guidance:** Microsoft has published insights and perspectives to help your organization plan its transformation to the cloud and a modern security strategy. The material includes recorded training, documentation, and security best practices and recommended standards.

  For technical guidance to help build your plan and architecture, see the Microsoft security documentation.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Cloud security team	• Cloud strategy team • Cloud governance team • Any risk teams in your organization • Cloud center of excellence or central IT team

Security plan approval:

The security leadership team (CISO or equivalent) should approve the plan.

Step 4: Secure new workloads

It's a lot easier to start in a secure state than to retrofit security later into your environment. We strongly recommend starting with a secure configuration to ensure that workloads are migrated to, and developed and tested in, a secure environment.

During landing zone implementation, many decisions can affect security and risk profiles. The cloud security team should review the landing zone configuration to ensure that it meets the security standards and requirements in your organization's security baselines.

Deliverables:

- Ensure that new landing zones meet the organization's compliance and security requirements.

Guidance to support deliverable completion:

- **Blend existing requirements and cloud recommendations:** Start with recommended guidance and then adapt this to your unique security requirements. We have seen challenges with trying to enforce existing on-premises policies and standards, because these often refer to outdated technology or security approaches.

  Microsoft has published guidance to help you build your security baselines:

  - Azure security standards for strategy and architecture: Strategy and architectural recommendations to shape your environment's security posture.
  - Azure security benchmarks: Specific configuration recommendations for securing Azure environments.
  - Azure security baseline training.
- **Provide guardrails:** Safeguards should include automated policy auditing and enforcement. For these new environments, teams should strive to both audit and enforce the organization's security baselines. These efforts can help minimize security surprises during the development of workloads, as well as continuous integration and continuous deployment (CI/CD) of workloads.

  Microsoft provides several native capabilities in Azure to enable this:

  - Secure score: Use a scored assessment of your Azure security posture to track security efforts and projects in your organization.
  - Azure Blueprints: Cloud architects and centralized IT groups can define a repeatable set of Azure resources that implements and adheres to an organization's standards, patterns, and requirements.
  - Azure Policy: This is the foundation of the visibility and control capabilities that the other services use. Azure Policy is integrated into Azure Resource Manager, so you can audit changes and enforce policies across any resource in Azure before, during, or after its creation.
  - Improve landing zone operations: Use best practices for improving security within a landing zone.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Cloud security team	• Cloud adoption team • Cloud platform team • Cloud strategy team • Cloud governance team • Cloud center of excellence or central IT team

Step 5: Secure existing cloud workloads

Many organizations have already deployed assets to enterprise cloud environments without applying the security best practices, creating increased business risk.

After you ensure that new applications and landing zones follow security best practices, you should focus on bringing existing environments up to the same standards.

Deliverables:

- Ensure that all existing cloud environments and landing zones meet the organization's compliance and security requirements.
- Test operational readiness of production deployments by using policies for security baselines.
- Validate adherence to design guidance and security requirements for security baselines.

Guidance to support deliverable completion:

- Use the same security baselines that you built in Step 4 as your ideal state. You might have to adjust some policy settings to only audit instead of enforcing them.
- Balance operational and security risk. Because these environments might host production systems that enable critical business processes, you might need to implement security improvements incrementally to avoid risking operational downtime.
- Prioritize the discovery and remediation of security risk by business criticality. Start with workloads that have a high business impact if compromised and workloads that have a high exposure to risk.

For more information, see Identify and classify business-critical applications.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Cloud adoption team	• Cloud adoption team • Cloud strategy team • Cloud security team • Cloud governance team • Cloud center of excellence or central IT team

Step 6: Govern to manage and improve security posture

Like all modern disciplines, security is an iterative process that should focus on continuous improvement. Security posture can also decay if organizations don't sustain focus on it over time.

Consistent application of security requirements comes from sound governance disciplines and automated solutions. After the cloud security team defines the security baselines, those requirements should be audited to ensure they're applied consistently to all cloud environments (and enforced where applicable).

Deliverables:

- Ensure that the organization's security baselines are applied to all relevant systems. Audit anomalies by using a secure score or a similar mechanism.
- Document your Security Baseline policies, processes, and design guidance in the Security Baseline discipline template.

Guidance to support deliverable completion:

- Use the same security baselines and auditing mechanisms that you built in Step 4 as technical components of monitoring the baselines. Complement these baselines with people and process controls to ensure consistency.
- Ensure that all workloads and resources follow proper naming and tagging conventions. Enforce tagging conventions by using Azure Policy, with a specific emphasis on tags for data sensitivity.
- If you're new to cloud governance, establish governance policies, processes, and disciplines by using the Govern methodology.

ACCOUNTABLE TEAM	RESPONSIBLE AND SUPPORTING TEAMS
• Cloud governance team	• Cloud strategy team • Cloud security team • Cloud center of excellence or central IT team

Next steps

The steps in this guide have helped you implement the strategy, controls, processes, skills, and culture needed to consistently manage security risks across the enterprise.

As you continue into the operations mode of cloud security, consider these next steps:

- Review Microsoft security documentation. It provides technical guidance to help security professionals build and improve cybersecurity strategy, architecture, and prioritized roadmaps.
- Review security information in Built-in security controls for Azure services.
- Review Azure security tools and services in Security services and technologies available on Azure.
- Review the Microsoft Trust Center. It contains extensive guidance, reports, and related documentation that can help you perform risk assessments as part of your regulatory compliance processes.
- Review third-party tools available to facilitate meeting your security requirements. For more information, see Integrate security solutions in Azure Security Center.

Azure security best practices

8/2/2021 • 26 minutes to read • Edit Online

These are the top Azure security best practices that Microsoft recommends based on lessons learned across customers and our own environments.

For a video presentation of these best practices, see Top 10 best practices for Azure security.

1. People: Educate teams about the cloud security journey

The team needs to understand the journey they're on.

What: educate your security and IT teams on the cloud security journey and the changes they will be navigating including:

- Changes to threats in the cloud
- Shared responsibility model and how it impacts security
- Cultural and role/responsibility changes that typically accompany cloud adoption

Why: moving to the cloud is a significant change that requires a shift in mindset and approach for security. While the outcomes security provides to the organization won't change, the best way to accomplish this in the cloud often changes, sometimes significantly.

In many ways, moving to the cloud is similar to moving from a standalone house into a high-rise luxury apartment building. You still have basic infrastructure (like plumbing and electricity) and perform similar activities (such as socializing, cooking, TV and internet, and so on). However, there is often quite a difference in what comes with the building (like restaurants or a gym), who provides and maintains them, and your daily routine.

Who: everyone in the security and IT organization with any security responsibilities should be familiar with this context and the changes (from CIO/CISO to technical practitioners).

How: provide teams with the context required to successfully deploy and operate during the transition to the cloud environment.

Microsoft has published lessons learned by our customers and our own IT organization on their journeys to the cloud:

- How security roles and responsibilities are evolving in the security organization
- Evolution of threat environment, roles, and digital strategies
- Transformation of security, strategies, tools, and threats
- Learnings from Microsoft experience securing hyperscale cloud environment that can help you on your journey

Also see the Azure Security Benchmark gs-3: align organization roles, responsibilities, and accountabilities.

2. People: Educate teams on cloud security technology

People need to understand where they're going.

What: ensure your teams have time set aside for technical education on securing cloud resources including:

- Cloud technology and cloud security technology

- Recommended configurations and best practices
- Where to learn more technical details as needed

Why: technical teams need access to technical information to make sound informed security decisions. Technical teams are good at learning new technologies on the job, but the volume of details in the cloud often overwhelms their ability to fit learning into their daily routine.

Structuring dedicated time for technical learning helps ensure people have time to build confidence on their ability to assess cloud security and think through how to adapt their existing skills and processes. Even the most talented special operations teams in the military need training and intelligence to perform at their best.

Who: All roles that directly interact with cloud technology (in security and IT departments) should dedicate time for technical learning on cloud platforms and how to secure them.

Additionally security and IT technical managers (and often project managers) should develop familiarity with some technical details for securing cloud resources (as this will help them more effectively lead and coordinate cloud initiatives).

How: Ensure that technical professionals in security have time set aside for self-paced training on how to secure cloud assets. While not always feasible, ideally provide access to formal training with an experienced instructor and hands-on labs.

> **IMPORTANT**
> Identity protocols are critical to access control in the cloud but often not prioritized in on-premises security, so security teams should ensure to focus on developing familiarity with these protocols and logs.

Microsoft provides extensive resources to help technical professionals ramp up on securing Azure resources and report compliance:

- Azure security
  - Az-500 learning path (and certification)
  - Azure Security Benchmark (ASB) -prescriptive best practices and controls for Azure security
    - Security baselines for Azure: Application of ASB to individual Azure services
  - Microsoft security best practices - videos and documentation
- Azure compliance
  - Regulatory compliance evaluation with Azure Security Center
- Identity protocols and security
  - Azure security documentation site
  - Azure AD authentication YouTube series
  - Securing Azure environments with Azure Active Directory

Also see the Azure Security Benchmark gs-3: align organization roles, responsibilities, and accountabilities

3. Process: Assign accountability for cloud security decisions

If nobody is accountable for making security decisions, they won't get made.

What: designate who is responsible for making each type of security decision for the enterprise Azure environment.

Why: clear ownership of security decisions speeds up cloud adoption **and** increases security. Lack of typically creates friction because nobody feels empowered to make decisions, nobody knows who to ask for a decision, and nobody is incentivized to research a well-informed decision. This friction frequently impedes business goals, developer timelines, IT goals, and security assurances, resulting in:

- Stalled projects that are waiting for security approval
- Insecure deployments that couldn't wait for security approval

Who: security leadership designates which teams or individuals are accountable for making security decisions about the cloud.

How: designate groups (or individuals) that will be responsible for making key security decisions.

Document these owners, their contact information, and socialize this widely within the security, IT, and cloud teams to ensure it's easy for all roles to contact them.

These are the typical areas where security decisions are needed, descriptions, and which teams typically make the decisions.

DECISION	DESCRIPTION	TYPICAL TEAM
Network security	Configuration and maintenance of Azure Firewall, network virtual appliances (and associated routing), WAFs, NSGs, ASGs, and so on.	*Typically Infrastructure and endpoint security team focused on network security*
Network management	Enterprise-wide virtual network and subnet allocation	*Typically existing network operations team in Central IT Operations*
Server endpoint security	Monitor and remediate server security (patching, configuration, endpoint security, and so on.)	*Typically Central IT Operations and Infrastructure and endpoint security teams jointly*
Incident monitoring and response	Investigate and remediate security incidents in SIEM or source console (Azure Security Center, Azure AD identity protection, and so on.)	*Typically security operations team*
Policy management	Set direction for use of Azure role-based access control (Azure RBAC), Azure Security Center, administrator protection strategy, and Azure Policy to govern Azure resources	*Typically Policy and Standards + Security Architecture Teams jointly*
Identity security and standards	Set direction for Azure AD directories, PIM/pam usage, multifactor authentication, password/synchronization configuration, application identity standards	*Typically Identity and Key Management + Policy and Standards + Security Architecture Teams jointly*

NOTE

- Ensure decision makers have the appropriate education in their area of the cloud to accompany this responsibility.
- Ensure decisions are documented in policy and standards to provide a record and guide the organization over the long term.

Also see the Azure Security Benchmark gs-3: align organization roles, responsibilities, and accountabilities

4. Process: Update incident response processes for cloud

You don't have time to plan for a crisis during a crisis.

What: update processes and prepare analysts to for responding to security incidents on your Azure cloud platform (including any native threat detection tools you have adopted). Update processes, prepare your team, and practice with simulated attacks so they can perform at their best during incident investigation, remediation, and threat hunting.

Why: active attackers present an immediate risk to the organization that can quickly become a difficult to control situation, so you must rapidly effectively respond to attacks. This incident response (IR) process must be effective for your entire estate including all cloud platforms hosting enterprise data, systems, and accounts.

While similar in many ways, cloud platforms have important technical difference from on-premises systems that can break existing processes, typically because information is available in a different form. Security analysts may also have challenges rapidly responding to an unfamiliar environment that can slow them down (especially if they are trained only on classic on-premises architectures and network/disk forensics approaches).

Who: modernizing the IR processes is typically led by security operations with support from other groups for knowledge and expertise.

- **Sponsorship:** This process modernization is typically sponsored by the security operations director or equivalent.
- **Execution:** Adapting existing processes (or writing them for the first time) is a collaborative effort involving:
  - Security operations incident management team or leadership -leads updates to process and integration of key external stakeholders including legal and communications/public relations teams
  - Security operations security analysts - provide expertise on technical incident investigation and triage
  - Central IT operations - provides expertise on cloud platform (directly, via cloud center of excellence, or via external consultants)

How: update processes and prepare your team so they know what to do when they find an active attacker.

- **Processes and playbooks:** Adapt existing investigations, remediation, and threat hunting processes to the differences of how cloud platforms work (new/different tools, data sources, identity protocols, and so on.).
- **Education:** Educate analysts on the overall cloud transformation, technical details of how the platform works, and new/updated processes so that they know what will be different and where to go for what they need.

Key Focus Areas: while there are many details described in the resource links, these are key areas to focus your education and planning efforts:

- **Shared responsibility model and cloud architectures:** To a security analyst, Azure is a software defined datacenter that provides many services including VMs (familiar) and others that are very different from on-premises such as Azure SQL Database Azure Functions, and so on. Where the best data is in the service logs or the specialized threat detection services rather than in logs for the underlying OS/VMs (which are operated by Microsoft and service multiple customers). Analysts need to understand and integrate this context into their daily workflows so they know what data to expect, where to get it, and what format it will be in.
- **Endpoint data sources:** Getting insights and data for attacks and malware on cloud hosted servers is often faster, easier, and more accurate with native cloud detection tools like Azure Security Center and endpoint detection and response (EDR) solutions as opposed to traditional approaches of direct disk access. While direct disk forensics are available for scenarios where it's possible and required for legal proceedings (computer forensics in Azure), this is often the most inefficient way to detect and investigate attacks.
- **Network and identity data sources:** Many functions of cloud platforms primarily use identity primarily

for access control such as access to the Azure portal (though network access controls are used extensively as well). This requires analysts to develop an understanding of cloud identity protocols to get a full, rich, picture of attacker activity (and legitimate user activity) to support incident investigation and remediation. Identity directories and protocols are also different from on-premises as they are typically based on SAML, OAuth, and OpenID Connect and cloud directories rather than LDAP, Kerberos, NTLM, and Active Directory that are commonly found on-premises.

- **Practice exercises:** Simulated attacks and response can help build organizational muscle memory and technical readiness for your security analysts, threat hunters, incident managers, and other stakeholders in your organization. Learning on the job and adapting is a natural part of incident response, but you should work to minimize how much you have to learn in a crisis.

Key Resources:

- Incident response reference guide (IRRG)
- Guidance on building your own security incident response process
- Azure logging and alerting
- Microsoft security best practices
  - Transformation of security, strategies, tools, & threats
  - Security operations
- Microsoft learnings from Cyber Defense Operations Center (CDOC)
  - Overall lessons learned
  - Incident investigation
  - Incident remediation

Also see the Azure Security Benchmark IR-1: preparation - update incident response process for Azure.

5. Process: Establish security posture management

First, know thyself.

What: ensure that you are actively managing the security posture of your Azure environment by:

- Assigning clear ownership of responsibilities for
  - Monitoring security posture
  - Mitigating risks to assets
- Automating and simplifying these tasks

Why: rapidly identifying and remediating common security hygiene risks significantly reduces organizational risk.

The software-defined nature of cloud datacenters enables continuous monitoring of security risks (such as software vulnerabilities or security misconfigurations) with extensive asset instrumentation. The speed at which developers and IT team can deploy VMs, databases, and other resources also create a need to ensure resources are configured securely and actively monitored.

These new capabilities offer new possibilities, but realizing value from them requires assigning responsibility for using them. Executing consistently on with rapidly evolving cloud operations also requires keeping human processes as simple and automated as possible. See the "drive simplicity" security principle.

> **NOTE**
> The goal of simplification and automation isn't about getting rid of jobs, but about removing the burden of repetitive tasks from people so they can focus on higher value human activities like engaging with and educating IT and DevOps teams.

Who: this is typically divided into two sets of responsibilities:

- Security posture management - this newer function is often an evolution of existing vulnerability management or governance functions. This includes monitoring overall security posture using Azure Security Center secure score and other data sources, actively working with resource owners to mitigate risks, and reporting risk to security leadership.
- **Security remediation:** Assign accountability for addressing these risks to the teams responsible for managing those resources. This should either the DevOps teams managing their own application resources or the technology-specific teams in **Central IT Operations**:
  - **Compute and application resources:**
    - **App services:** Application development/security teams
    - **Containers:** Application development or infrastructure/IT operations
    - **VMs/scale sets/compute:** IT/infrastructure operations
  - **Data and storage resources:**
    - **SQL/Redis/Data Lake Analytics/data lake store:** Database team
    - **Storage accounts:** Storage/infrastructure team
  - **Identity and access resources:**
    - **Subscriptions:** Identity teams
    - **Key Vault:** Identity or information/data security team
  - **Networking resources:** Network security team
  - **IoT security:** IoT operations team

How: security is everyone's job, but not everyone currently knows how important it is, what to do, and how to do it.

- Hold resource owners accountable for the security risk just as they are held accountable for availability, performance, cost, and other success factors.
- Support resource owners with a clear understanding of why security risk matters to their assets, what they should do to mitigate risk, and how to implement it with minimal productivity loss.

> **IMPORTANT**
> The explanations for why, what, and how to secure resources are often similar across different resource types and applications, but it's critical to relate these to what each team already knows and cares about. Security teams should engage with their IT and DevOps counterparts as a trusted advisor and partner focused on enabling these teams to be successful.

Tooling: secure score in Azure Security Center provides an assessment of the most important security information in Azure for a wide variety of assets. This should be your starting point on posture management and can be supplemented with custom Azure policies and other mechanisms as needed.

Frequency: set up a regular cadence (typically monthly) to review Azure secure score and plan initiatives with specific improvement goals. The frequency can be increased as needed.

> **TIP**
> Gamify the activity if possible to increase engagement, such as creating fun competitions and prizes for the DevOps teams that improve their score the most.

Also see the Azure Security Benchmark gs-2: define security posture management strategy.

6. Technology: Require passwordless or multifactor authentication

Are you willing to bet the security of your enterprise that professional attackers can't guess or steal your administrator's password?

What: require all critical impact admins to use passwordless or multifactor authentication.

Why: just as antique skeleton keys won't protect a house against a modern day burglar, passwords cannot protect accounts against common attacks we see today. Technical details are described in Your pa$$word doesn't matter.

While multifactor authentication was once a burdensome extra step, passwordless approaches today improve the logon experience using biometric approaches like facial recognition in Windows Hello and mobile devices (where you don't have to remember or type a password). Additionally, zero trust approaches remember trusted devices, which reduce prompting for annoying out of band multifactor authentication actions (see User sign-in frequency).

Who: password and multifactor initiative is typically led by identity and key management or security architecture.

- **Sponsorship:** This is typically sponsored by CISO, CIO, or director of identity
- **Execution:** This is a collaborative effort involving
  - Policy and standards team establish clear requirements
  - Identity and key management or central IT operations to implement the policy
  - Security compliance management monitors to ensure compliance

How: implement passwordless or multifactor authentication, train administrators on how to use it (as needed), and require admins to follow using written policy. This can be accomplished by one or more of these technologies:

- Passwordless (Windows Hello)
- Passwordless (authenticator app)
- Azure multifactor authentication
- Third-party multifactor authentication solution

> **NOTE**
> Text message-based multifactor authentication is now relatively inexpensive for attackers to bypass, so focus on passwordless and stronger multifactor authentication.

Also see the Azure Security Benchmark ID-4: use strong authentication controls for all Azure Active Directory-based access.

7. Technology: Integrate native firewall and network security

Simplify protection of systems and data against network attacks.

What: simplify your network security strategy and maintenance by integrating Azure Firewall, Azure web app firewall (WAF), and distributed denial of service (DDoS) mitigations into your network security approach.

Why: simplicity is critical to security as it reduces likelihood of risk from confusion, misconfigurations, and other human errors. See the "drive simplicity" security principle.

Firewalls and WAFs are important basic security controls to protect applications from malicious traffic, but their setup and maintenance can be complex and consume a significant amount of the security team's time and attention (similar to adding custom aftermarket parts to a car). Azure's native capabilities can simplify implementation and operation of firewalls, web application firewalls, distributed denial of service (DDoS) mitigations, and more.

This can free up your team's time and attention for higher value security tasks like evaluating the security of Azure services, automating security operations, and integrating security with applications and IT solutions.

Who:

- **Sponsorship:** This update of network security strategy is typically sponsored by security leadership or IT leadership
- **Execution:** Integrating these into your cloud network security strategy is a collaborative effort involving:
  - Security architecture - establish cloud network security architecture with cloud network and cloud network security leads.
  - **Cloud network leads** (central IT operations) + **Cloud Network security leads** (infrastructure security team)
    - Establish cloud network security architecture with security architects
    - Configure firewall, NSG, and WAF capabilities and work with application architects on WAF rules
  - **Application architects:** Work with network security to build and refine WAF rule sets and DDoS configurations to protect the application without disrupting availability

How: organizations looking to simplify their operations have two options:

- **Extend existing capabilities and architectures:** Many organizations often choose to extend the use of existing firewall capabilities so they can capitalize on existing investments into skills and process integration, particularly as they first adopt the cloud.
- **Embrace native security controls:** More and more organizations are starting to prefer using native controls to avoid the complexity of integrating third-party capabilities. These organizations are typically seeking to avoid the risk of a misconfiguration in load balancing, user-defined routes, the firewall/WAF itself, and delays in handoffs between different technical teams. This option is particularly compelling to organizations embracing infrastructure as code approaches as they can automate and instrument the built-in capabilities more easily than third-party capabilities.

Documentation on Azure native network security capabilities can be found at:

- Azure Firewall
- Azure Web Application Firewall (WAF)
- Azure DDoS protection

Azure Marketplace includes many third-party firewall providers.

Also see the Azure Security Benchmark ns-4: protect applications and services from external network attacks.

8. Technology: Integrate native threat detection

Simplify detection and response of attacks against Azure systems and data.

What: simplify your threat detection and response strategy by incorporating native threat detection capabilities into your security operations and SIEM.

Why: the purpose of security operations is to reduce the impact of active attackers who get access to the environment, as measured by mean time to acknowledge (MTTA) and remediate (MTTR) incidents. This requires both accuracy and speed in all elements of incident response, so the quality of tools and the efficiency of process execution are paramount.

It's difficult to get high threat detections using existing tools and approaches designed for on-premises threat detection because of differences in cloud technology and its rapid pace of change. Natively integrated detections provide industrial scale solutions maintained by the cloud providers that can keep up with current threats and cloud platform changes.

These native solutions also enable security operations teams to focus on incident investigation and remediation instead of wasting time trying to create alerts from unfamiliar log data, integrating tools, and maintenance tasks.

Who: this is typically driven by the security operations team.

- **Sponsorship:** This is typically sponsored by the security operations director or equivalent role.
- **Execution:** Integrating native threat detection is a collaborative effort involving those with:
  - Security operations: Integrate alerts into SIEM and incident investigation processes, educate analysts on cloud alerts and what they mean, and how to use the native cloud tools.
  - Incident preparation: Integrate cloud incidents into practice exercises and ensure practice exercises are conducted to drive team readiness.
  - Threat intelligence: Research and integrate information on cloud attacks to inform teams with context and intelligence.
  - Security architecture: Integrate native tooling into security architecture documentation.
  - Policy and standards: Set standards and policy for enabling native tooling throughout the organization. Monitor for compliance.
  - Infrastructure and endpoint / **Central IT Operations**: Configure and enable detections, integrate into automation and infrastructure as code solutions.

How: enable threat detection in Azure Security Center for all the resources you are using and have each team integrate these into their processes as described above.

Also see the Azure Security Benchmark LT-1: enable threat detection for Azure resources.

9. Architecture: Standardize on a single directory and identity

Nobody wants to deal with multiple identities and directories.

What: standardize on a single Azure AD directory and single identity for each application and user in Azure (for all enterprise identity functions).

> **NOTE**
> This best practice refers specifically to enterprise resources. For partner accounts, use Azure AD B2B so you don't have to create and maintain accounts in your directory. For customer/citizen accounts, use Azure AD B2C to manage them.

Why: multiple accounts and identity directories create unnecessary friction and confusion in daily workflows for productivity users, developers, IT and identity admins, security analysts, and other roles.

Managing multiple accounts and directories also creates an incentive for poor security practices such as reusing the same password across accounts and increases the likelihood of stale/abandoned accounts that attackers can

target.

While it sometimes seems easier to quickly stand up a custom LDAP directory for a particular application or workload, this creates much more work to integrate and manage. This is similar to choosing to set up an additional Azure tenant or on-premises Active Directory forest rather than using the existing enterprise tenant. See also the security principle of driving simplicity.

Who: this is often a cross-team effort, driven by security architecture or identity and key management teams.

- **Sponsorship:** This is typically sponsored by identity and key management and security architecture (though some organizations may require sponsorship by CISO or CIO)
- **Execution:** This is a collaborative effort involving:
  - Security architecture: Incorporates into security and IT architecture documents and diagrams
  - Policy and standards: Document policy and monitor for compliance
  - Identity and key management or **Central IT Operations** to implement the policy by enabling features and supporting developers with accounts, education, and so on.
  - **Application developers** or **Central IT Operations**: Use identity in applications and Azure service configurations (responsibilities will vary based on level of DevOps adoption)

How: adopt a pragmatic approach that starts with new *greenfield* capabilities (growing today) and then clean up challenges with the *brownfield* of existing applications and services as a follow-up exercise:

- **Greenfield:** Establish and implement a clear policy that all enterprise identity going forward should use a single Azure AD directory with a single account for each user.
- **Brownfield:** Many organizations often have multiple legacy directories and identity systems. Address these when the cost of ongoing management friction exceeds the investment to clean it up. While identity management and synchronization solutions can mitigate some of these issues, they lack deep integration of security and productivity features that enable a seamless experience for users, admins, and developers.

The ideal time to consolidate your use of identity is during application development cycles as you:

- Modernize applications for the cloud.
- Update cloud applications with DevOps processes.

While there are valid reasons for a separate directory in the case of extremely independent business units or regulatory requirements, multiple directories should be avoided in all other circumstances.

Also see the Azure Security Benchmark ID-1: standardize Azure Active Directory as the central identity and authentication system.

> **IMPORTANT**
>
> The only exception to the single accounts rule is that privileged users (including IT administrators and security analysts) should have separate accounts for standard user tasks vs. administrative tasks.
>
> For more information, see Azure Security Benchmark privileged access.

10. Architecture: use identity-based access control (instead of keys)

What: use Azure AD identities instead of key-based authentication wherever possible (Azure services, applications, APIs, and so on.).

Why: key-based authentication can be used to authenticate to cloud services and APIs but requires managing keys securely, which is challenging to perform well (especially at scale). Secure key management is difficult for non-security professionals like developers and infrastructure professionals and they often fail to do it securely,

often creating major security risks for the organization.

Identity-based authentication overcomes many of these challenges with mature capabilities for secret rotation, lifecycle management, administrative delegation, and more.

Who: this is often a cross-team effort, driven by security architecture or identity and key management teams.

- **Sponsorship:** This is typically sponsored by security architecture or identity and key management (though some organizations may require sponsorship by CISO or CIO).
- **Execution:** This is a collaborative effort involving
  - Security architecture: Incorporates into security and IT architecture diagrams and documents.
  - Policy and standards: Document policy and monitor for compliance.
  - Identity and key management or **Central IT Operations** to implement the policy by enabling features and supporting developers with accounts, education, and so on.
  - **App developers** or **Central IT Operations:** Use identity in applications and Azure service configurations (responsibilities will vary based on level of DevOps adoption).

How: setting an organizational preference and habit for using identity-based authentication requires following a process and enabling technology.

The process:

1. **Establish policy** and standards that clearly outline the default identity-based authentication, as well as acceptable exceptions.
2. **Educate** developers and infrastructure teams on why to use the new approach, what they need to do, and how to do it.
3. **Implement** changes in a pragmatic way - starting with new greenfield capabilities being adopted now and in the future (new Azure services, new applications) and then following up with a clean-up of existing brownfield configurations.
4. **Monitor** for compliance and follow up with developer and infrastructure teams to remediate.

The technologies: For non-human accounts such as services or automation, use managed identities. Azure managed identities can authenticate to Azure services and resources that support Azure AD authentication. Authentication is enabled through predefined access grant rules, avoiding hard-coded credentials in source code or configuration files.

For services that do not support managed identities, use Azure AD to create a service principal with restricted permissions at the resource level instead. We recommended configuring service principals with certificate credentials and fall back to client secrets. In both cases, Azure Key Vault can be used in conjunction with Azure managed identities, so that the runtime environment (such as an Azure function) can retrieve the credential from the key vault.

Also see the Azure Security Benchmark ID-2: manage application identities securely and automatically.

11. Architecture: establish a single unified security strategy

Everyone needs to row in the same direction for the boat to go forward.

What: ensure all teams are aligned to a single strategy that both enables and secures enterprise systems and data.

Why: when teams work in isolation without being aligned to a common strategy, their individual actions can inadvertently undermine each other's efforts, creating unnecessary friction that slows down progress against everyone's goals.

One example of this that has played out consistently in many organizations is the segmentation of assets:

- The *network security team* develops a strategy for segmenting a *flat network* to increase security (often based on physical sites, assigned IP address addresses/ranges, or similar)
- Separately, the *identity team* developed a strategy for groups and Active Directory organizational units (OUs) based on their understanding and knowledge of the organization.
- Often, *application teams* find it difficult to work with these systems because they were designed with limited input and understanding of business operations, goals, and risks.

In organizations where this happens, teams frequently experience conflicts over firewall exceptions, which negatively impact both security (exceptions are usually approved) and productivity (deployment is slowed for application functionality the business needs).

While security can create healthy friction by forcing critical thinking, this conflict only creates unhealthy friction that impedes goals. For more information, see Security strategy guidance: the right level of security friction.

Who:

- **Sponsorship:** The unified strategy typically co-sponsored by CIO, CISO, and CTO (often with business leadership support for some high-level elements) and championed by representatives from each team.
- **Execution:** Security strategy must be implemented by everyone, so it should integrate input from across teams to increase ownership, buy-in, and likelihood of success.
  - Security architecture: Leads the effort to build security strategy and resulting architecture, actively gather feedback from teams, and document it in presentations, documents, and diagrams for the various audiences.
  - Policy and standards: Captures the appropriate elements into standards and policy and then monitors for compliance.
  - **All technical IT and security teams:** Provide input requirements, then align to and implement the enterprise strategy.
  - **Application owners and developers:** Read and understand strategy documentation that applies to them (ideally, guidance tailored to their role).

How:

Build and implement a security strategy for cloud that includes the input and active participation of all teams. While the process documentation format will vary, this should always include:

- **Active input from teams:** Strategies typically fail if people in the organization don't buy into them. Ideally, get all teams in the same room to collaboratively build the strategy. In the workshops we conduct with customers, we often find organizations have been operating in de facto silos and these meetings often result in people meeting each other for the first time. We also find that inclusiveness is a requirement. If some teams are not invited, this meeting typically has to be repeated until all participants join it (or the project doesn't move forward).
- **Documented and communicated clearly:** All teams should have awareness of the security strategy (ideally a security component of the overall technology strategy) including why to integrate security, what is important in security, and what security success looks like. This should include specific guidance for application and development teams so they can get a clear prioritized guidance without having to read through non-relevant parts of the guidance.
- **Stable, but flexible:** Strategies should remain relatively consistent and stable, but the architectures and the documentation may need changes to add clarity and accommodate the dynamic nature of cloud. For example, filtering out malicious external traffic would stay consistent as a strategic imperative even if you shift from the use of a third-party next generation firewall to Azure Firewall and adjust diagrams/guidance on how to do it.
- **Start with segmentation:** Over the course of cloud adoption, your teams will address many strategy topics large and small, but you need to start somewhere. We recommend starting the security strategy with enterprise asset segmentation as it's a foundational decision that would be challenging to change later and

requires both business input and many technical teams.

Microsoft has published video guidance for applying a segmentation strategy to Azure as well as documents on enterprise segmentation and aligning network security to it.

The Cloud Adoption Framework includes guidance to help your teams with:

- Building a cloud strategy team: Ideally, security should be integrated into an existing cloud strategy.
- Build or modernize a security strategy: To meet business and security goals in the current age of cloud services and modern threats.

Also see the Azure Security Benchmark governance and strategy.

Manage organizational alignment

3/6/2021 • 2 minutes to read • Edit Online

Cloud adoption can't happen without well-organized people. Successful cloud adoption is the result of properly skilled people doing the appropriate types of work, in alignment with clearly defined business goals, and in a well-managed environment. To deliver an effective operating model for the cloud, it's important to establish appropriately staffed organizational structures. This article outlines an approach to establishing and maintaining the proper organizational structures in four steps.

The following exercises will help guide the process of creating a landing zone to support cloud adoption.

	Structure type: Define the type of organizational structure that best fits your operating model.
	Cloud functions: Understand the cloud functionality required to adopt and operate the cloud.
	Mature team structures: Define the teams that can provide various cloud functions.
	RACI matrix: Clearly defined roles are an important aspect of any operating model. Use the provided RACI matrix to map responsibility, accountability, consulted, and informed roles to each of the teams for various functions of the cloud operating model.

Structure type

The following organizational structures do not necessarily have to map to an organizational chart (org chart). Org charts generally reflect command and control management structures. Conversely, the following organizational structures are designed to capture alignment of roles and responsibilities. In an agile, matrix organization, these structures may be best represented as virtual teams. There is no limitation suggesting that these organizational structures couldn't be represented in an org chart, but it is not necessary in order to produce an effective operating model.

The first step of managing organizational alignment is to determine how the following organizational structures will be fulfilled:

- **Org chart alignment:** Management hierarchies, manager responsibilities, and staff alignment will align to organizational structures.
- **Virtual teams:** Management structures and org charts remain unchanged. Instead, virtual teams will be

created and tasked with the required functions.

- **Mixed model:** More commonly, a mixture of org chart and virtual team alignment will be required to deliver on transformation goals.

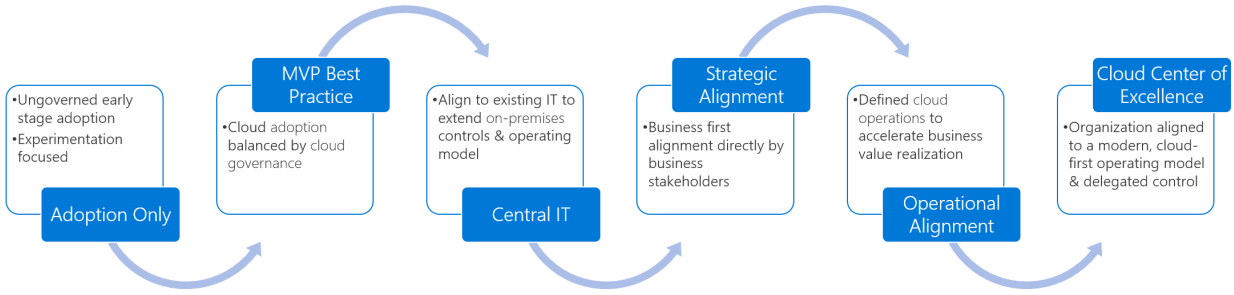
Understand required cloud functions

The following is a list of functions that are required to succeed at cloud adoption and longer-term operating models. After you become familiar with these, they can be aligned to organizational structures based on staffing and maturity:

- Cloud strategy: Align technical change to business needs.
- Cloud adoption: Deliver technical solutions.
- Cloud governance: Manage risk.
- Central IT team: Support from existing IT staff.
- Cloud operations: Support and operate adopted solutions.
- Cloud center of excellence: Improve quality, speed, and resiliency of adoption.
- Cloud platform: Operate and mature the platform.
- Cloud automation: Accelerate adoption and innovation.
- Cloud data: Manage data and enable analytics solutions.
- Cloud security: Manage information security risk.

To some degree, each of these functions are delivered in every cloud adoption effort, either explicitly or in accordance with a defined team structure.

As adoption needs grow, so does the need to create balance and structure. To meet those needs, companies often follow a process of maturing organizational structures.



The article on determining organizational structure maturity provides additional detail regarding each level of maturity.

To track organization structure decisions over time, download and modify the RACI template.

Cloud strategy functions

6/8/2021 • 3 minutes to read • Edit Online

A cloud strategy team defines motivations and business outcomes, and validates and maintains alignment between business priorities and cloud adoption efforts. In the absence of a defined cloud strategy team, someone must still provide the functionality that aligns technical activities to business outcomes. That same person or group should also manage change across the project.

Cloud strategy functions are commonly provided by the following types of roles. When a cloud strategy team is defined, it should include many of the following roles:

- Finance
- Line of business
- Human resources
- Operations
- Enterprise architecture
- IT infrastructure
- Application groups
- Project managers (often with agile project management experience)

This helps guide critical prioritization and discovery efforts during cloud adoption. It may also trigger changes in business processes, the execution of operations, customer interactions, or even product development. If these functions are confined to IT, the success of cloud adoption efforts will be constrained. To drive true business change, business leaders should be the primary source of this functionality. A defined cloud strategy team provides a means for involving key participants in a structured way.

> **NOTE**
> The organization's CEO and CIO often assign the team. Assignments are typically based on empowering this team to drive change that cuts across various different organizations within the enterprise. The cloud strategy team members should be assigned based on the motivations for cloud adoption, business outcomes, and relevant financial models.

Preparation

- Learn the business value of Microsoft Azure.
- Learn how the Cloud Adoption Framework can help you align the strategy for business, people, and technology.
- Review the cloud adoption strategy process.
- Download the strategy and plan template.

Minimum scope

Align business stakeholders to maximize the business value of cloud adoption investments.

Whenever possible, business outcomes and the cloud strategy should both be defined early in the process. As investments in cloud adoption grows and business values are realized, business stakeholders often become more engaged. When cloud adoption efforts are led by the business, the focus might be on an operating model and the organization.

Establish a vision

- Adoption motivations: Document and articulate the reasons behind the technical effort.
- Business outcomes: Clearly articulate what's expected of the technical team in terms of business changes.
- Learning metrics: Establish short-term metrics that can show progress toward longer-term business outcomes.

Build business justification

- Cloud migration business case. Establish a business case for cloud migration.

Rationalize the digital estate

- Incremental rationalization: An agile approach to rationalization that properly aligns late-bound technical decisions.
- The five Rs of rationalization: Understand the various rationalization options.

Deliverable

The cloud strategy team drives critical prioritization and discovery efforts during cloud adoption. They may also change business processes, the execution of operations, customer interactions, or even product development. The primary focus of the cloud strategy team is to validate and maintain alignment between business priorities and cloud adoption efforts. Secondarily, this team should focus on change management across the adoption efforts. The cloud strategy team should be capable of delivering on the following tasks.

Early planning tasks:

- Review and provide feedback on business outcomes and financial models.
- Aid in establishing clear motivations for cloud adoption that align with corporate objectives.
- Define relevant learning metrics that clearly communicate progress toward business outcomes.
- Understand business risks introduced by the plan, represent the business's tolerance for risk.
- Review and approve the rationalization of the digital estate.

Ongoing monthly tasks:

- Support the cloud governance team during risk/tolerance conversations.
- Review release plans to understand timelines and business impact of technical change.
- Define business change plans associated with planned releases.
- Ensure business teams are ready to execute business testing and the business change plan.

Meeting cadence:

Cloud strategy team members must be able to allocate time to planning and developing the cloud strategy:

- During early planning efforts, allocate an hour each week to meet with the team. After the adoption plan is solidified (usually within 4-6 weeks), the time requirements can be reduced.
- Throughout the adoption efforts, allocate 1-2 hours each month to review progress and validate continued priorities.
- Additional time is likely required from delegated members of the executive's team on an as-needed basis. Each member of the cloud strategy team should appoint a delegate who can allocate 5-10 hours per week to support ongoing prioritization questions and report on any urgent needs.

Next steps

- Start a cloud strategy team
- Align your strategy with the cloud adoption functions by creating a cloud adoption team to work with.
- Use the RACI template to align responsibilities across teams.

Cloud adoption functions

3/6/2021 • 3 minutes to read • Edit Online

Cloud adoption functions enable the implementation of technical solutions in the cloud. Like any IT project, the people delivering the actual work will determine success. The teams providing the necessary cloud adoption functions can be staffed from multiple subject matter experts or implementation partners.

Cloud adoption teams are the modern-day equivalent of technical implementation teams or project teams. But the nature of the cloud may require a more fluid team structure. Some teams focus exclusively on cloud migration, while other teams focus on innovations that take advantage of cloud technologies. Some teams include the broad technical expertise required to complete large adoption efforts, like a full datacenter migration. Other teams have a tighter technical focus and may move between projects to accomplish specific goals. One example would be a team of data platform specialists who help convert SQL VMs to SQL PaaS instances.

Regardless of the type or number of cloud adoption teams, the functionality required for cloud adoption is provided by subject matter experts found in IT, business analysis, or implementation partners.

Depending on the desired business outcomes, the skills needed to provide full cloud adoption functions could include:

- Infrastructure implementers
- DevOps engineers
- Application developers
- Data scientists
- Data or application platform specialists

For optimal collaboration and efficiency, we recommend that cloud adoption teams have an average team size of six people. These teams should be self-organizing from a technical execution perspective. We highly recommend that these teams also include project management expertise, with deep experience in agile, Scrum, or other iterative models. This team is most effective when managed using a flat structure.

Preparation

- Create an Azure account: The first step to using Azure is to create an account.
- Azure portal: Tour the Azure portal features and services, and customize the portal.
- Introduction to Azure: Get started with Azure. Create and configure your first virtual machine in the cloud.
- Azure fundamentals: Learn cloud concepts, understand the benefits, compare and contrast basic strategies, and explore the breadth of services available in Azure.
- Review the Migrate methodology.

Minimum scope

The nucleus of all cloud adoption efforts is the cloud migration team. This team drives the technical changes that enable adoption. Depending on the objectives of the adoption effort, this team may include a diverse range of team members who handle a broad set of technical and business tasks.

At a minimum, the team scope includes:

- Rationalization of the digital estate
- Review, validation, and advancement of the prioritized migration backlog

- The execution of the first workload as a learning opportunity.

Deliverable

The primary need from any cloud adoption function is the timely, high-quality implementation of the technical solutions outlined in the adoption plan. These solutions should align with governance requirements and business outcomes, and should take advantage of technology, tools, and automation solutions that are available to the team.

Early planning tasks:

- Execute the rationalization of the digital estate.
- Review, validate, and advance the prioritized migration backlog.
- Begin execution of the first workload as a learning opportunity.

Ongoing monthly tasks:

- Oversee change management processes.
- Manage the release and sprint backlogs.
- Build and maintain the adoption landing zone in conjunction with governance requirements.
- Execute the technical tasks outlined in the sprint backlogs.

Meeting cadence:

We recommend that teams providing cloud adoption functions be dedicated to the effort full-time.

It's best if these teams meet daily in a self-organizing way. The goal of daily meetings is to quickly update the backlog, and to communicate what has been completed, what is to be done today, and what things are blocked, requiring additional external support.

Release schedules and iteration durations are unique to each company. But a range of one to four weeks per iteration seems to be the average duration. Regardless of iteration or release cadence, we recommend that the team meets all supporting teams at the end of each release to communicate the outcome of the release, and to reprioritize upcoming efforts. Likewise, it's valuable to meet as a team at the end of each sprint, with the cloud center of excellence or cloud governance team to stay aligned on common efforts and any needs for support.

Some of the technical tasks associated with cloud adoption can become repetitive. Team members should rotate every 3–6 months to avoid employee satisfaction issues and maintain relevant skills. A rotating seat on a cloud center of excellence or cloud governance team can provide an excellent opportunity to keep employees fresh and harness new innovations.

Learn more about the function of a cloud center of excellence or cloud governance team.

Next steps

- Build a cloud adoption team
- Align cloud adoption efforts with cloud governance functions to accelerate adoption and implement best practices, while reducing business and technical risks.

Cloud governance functions

5/26/2021 • 7 minutes to read • Edit Online

A cloud governance team ensure that risks and risk tolerance are properly evaluated and managed. This team ensures the proper identification of risks that can't be tolerated by the business. The people on this team convert risks into governing corporate policies.

Depending on the desired business outcomes, the skills needed to provide full cloud governance functions include:

- IT governance
- Enterprise architecture
- Security
- IT operations
- IT infrastructure
- Networking
- Identity
- Virtualization
- Business continuity and disaster recovery
- Application owners within IT
- Finance owners

These baseline functions help you identify risks related to current and future releases. These efforts help you evaluate risk, understand the potential impacts, and make decisions regarding risk tolerance. When doing so, quickly update plans to reflect the changing needs of the cloud migration team.

Preparation

- Review the Govern methodology.
- Take the governance benchmark assessment.
- Introduction to security in Azure: Learn the basic concepts to protect your infrastructure and data in the cloud. Understand what responsibilities are yours and what Azure handles for you.
- Understand how to work across groups to manage cost.

Minimum scope

- Understand business risks introduced by the plan.
- Represent the business's tolerance for risk.
- Help create a governance MVP.

Involve the following participants in cloud governance activities:

- Leaders from middle management and direct contributors in key roles should represent the business and help evaluate risk tolerances.
- The cloud governance functions are delivered by an extension of the cloud strategy team. Just as the CIO and business leaders are expected to participate in cloud strategy functions, their direct reports are expected to participate in cloud governance activities.
- Business employees that are members of the business unit who work closely with the leadership of the line-of-business should be empowered to make decisions regarding corporate and technical risk.

- Information technology (IT) and information security (IS) employees who understand the technical aspects of the cloud transformation may serve in a rotating capacity instead of being a consistent provider of cloud governance functions.

Deliverable

The cloud governance mission is to balance competing forces of transformation and risk mitigation. Additionally, cloud governance ensures that the cloud migration team is aware of data and asset classification, as well as architecture guidelines that govern adoption. Governance teams or individuals also works with the cloud center of excellence to apply automated approaches to governing cloud environments.

Ongoing monthly tasks:

- Understand business risks introduced during each release.
- Represent the business's tolerance for risk.
- Aid in the incremental improvement of policy and compliance requirements.

Meeting cadence:

The time commitment from each team member of the cloud governance team will represent a large percentage of their daily schedules. Contributions will not be limited to meetings and feedback cycles.

Out of scope

As adoption scales, the cloud governance team may struggle to keep pace with innovations. This is especially true if your environment has heavy compliance, operations, or security requirements. If this happens you can shift some responsibilities to an existing IT team to reduce scope for the governance team.

Next steps

Some large organizations have dedicated teams that focus on IT governance. These teams specialize in risk management across the IT portfolio. When those teams exist, the following maturity models can be accelerated quickly. But the IT governance team is encouraged to review the cloud governance model to understand how governance shifts slightly in the cloud. Key articles include extending corporate policy to the cloud and the Five Disciplines of Cloud Governance.

No governance: Organizations often move into the cloud with no clear plans for governance. Before long, concerns around security, cost, scale, and operations begin to trigger conversations about the need for a governance model and people to staff the processes associated with that model. Starting those conversations before they become concerns is always a good first step to overcome the antipattern of "no governance." The section on defining corporate policy can help facilitate those conversations.

Governance blocked: When concerns around security, cost, scale, and operations go unanswered, projects and business goals tend to get blocked. Lack of proper governance generates fear, uncertainty, and doubt among stakeholders and engineers. Stop this in its tracks by taking action early. The two governance guides defined in the Cloud Adoption Framework can help you start small, set initially limiting policies to minimize uncertainty and mature governance over time. Choose from the complex enterprise guide or standard enterprise guide.

Voluntary governance: There tend to be brave souls in every enterprise. Those gallant few who are willing to jump in and help the team learn from their mistakes. Often this is how governance starts, especially in smaller companies. These brave souls volunteer time to fix some issues and push cloud adoption teams toward a consistent well-managed set of best practices.

The efforts of these individuals are much better than "no governance" or "governance blocked" scenarios. While their efforts should be commended, this approach should not be confused with governance. Proper governance requires more than sporadic support to drive consistency, which is the goal of any good governance approach.

The guidance in the Five Disciplines of Cloud Governance can help develop this discipline.

Cloud custodian: This moniker has become a badge of honor for many cloud architects who specialize in early stage governance. When governance practices first start out, the results appear similar to those of governance volunteers. But there is one fundamental difference. A cloud custodian has a plan in mind. At this stage of maturity, the team is spending time cleaning up the messes made by the cloud architects who came before them. But the cloud custodian aligns that effort to well structured corporate policy. They also use governance tools, like those outlined in the governance MVP.

Another fundamental difference between a cloud custodian and a governance volunteer is leadership support. The volunteer puts in extra hours above regular expectations because of their quest to learn and do. The cloud custodian gets support from leadership to reduce their daily duties to ensure regular allocations of time can be invested in improving cloud governance.

Cloud guardian: As governance practices solidify and become accepted by cloud adoption teams, the role of cloud architects who specialize in governance changes a bit, as does the role of the cloud governance team. Generally, the more mature practices gain the attention of other subject matter experts who can help strengthen the protections provided by governance implementations.

While the difference is subtle, it is an important distinction when building a governance-focused IT culture. A cloud custodian cleans up the messes made by innovative cloud architects, and the two roles have natural friction and opposing objectives. A cloud guardian helps keep the cloud safe, so other cloud architects can move more quickly with fewer messes.

Cloud guardians begin using more advanced governance approaches to accelerate platform deployment and help teams self-service their environmental needs, so they can move faster. Examples of these more advanced functions are seen in the incremental improvements to the governance MVP, such as improvement of the security baseline.

Cloud accelerators: Cloud guardians and cloud custodians naturally harvest scripts and governance tools that accelerate the deployment of environments, platforms, or even components of various applications. Curating and sharing these scripts in addition to centralized governance responsibilities develops a high degree of respect for these architects throughout IT.

Those governance practitioners who openly share their curated scripts help deliver technology projects faster and embed governance into the architecture of the workloads. This workload influence and support of good design patterns elevate cloud accelerators to a higher rank of governance specialist.

Global governance: When organizations depend on globally dispersed IT needs, there can be significant deviations in operations and governance in various geographies. Business unit demands and even local data sovereignty requirements can cause governance best practices to interfere with required operations. In these scenarios, a tiered governance model allows for minimally viable consistency and localized governance. The article on multiple layers of governance provides more insights on reaching this level of maturity.

Every company is unique, and so are their governance needs. Choose the level of maturity that fits your organization and use the Cloud Adoption Framework to guide the practices, processes, and tooling to help you get there.

As cloud governance matures, teams are empowered to adopt the cloud at faster paces. Continued cloud adoption efforts tend to trigger maturity in IT operations. Either develop a cloud operations team, or sync with your cloud operations team to ensure governance is a part of operations development.

Learn more about starting a cloud governance team or a cloud operations team.

After you've established an initial cloud governance foundation, use these best practices in Governance foundation improvements to get ahead of your adoption plan and prevent risks.

Central IT team functions

6/8/2021 • 7 minutes to read • Edit Online

As cloud adoption scales, cloud governance functions alone may not be sufficient to govern adoption efforts. When adoption is gradual, teams tend to organically develop the skills and processes needed to be ready for the cloud over time.

But when one cloud adoption team uses the cloud to achieve a high-profile business outcome, gradual adoption is seldom the case. Success follows success. This is also true for cloud adoption, but it happens at cloud scale. When cloud adoption expands from one team to multiple teams relatively quickly, additional support from existing IT staff is needed. But those staff members may lack the training and experience required to support the cloud using cloud-native IT tools. This often drives the formation of a central IT team governing the cloud.

Caution

While this is a common maturity step, it can present a high risk to adoption, potentially blocking innovation and migration efforts if not managed effectively. See the risk section below to learn how to mitigate the risk of centralization becoming a cultural antipattern.

The skills needed to provide centralized IT functions could be provided by:

- An existing central IT team
- Enterprise architects
- IT operations
- IT governance
- IT infrastructure
- Networking
- Identity
- Virtualization
- Business continuity and disaster recovery
- Application owners within IT

WARNING

Centralized IT should only be applied in the cloud when existing delivery on-premises is based on a central IT team model. If the current on-premises model is based on delegated control, consider a cloud center of excellence (CCoE) approach for a more cloud-compatible alternative.

Key responsibilities

Adapt existing IT practices to ensure adoption efforts result in well-governed, well-managed environments in the cloud.

The following tasks are typically executed regularly:

Strategic tasks

- Review:
  - Business outcomes.
  - Financial models.
  - Motivations for cloud adoption.

    - Business risks.
    - Rationalization of the digital estate.
- Monitor adoption plans and progress against the prioritized migration backlog.
- Identify and prioritize platform changes that are required to support the migration backlog.
- Act as an intermediary or translation layer between cloud adoption needs and existing IT teams.
- Take advantage of existing IT teams to accelerate platform functions and enable adoption.

Technical tasks

- Build and maintain the cloud platform to support solutions.
- Define and implement the platform architecture.
- Operate and manage the cloud platform.
- Continuously improve the platform.
- Keep up with new innovations in the cloud platform.
- Deliver new cloud functionality to support business value creation.
- Suggest self-service solutions.
- Ensure that solutions meet existing governance and compliance requirements.
- Create and validate deployment of platform architecture.
- Review release plans for sources of new platform requirements.

Meeting cadence

Central IT team expertise usually comes from a working team. Expect participants to commit much of their daily schedules to alignment efforts. Contributions aren't limited to meetings and feedback cycles.

Central IT team risks

Each of the cloud functions and phases of organizational maturity are prefixed with the word "cloud". The central IT team is the only exception. Centralized IT became prevalent when all IT assets could be housed in few locations, managed by a small number of teams, and controlled through a single operations management platform. Global business practices and the digital economy have largely reduced the instances of those centrally managed environments.

In the modern view of IT, assets are globally distributed. Responsibilities are delegated. Operations management is delivered by a mixture of internal staff, managed service providers, and cloud providers. In the digital economy, IT management practices are transitioning to a model of self-service and delegated control with clear guardrails to enforce governance. A central IT team can be a valuable contributor to cloud adoption by becoming a cloud broker and a partner for innovation and business agility.

A central IT team is well positioned to take valuable knowledge and practices from existing on-premises models and apply those practices to cloud delivery. But this process requires change. New processes, new skills, and new tools are required to support cloud adoption at scale. When a central IT team adapts, it becomes an important partner in cloud adoption efforts. But if the central IT team doesn't adapt to the cloud, or attempts to use the cloud as a catalyst for tight-grain controls, it quickly becomes a blocker to adoption, innovation, and migration.

The measures of this risk are speed and flexibility. The cloud simplifies adopting new technologies quickly. When new cloud functionality can be deployed within minutes, but the reviews by the central IT team add weeks or months to the deployment process, then these centralized processes become a major impediment to business success. When this indicator is encountered, consider alternative strategies to IT delivery.

Exceptions

Many industries require rigid adherence to third-party compliance. Some compliance requirements still demand

centralized IT control. Delivering on these compliance measures can add time to deployment processes, especially for new technologies that haven't been used broadly. In these scenarios, expect delays in deployment during the early stages of adoption. Similar situations my exist for companies that deal with sensitive customer data, but may not be governed by a third-party compliance requirement.

Operate within the exceptions

When centralized IT processes are required and those processes create appropriate checkpoints in adoption of new technologies, these innovation checkpoints can still be addressed quickly. Governance and compliance requirements are designed to protect those things that are sensitive, not to protect everything. The cloud provides simple mechanisms for acquiring and deploying isolated resources while maintaining proper guardrails.

A mature central IT team maintains necessary protections but negotiates practices that still enable innovation. Demonstrating this level of maturity depends on proper classification and isolation of resources.

Example narrative of operating within exceptions to empower adoption

This example narrative illustrates the approach taken by a mature central IT team at the fictional company Contoso to empower adoption.

Contoso has adopted a central IT team model for the support of the business's cloud resources. To deliver this model, they have implemented tight controls for various shared services such as ingress network connections. This wise move reduced the exposure of their cloud environment and provided a single "break-glass" device to block all traffic if a breach occurs. Their Security Baseline policies state that all ingress traffic must come through a shared device managed by the central IT team.

But one of their cloud adoption teams now requires an environment with a dedicated and specially configured ingress network connection to use a specific cloud technology. An immature central IT team would simply refuse the request and prioritize its existing processes over adoption needs. Contoso's central IT team is different. They quickly identified a simple four-part solution to this dilemma:

1. **Classification:** Since the cloud adoption team was in the early stages of building a new solution and didn't have any sensitive data or mission-critical support needs, the assets in the environment were classified as low risk and noncritical. Effective classification is a sign of maturity in a central IT team. Classifying all assets and environments allows for clearer policies.
2. **Negotiation:** Classification alone isn't sufficient. Shared services were implemented to consistently operate sensitive and mission-critical assets. Changing the rules would compromise governance and compliance policies designed for the assets that need more protection. Empowering adoption can't happen at the cost of stability, security, or governance. This led to a negotiation with the adoption team to answer specific questions. Could a business-led DevOps team provide operations management for this environment? Would this solution require direct access to other internal resources? If the cloud adoption team is comfortable with those tradeoffs, then the ingress traffic might be possible.
3. **Isolation:** Since the business can provide its own ongoing operations management, and since the solution doesn't rely on direct traffic to other internal assets, it can be cordoned off in a new subscription. That subscription is also added to a separate node of the new management group hierarchy.
4. **Automation:** Another sign of maturity in this team is their automation principles. The team uses Azure Policy to automate policy enforcement. They also use Azure Blueprints to automate deployment of common platform components and enforce adherence to the defined identity baseline. For this subscription and any others in the new management group, the policies and templates are slightly different. Policies blocking ingress bandwidth have been lifted. They have been replaced by requirements to route traffic through the shared services subscription, like any ingress traffic, to enforce traffic isolation. Since the on-premises operations management tooling can't access this subscription, agents for that tool are no longer required either. All other governance guardrails required by other subscriptions in the management group hierarchy are still enforced, ensuring sufficient guardrails.

The mature creative approach of Contoso's central IT team provided a solution that didn't compromise governance or compliance, but still encouraged adoption. This approach of brokering rather than owning cloud-native approaches to centralized IT is the first step toward building a cloud center of excellence (CCoE). Adopting this approach to quickly evolve existing policies will allow for centralized control when required and governance guardrails when more flexibility is acceptable. Balancing these two considerations mitigates the risks associated with centralized IT in the cloud.

Next steps

- As a central IT team matures its cloud capabilities, the next maturity step is typically looser coupling of cloud operations. The availability of cloud-native operations management tooling and lower operating costs for PaaS-first solutions often lead to business teams (or more specifically, DevOps teams within the business) assuming responsibility for cloud operations.

Learn more about:

- Building a cloud operations team
- Cloud operations functions

Cloud operations functions

6/8/2021 • 2 minutes to read • Edit Online

An operations team focuses on monitoring, repairing, and the remediation of issues related to traditional IT operations and assets. In the cloud, many of the capital costs and operations activities are transferred to the cloud provider, giving IT operations the opportunity to improve and provide significant additional value.

The skills needed to provide cloud operations functions can be provided by:

- IT operations
- Outsource IT operations vendors
- Cloud service providers
- Cloud-managed service providers
- Application-specific operations teams
- Business application operations teams
- DevOps teams

> **IMPORTANT**
> The individuals or teams accountable for cloud operations are generally responsible for making reactive changes to configuration during remediation. They're also likely to be responsible for proactive configuration changes to minimize operational disruptions. Depending on the organizations cloud operating model, those changes could be delivered via infrastructure-as-code, Azure Pipelines, or direct configuration in the portal. Since operations team will likely have elevated permissions, it is extremely important that those who fill this role are following identity and access control best practices to minimize unintended access or production changes.

Preparation

- Manage resources in Azure: Learn how to work through the Azure CLI and web portal to create, manage, and control cloud-based resources.
- Azure network services: Learn Azure networking basics and how to improve resiliency and reduce latency.

Review the following:

- Business outcomes
- Financial models
- Motivations for cloud adoption
- Business risks
- Rationalization of the digital estate

Minimum scope

The duties of the people on the cloud operations team involve delivering maximum workload performance and minimum business interruptions within an agreed-upon operations budget.

- Determine workload criticality, impact of disruptions, or performance degradation.
- Establish business-approved cost and performance commitments.
- Monitor and operate cloud workloads.

Deliverables

- Maintain asset and workload inventory
- Monitor performance of workloads
- Maintain operational compliance
- Protect workloads and associated assets
- Recover assets if there is performance degradation or business interruption
- Mature functionality of core platforms
- Continuously improve workload performance
- Improve budgetary and design requirements of workloads to fit commitments to the business

Meeting cadence

The cloud operations team should be involved in release planning and cloud center of excellence planning to provide feedback and prepare for operational requirements.

Out of scope

Traditional IT operations that focus on maintaining current-state operations for low-level technical assets is out of scope for the cloud operations team. Things like storage, CPU, memory, network equipment, servers, and virtual machine hosts require continuous maintenance, monitoring, repair, and remediation of issues to maintain peak operations. In the cloud, many of these capital costs and operations activities are transferred to the cloud provider.

Next steps

As adoption and operations scale, it's important to define and automate governance best practices that extend existing IT requirements. Forming a cloud center of excellence is an important step to scaling cloud adoption, cloud operations, and cloud governance efforts.

Learn more about:

- Cloud center of excellence functions.
- Organizational antipatterns: silos and fiefdoms.

Learn to align responsibilities across teams by developing a cross-team matrix that identifies responsible, accountable, consulted, and informed (RACI) parties. Download and modify the RACI template.

Cloud center of excellence (CCoE) functions

8/25/2021 • 9 minutes to read • Edit Online

Many IT organizations share the core objectives of achieving business and technical agility. A cloud center of excellence (CCoE) is a function that helps organizations to balance speed and stability while they pursue these objectives.

Function structure

A CCoE model requires collaboration between each of the following:

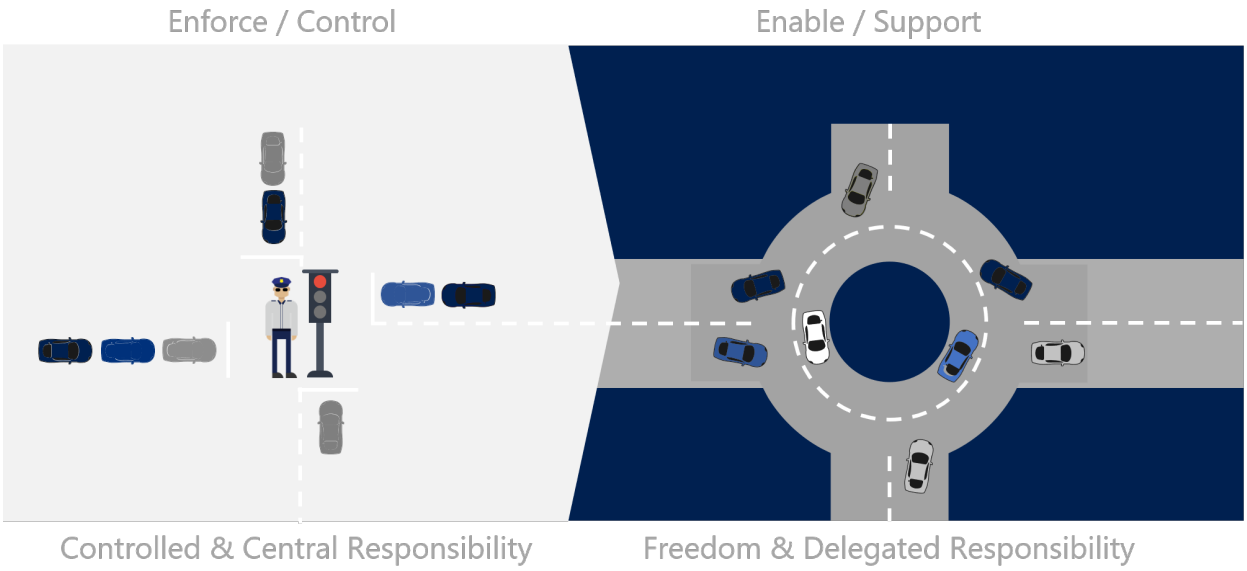
- Cloud adoption (specifically solution architects)
- Cloud strategy (specifically the program and project managers)
- Cloud governance
- Cloud platform
- Cloud automation

Impact and cultural change

When this function is properly structured and supported, the participants can accelerate innovation and migration efforts while reducing the overall cost of change and increasing business agility. When successfully implemented, this function can produce noticeable reductions in time-to-market. As team practices mature, quality indicators will improve, including reliability, performance efficiency, security, maintainability, and customer satisfaction. These gains in efficiency, agility, and quality are especially vital if the company plans on implementing large-scale cloud migration efforts or has a desire to use the cloud to drive innovations associated with market differentiation.

When successful, a CCoE model will create a significant cultural shift in IT. The fundamental premise of a CCoE approach is that IT serves as a broker, partner, or representative to the business. This model is a paradigm shift away from the traditional view of IT as an operations unit or abstraction layer between the business and IT assets.

The following image provides an analogy for this cultural change. Without a CCoE approach, IT tends to focus on providing control and central responsibility, acting like the stoplights at an intersection. When the CCoE is successful, this resembles a roundabout at an intersection where the focus is on freedom and delegated responsibility.



Neither approach illustrated in the analogy image above is right or wrong; they're alternative views of responsibility and management. If the desire is to establish a self-service model that allows business units to make their own decisions while adhering to a set of guidelines and established, repeatable controls, then a CCoE model could fit within the technology strategy.

Key responsibilities

The primary duty of the CCoE team is to accelerate cloud adoption through cloud-native or hybrid solutions.

The objective of the CCoE is to:

- Help build a modern IT organization through agile approaches to capture and implement business requirements.
- Use reusable deployment packages that align with security, compliance, and service management policies.
- Maintain a functional Azure platform in alignment with operational procedures.
- Review and approve the use of cloud-native tools.
- Over time, standardize and automate commonly needed platform components and solutions.

Meeting cadence

The CCoE is a function staffed by four high demand teams. It is important to allow for organic collaboration and track growth through a common repository/solution catalog. Maximize natural interactions, but minimize meetings. When this function matures, the teams should try to limit dedicated meetings. Attendance at recurring meetings, like release meetings hosted by the cloud adoption team, will provide data inputs. In parallel, a meeting after each release plan is shared can provide a minimum touch point for this team.

Solutions and controls

Each member of the CCoE is tasked with understanding the necessary constraints, risks, and protections that led to the current set of IT controls. The collective efforts of the CCoE should turn that understanding into cloud-native (or hybrid) solutions or controls, which enable the desired self-service business outcomes. As solutions are created, they're shared with various teams in the form of controls or automated processes that serve as guardrails for various efforts. Those guardrails help to route the free-flowing activities of various teams, while delegating responsibilities to the participants in various migration or innovation efforts.

Examples of this transition:

SCENARIO	PRE-CCOE SOLUTION	POST-CCOE SOLUTION
Provision a production SQL Server	Network, IT, and data platform teams provision various components over the course of days or even weeks.	The team requiring the server deploys a PaaS instance of Azure SQL Database. Alternatively, a preapproved template could be used to deploy all of the IaaS assets to the cloud in hours.
Provision a development environment	Network, IT, development, and DevOps teams agree to specs and deploy an environment.	The development team defines their own specs and deploys an environment based on allocated budget.
Update security requirements to improve data protection	Networking, IT, and security teams update various networking devices and VMs across multiple environments to add protections.	Cloud governance tools are used to update policies that can be applied immediately to all assets in all cloud environments.

Negotiations

An ongoing negotiation process is at the root of CCoE efforts. A CCoE team negotiates with existing IT functions to reduce central control. The trade-offs for the business in this negotiation are freedom, agility, and speed, and the value of the trade-off for existing IT teams is delivered as new solutions. New solutions provide the existing IT team with one or more of the following benefits:

- Ability to automate common issues
- Improvements in consistency (reduction in day-to-day frustrations)
- Opportunity to learn and deploy new technical solutions
- Reductions in high severity incidents (fewer quick fixes or late-night pager-duty responses)
- Ability to broaden their technical scope, addressing broader topics
- Participation in higher-level business solutions, addressing the impact of technology
- Reduction in menial maintenance tasks
- Increase in technology strategy and automation

In exchange for these benefits, the existing IT function may be trading the following values, whether real or perceived:

- Sense of control from manual approval processes
- Sense of stability from change control
- Sense of job security from completion of necessary yet repetitive tasks
- Sense of consistency that comes from adherence to existing IT solution vendors

In healthy cloud-forward companies, this negotiation process is a dynamic conversation between peers and partnering IT teams. The technical details may be complex, but are manageable when IT understands the objective and is supportive of the CCoE efforts. When IT is less than supportive, the following section on enabling CCoE success can help overcome cultural blockers.

Enable CCoE success

Before proceeding with this model, it is important to consider the company's tolerance for a growth mindset and IT's comfort level with releasing central responsibilities. As mentioned above, the purpose of a CCoE is to exchange control for agility and speed.

This type of change takes time, experimentation, and negotiation. There will be bumps and set backs during this maturation process. But if the team stays diligent and isn't discouraged from experimentation, there is a high

probability of success in improving agility, speed, and reliability. One of the biggest factors in success or failure of a CCoE is support from leadership and key stakeholders.

Key stakeholders

IT leadership is the first and most obvious stakeholder. IT managers will play an important part. But the support of the CIO and other executive-level IT leaders is needed during this process.

Less obvious is the need for business stakeholders. Business agility and time-to-market are key motivations for CCoE formation. As such, the key stakeholders should have a vested interest in these areas. Examples of business stakeholders include line-of-business leaders, finance executives, operations executives, and business product owners.

Business stakeholder support

CCoE efforts can be accelerated with support from the business stakeholders. Much of the focus of CCoE efforts is centered around making long-term improvements to business agility and speed. Defining the impact of current operating models and the value of improvements is valuable as a guide and negotiation tool for the CCoE. Documenting the following items is suggested for CCoE support:

- Establish a set of business outcomes and goals that are expected as a result of business agility and speed.
- Clearly define pain points created by current IT processes (such as speed, agility, stability, and cost challenges).
- Clearly define the historical impact of those pain points (such as lost market share, competitor gains in features and functions, poor customer experiences, and budget increases).
- Define business improvement opportunities that are blocked by the current pain points and operating models.
- Establish timelines and metrics related to those opportunities.

These data points are not an attack on IT. Instead, they help CCoE learn from the past and establish a realistic backlog and plan for improvement.

Ongoing support and engagement: CCoE teams can demonstrate quick returns in some areas. But the higher-level goals, like business agility and time-to-market, can take much longer. During maturation, there is a high risk of the CCoE becoming discouraged or being pulled off to focus on other IT efforts.

During the first six to nine months of CCoE efforts, we recommend that business stakeholders allocate time to meet monthly with the IT leadership and the CCoE. There is little need for formal ceremony to these meetings. Simply reminding the CCoE members and their leadership of the importance of this program can go along way to driving CCoE success.

Additionally, we recommend that the business stakeholders stay informed of the progress and blockers experienced by the CCoE team. Many of their efforts will seem like technical minutiae. But it is important for business stakeholders to understand the progress of the plan, so they can engage when the team loses steam or becomes distracted by other priorities.

IT stakeholder support

Support the vision: A successful CCoE effort requires a great deal of negotiation with existing IT team members. When done well, all of IT contributes to the solution and feels comfortable with the change. When this is not the case, some members of the existing IT team may want to hold on to control mechanisms for various reasons. Support of IT stakeholders will be vital to the success of the CCoE when those situations occur. Encouragement and reinforcement of the overall goals of the CCoE is important to resolve blockers to proper negotiation. On rare occasions, IT stakeholders may even need to step in and break up a deadlock or tied vote to keep the CCoE progressing.

Maintain focus: A CCoE can be a significant commitment for any resource-constrained IT team. Removing strong architects from short-term projects to focus on long-term gains can create difficulty for team members who aren't part of the CCoE. It is important that IT leadership and IT stakeholders stay focused on the goal of the CCoE. The support of IT leaders and IT stakeholders is required to deprioritize the disruptions of day-to-day operations in favor of CCoE duties.

Create a buffer: The CCoE team will experiment with new approaches. Some of those approaches won't align well with existing operations or technical constraints. There is a real risk of the CCoE experiencing pressure or recourse from other teams when experiments fail. Encouragement and buffering the team from the consequences of "fast fail" learning opportunities is important. It's equally important to hold the team accountable to a growth mindset, ensuring that they're learning from those experiments and finding better solutions.

Next steps

A CCoE model requires cloud platform functions and cloud automation functions. The next step is to align cloud platform functions.

Learn more about:

- Cloud platform functions
- Cloud automation functions

Cloud platform functions

6/8/2021 • 2 minutes to read • Edit Online

The cloud introduces many technical changes as well as opportunities to streamline technical solutions. But general IT principles and business needs stay the same. You still need to protect sensitive business data. If your IT platform depends on a local area network, there's a good chance that you'll need network definitions in the cloud. Users who need to access applications and data will want their current identities to access relevant cloud resources.

While the cloud presents the opportunity to learn new skills, your current architects should be able to directly apply their experiences and subject matter expertise. Cloud platform functions are usually provided by a select group of architects who focus on learning about the cloud platform. These architects then aid others in decision making and the proper application of controls to cloud environments.

The skills needed to provide full platform functionality can be provided by:

- Enterprise architecture
- IT operations
- IT governance
- IT infrastructure
- Networking
- Identity
- Virtualization
- Business continuity and disaster recovery
- Application owners within IT

Preparation

- Foundations for Cloud Architecture: A Pluralsight course to help architect the right foundational solutions.
- Microsoft Azure architecture: A Pluralsight course to ground architects in Azure architecture.
- Azure network services: Learn Azure networking basics and how to improve resiliency and reduce latency.

Review the following:

- Business outcomes
- Financial models
- Motivations for cloud adoption
- Business risks
- Rationalization of the digital estate

Minimum scope

Cloud platform duties center around the creation and support of your cloud platform or landing zones.

The following tasks are typically executed on a regular basis:

- Monitor adoption plans and progress against the prioritized migration backlog.
- Identify and prioritize platform changes that are required to support the migration backlog.

Meeting cadence:

Cloud platform expertise usually comes from a working team. Expect participants to commit a large portion of their daily schedules to cloud platform work. Contributions aren't limited to meetings and feedback cycles.

Deliverables

- Build and maintain the cloud platform to support solutions.
- Define and implement the platform architecture.
- Operate and manage the cloud platform.
- Continuously improve the platform.
- Keep up with new innovations in the cloud platform.
- Bring new cloud functionality to support business value creation.
- Suggest self-service solutions.
- Ensure solutions meet existing governance and compliance requirements.
- Create and validate deployment of platform architecture.
- Review release plans for sources of new platform requirements.

Next steps

As your cloud platform becomes better defined, aligning cloud automation functions can accelerate adoption. It can also help establish best practices while reducing business and technical risks.

Learn to align responsibilities across teams by developing a cross-team matrix that identifies responsible, accountable, consulted, and informed (RACI) parties. Download and modify the RACI template.

Cloud automation functions

5/26/2021 • 2 minutes to read • Edit Online

During cloud adoption efforts, cloud automation functions unlock the potential of DevOps and a cloud-native approach. Expertise in each of these areas can accelerate adoption and innovation.

The skills needed to provide cloud automation functions can be provided by:

- DevOps engineers
- Developers with DevOps and infrastructure expertise
- IT engineers with DevOps and automation expertise

These subject matter experts might be providing functions in other areas such as cloud adoption, cloud governance, or cloud platform. After they demonstrate proficiency at automating complex workloads, you can recruit these experts to deliver automation value.

Preparation

Before you admit a team member to this group, they should demonstrate three key characteristics:

- Expertise in any cloud platform with a special emphasis on DevOps and automation.
- A growth mindset or openness to changing the way IT operates today.
- A desire to accelerate business change and remove traditional IT roadblocks.

Minimum scope

The primary duty of cloud automation is to own and advance the solution catalog. The solution catalog is a collection of prebuilt solutions or automation templates. These solutions can rapidly deploy various platforms as required to support needed workloads. These solutions are building blocks that accelerate cloud adoption and reduce the time to market during migration or innovation efforts.

Examples of solutions in the catalog include:

- A script to deploy a containerized application.
- A Resource Manager template to deploy a SQL HA AO cluster.
- Sample code to build a deployment pipeline using Azure DevOps.
- An Azure DevTest Labs instance of the corporate ERP for development purposes.
- Automated deployment of a self-service environment commonly requested by business users.

The solutions in the solution catalog aren't deployment pipelines for a workload. Instead, you might use automation scripts in the catalog to quickly create a deployment pipeline. You might also use a solution in the catalog to quickly provision platform components to support workload tasks like automated deployment, manual deployment, or migration.

Strategic tasks

- Rationalization of the digital estate:
  - Monitor adoption plans and progress against the prioritized migration backlog.
  - Identify opportunities to accelerate cloud adoption, reduce effort through automation, and improve security, stability, and consistency.
  - Prioritize a backlog of solutions for the solution catalog that delivers the most value given other strategic inputs.

- Review release plans for sources of new automation opportunities.

Meeting cadence:

Cloud automation is a working team. Expect participants to commit a large portion of their daily schedules to cloud automation work. Contributions aren't limited to meetings and feedback cycles.

The cloud automation team should align activities with other areas of capability. This alignment might result in meeting fatigue. To ensure cloud automation has sufficient time to manage the solution catalog, you should review meeting cadences to maximize collaboration and minimize disruptions to development activities.

Deliverables

- Curate or develop solutions based on the prioritized backlog.
- Ensure solutions align to platform requirements.
- Ensure solutions are consistently applied and meet existing governance and compliance requirements.
- Create and validate solutions in the catalog.

Next steps

As essential cloud functions align, the collective teams can help develop necessary technical skills.

Understand cloud data functions

3/6/2021 • 3 minutes to read • Edit Online

There are multiple audiences involved in an analytics conversation, including the typical seller, database architect, and infrastructure team. In addition, analytics solutions involve influencers, recommenders, and decision-makers from enterprise architecture, data science, business analysts, and executive leadership roles.

Azure Synapse Analytics enables the entire business, from the IT stakeholder to the business analyst, to collaborate on analytics solutions and understand cloud data functions. The following sections discuss these roles in more detail.

Database administrators and architects

Database administrators and architects are responsible for integrating and routing data sources into a centralized repository. These experts also handle the administration and performance required for the system, and the accessibility and efficiency of query and analytic modeling against that data.

Using Azure Synapse Analytics, database administrators can match their expanding responsibilities for data warehouses and data lakes. They can use familiar languages and tools, such as T-SQL, to run as many workloads as they want. They can assign resources to escalate critical workloads based on intelligent workload importance, workload isolation, and enhanced concurrency capabilities.

Infrastructure teams

These teams deal with the provisioning and architecture of the underlying compute resources required for large analytics systems. In many cases, they are managing transitions between datacenter-based and cloud-based systems, and current needs for interoperability across both. Disaster recovery, business continuity, and high availability are common concerns.

With Azure Synapse Analytics, IT professionals can protect and manage their organization's data more efficiently. They can enable big data processing with both on-demand and provisioned compute. Through tight integration with Azure Active Directory, the service helps secure access to cloud and hybrid configurations. IT professionals can enforce privacy requirements by using data masking along with row-level and column-level security.

Enterprise architects and data engineers

These teams are responsible for putting together complex solutions with components spanning integration across a wide swath of data tools and solutions. These include:

- Structured and unstructured data
- Transformation
- Storage and retrieval
- Analytic modeling
- Message-based middleware
- Data marts
- Geo-redundancy and data consistency
- Dashboards and reporting

Enterprise architects and data engineers are concerned with building effective architectures that work in an integrated manner. Such architectures preserve performance, availability, ease of administration,

flexibility/extensibility, and actionability.

Data engineers can use Azure Synapse Analytics to simplify the steps to wrangle multiple data types from multiple sources, including streaming, transactional, and business data. They can use a code-free visual environment to connect to data sources and ingest, transform, and place data in the data lake.

Data scientists

Data scientists understand how to build advanced models for huge volumes of critical, yet often disparate data. Their work involves translating the needs of the business into the technology requirements for normalization and transformation of data. They create statistical and other analytical models, and ensure that line-of-business teams can get the analysis they need to run the business.

Using Azure Synapse Analytics, data scientists can build proofs of concept in minutes, and create or adjust end-to-end solutions. They can provision resources as needed, or query existing resources on demand across massive amounts of data. They can do their work in various languages, including T-SQL, R, Python, Scala, .NET, and Spark SQL.

Business analysts

These teams build and use dashboards, reports, and other forms of data visualization to gain rapid insights required for operations. Often, each line-of-business department will have dedicated business analysts who gather and package information and analytics from specialized applications. These specialized applications can be for credit cards, retail banking, commercial banking, treasury, marketing, and other organizations.

Using Azure Synapse Analytics, business analysts can securely access datasets and use Power BI to build dashboards. They can also securely share data within and outside their organization through Azure Data Share.

Executives

Executives are responsible for charting strategy and ensuring strategic initiatives are implemented effectively across both IT and line-of-business departments. Solutions must be cost-effective, prevent disruption to the business, allow for easy extensibility as requirements change and grow, and deliver results to the business.

Cloud security functions

6/8/2021 • 4 minutes to read • Edit Online

This article provides a summary of the organizational functions required to manage information security risk in an enterprise. These roles and responsibilities form the human portion of an overall cybersecurity system.

Security is a team sport

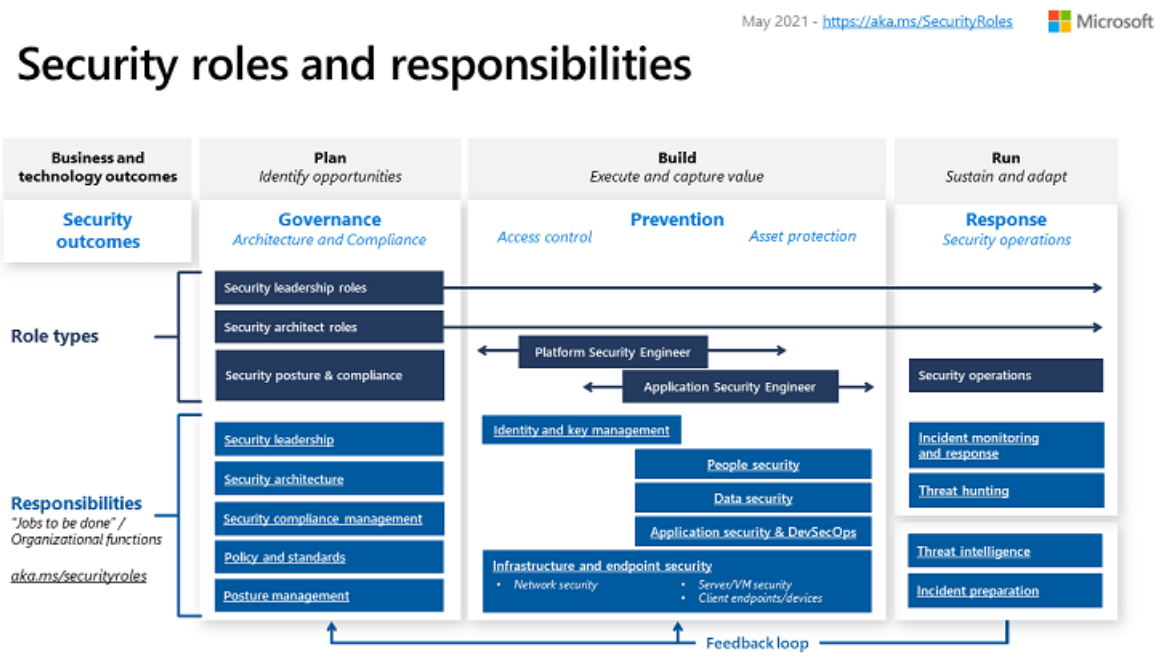
It's critical that individuals on the security team see each other as part of a whole security team, part of the whole organization, and part of a larger security community defending against the same adversaries.

This holistic worldview enables the team to work well in general. It's especially important as the teams work through any unplanned gaps and overlaps discovered during the evolution of roles and responsibilities.

Security responsibilities (functions)

This diagram depicts the specific organizational functions within security, often called responsibilities or "jobs to be done".

These diagrams and documentation represent an ideal view of a complete enterprise security team. This may be an aspirational view for security teams with limited resources that may not have formal responsibilities defined around all of these functions. Each function may be performed by one or more people, and each person may perform one or more functions depending on various factors such as culture, budget, and available resources.



Each of the following articles provides information about each function including a summary of objectives, how the function can evolve because of the threat environment or cloud technology changes, and the relationships and dependencies that are critical to its success.

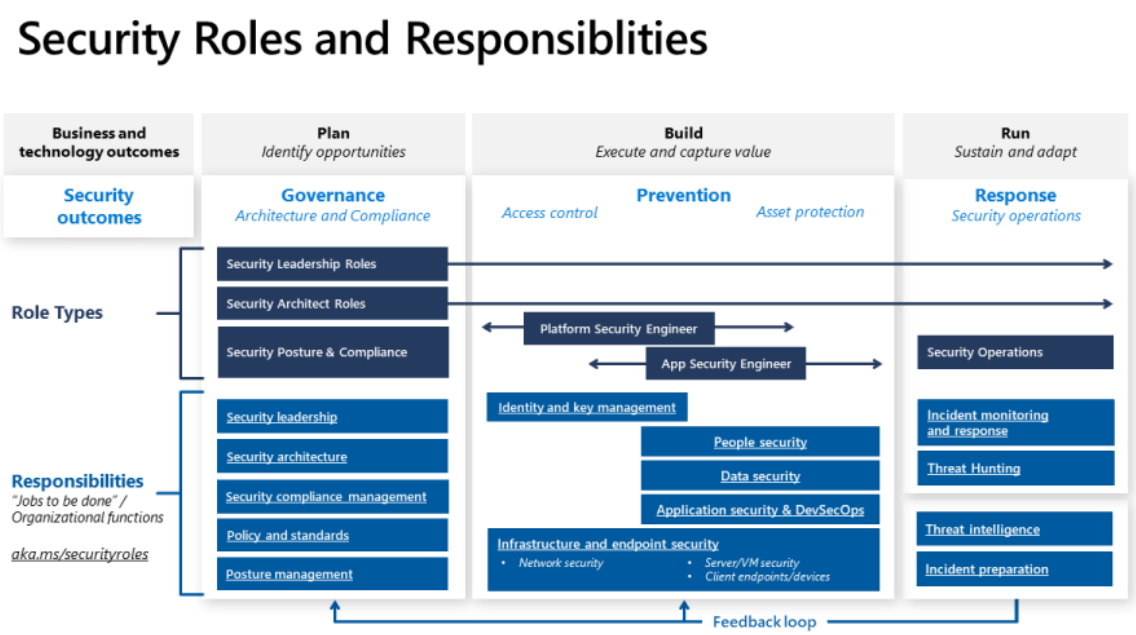
- Policy and standards
- Security operations
- Security architecture
- Security compliance management

- People security
- Application security and DevSecOps
- Data security
- Infrastructure and endpoint security
- Identity and key management
- Threat intelligence
- Posture management
- Incident preparation

These responsibilities are referenced throughout Microsoft documentation including the Azure Security Benchmark, securing privileged access: rapid modernization plan, and the Azure security top 10,

Roles and responsibilities

The following diagram depicts how these functions map to role types within an organization:



Mapping security to business outcomes

At the organizational level, the security disciplines map to standard plan-build-run phases seen widely across industries and organizations. While this cycle is accelerating into a continuous change cycle with the digital age and the advent of DevOps, this illustrates how security maps to normal business processes.

Security is both discipline with its own unique functions *and* a critical element to integrate into normal business operations.

Role types

The middle (dark blue) section groups these responsibilities into typical roles that have common skill sets and career profiles. These groupings also help provide clarity on how industry trends are affecting security professionals:

- **Security leadership:** These roles frequently span across functions, ensuring that teams coordinate with each other, providing prioritization and setting cultural norms, policies, and standards for security.
- **Security architect:** These roles span across functions and provide a key governance capability to ensure all of the technical functions work harmoniously within a consistent architecture

- **Security posture and compliance:** This is a newer role type that represents the increasing convergence of compliance reporting with traditional security disciplines like vulnerability management and configuration baselines. While the scope and audience are different for security and compliance reporting, they are answering different versions of the question of "how secure is the organization?". The way that question is answered is growing more similar via tools like Microsoft Secure Score and Azure Security Center:
  - The use of on-demand data feeds from cloud services is reducing the time required to report compliance.
  - The increased scope of data available is enabling security governance to look beyond traditional software updates/patches and discover/track "vulnerabilities" from security configurations and operational practices
- **Platform security engineer:** These are technology roles focused on platforms that host multiple workloads, focused on both access control and asset protection. These roles are often grouped into teams with specialized technical skill sets including network security, infrastructure and endpoints, identity and key management, and others. These teams work on both preventive controls and detective controls, with detective controls being a partnership with SecOps and preventive controls being primarily a partnership with IT operations. For more information, see Security integration.
- **Application security engineer:** These technology roles focused on security controls for specific workloads, supporting both classic development models and modern DevOps/DevSecOps model. This is a blend of application/development security skills for unique code and infrastructure skills for common technical components like VMs, databases, and containers. These roles may be located in central IT or security organizations or within business and development teams, depending on organizational factors.

> **NOTE**
>
> As both DevOps and infrastructure as code trends progress, we expect to see some security talent migrate from platform security engineering teams to application security teams and posture management roles. This is because the DevOps model requires infrastructure security skills (such as securing the *ops* in DevOps) and governance teams will also require these skills and experience to effectively monitor technical security posture in real time. Additionally, infrastructure as code will automate repetitive manual technical tasks, reducing the volume of time required for these skills in the platform security engineer roles (though increasing the need for broad technical skill sets and automation or scripting skills).

Next steps

Review the Secure methodology

Function of cloud security policy and standards

3/6/2021 • 2 minutes to read • Edit Online

Security policy and standards teams author, approve, and publish security policy and standards to guide security decisions within the organization.

The policies and standards should:

- Reflect the organizations security strategy at a detailed enough way to guide decisions in the organization by various teams
- Enable productivity throughout the organization while reducing risk to the organizations business and mission

Security policy should reflect long term sustainable objectives that align to the organizations security strategy and risk tolerance. Policy should always address:

- Regulatory compliance requirements and current compliance status (requirements met, risks accepted, and so on.)
- Architectural assessment of current state and what is technically possible to design, implement, and enforce
- Organizational culture and preferences
- Industry best practices
- Accountability of security risk assigned to appropriate business stakeholders who are accountable for other risks and business outcomes.

Security standards define the processes and rules to support execution of the security policy.

Modernization

While policy should remain static, standards should be dynamic and continuously revisited to keep up with pace of change in cloud technology, threat environment, and business competitive landscape.

Because of this high rate of change, you should keep a close eye on how many exceptions are being made as this may indicate a need to adjust standards (or policy).

Security standards should include guidance specific to the adoption of cloud such as:

- Secure use of cloud platforms for hosting workloads
- Secure use of DevOps model and inclusion of cloud applications, APIs, and services in development
- Use of identity perimeter controls to supplement or replace network perimeter controls
- Define your segmentation strategy prior to moving your workloads to IaaS platform
- Tagging and classifying the sensitivity of assets
- Define process for assessing and ensuring your assets are configured and secured properly

Team composition and key relationships

Cloud security policy and standards are commonly provided by the following types of roles. The organizational policy should inform (and be informed by):

- Security architectures
- Compliance and risk management teams
- Business unit's leadership and representatives

- Information technology
- Audit and legal teams

The policy should be refined based on many inputs/requirements from across the organization, including but not restricted to those depicted in the security overview diagram.

Next steps

Review the function of a cloud security operations center (SOC).

Security operations (SecOps) functions

5/24/2021 • 2 minutes to read • Edit Online

The main objective of a cloud security operations (SecOps) function is to detect, respond to, and recover from active attacks on enterprise assets.

As SecOps matures, security operations should:

- Reactively respond to attacks detected by tools
- Proactively hunt for attacks that slipped past reactive detections

Modernization

Detecting and responding to threats is currently undergoing significant modernization at all levels.

- **Elevation to business risk management:** SOC is growing into a key component of managing business risk for the organization
- **Metrics and goals:** Tracking SOC effectiveness is evolving from "time to detect" to these key indicators:
  - **Responsiveness** via mean time to acknowledge (MTTA).
  - **Remediation speed** via mean time to remediate (MTTR).
- **Technology evolution:** SOC technology is evolving from exclusive use of static analysis of logs in a SIEM to add the use of specialized tooling and sophisticated analysis techniques. This provides deep insights into assets that provide high quality alerts and investigation experience that complement the breadth view of the SIEM. Both types of tooling are increasingly using AI and machine learning, behavior analytics, and integrated threat intelligence to help spot and prioritize anomalous actions that could be a malicious attacker.
- **Threat hunting:** SOCs are adding hypothesis driven threat hunting to proactively identify advanced attackers and shift noisy alerts out of frontline analyst queues.
- **Incident management:** Discipline is becoming formalized to coordinate nontechnical elements of incidents with legal, communications, and other teams. **Integration of internal context:** To help prioritize SOC activities such as the relative risk scores of user accounts and devices, sensitivity of data and applications, and key security isolation boundaries to closely defend.

For more information, see:

- Security operations discipline
- Security operations best practices videos and slides
- CISO workshop module 4b: threat protection strategy
- Cyber Defense Operations Center (CDOC) blog series part 1, part 2a, part 2b, part 3a, part 3b, part 3c, part 3d
- NIST computer security incident handling guide
- NIST guide for cybersecurity event recovery

Team composition and key relationships

The cloud security operations center is commonly made up of the following types of roles.

- IT operations (close regular contact)
- Threat intelligence
- Security architecture
- Insider risk program

- Legal and human resources
- Communications teams
- Risk organization (if present)
- Industry specific associations, communities, and vendors (before incident occurs)

Next steps

Review the function of security architecture.

Cloud security architecture functions

3/6/2021 • 2 minutes to read • Edit Online

Security architecture translates the organizations business and assurance goals into documentation and diagrams to guide technical security decisions.

Modernization

Security architecture is affected by different factors:

- **Continuous engagement model:** Continuous release of software updates and cloud features make fixed engagement models obsolete. Architects should be engaged with all teams working in technical topic areas to guide decision making along those teams' capability lifecycles.
- **Security from the cloud:** Incorporate security capabilities from the cloud to reduce enablement time and ongoing maintenance costs (hardware, software, time, and effort).
- **Security of the cloud:** Ensure coverage of all cloud assets including software as a service (SaaS) applications, infrastructure as a service (IaaS) VMs, and platform as a service (PaaS) applications and services. This should include discovery and security of both sanctioned and unsanctioned services.
- **Identity integration:** Security architects should ensure tight alignment with identity teams to help organizations meet the dual goals of enabling productivity and providing security assurances.
- **Integration of internal context** in security designs to such as context from posture management and incidents investigated by security operations [center] (SOC). This should include elements like relative risk scores of user accounts and devices, sensitivity of data, and key security isolation boundaries to actively defend.

Team composition and key relationships

Security architecture is ideally provided by a dedicated individual or dedicated team, but resource constraints may require assigning this function to an individual with other responsibilities.

Security architecture should have a broad portfolio of relationships across the security organization, with key stakeholders in other organizations, and with peers in external organizations. Key internal relationships should include:

- IT/enterprise architects
- Security posture management
- Technology directors
- Key business leaders or their representatives
- Industry peers and others in the security community

Security architects should actively influence security policy and standards.

Next steps

Review the function of cloud security compliance management.

Cloud security compliance management functions

3/6/2021 • 2 minutes to read • Edit Online

The objective of cloud security compliance management is to ensure that the organization is compliant with regulatory requirements (and internal policies) and efficiently tracks and reports status.

Modernization

Cloud introduces changes to security compliance including:

- **Requirement to validate** the compliance status of the cloud provider with your regulatory requirements. This validation is a shared responsibility. For more information about how these responsibilities differ for cloud types, see adopting the shared responsibility model
- **Pre-cloud guidance:** While many regulatory requirements have been updated to incorporate the dynamic nature of cloud services, some requirements do not yet reflect these differences. Organizations should work with regulatory bodies to update these requirements and be prepared to explain these differences during audit exercises.
- **Linking compliance with risk:** Ensure that organizations are tying compliance violations and exceptions to organizational risks to ensure the right level of attention and funding to correct issues.
- **Tracking and reporting enabled by cloud:** This function should actively embrace the software-defined nature of the cloud, since it offers comprehensive logging, configuration data, and analytical insights that make reporting on compliance more efficient than traditional on-premises approaches.
- **Cloud-based compliance tools** are available to facilitate easier reporting of regulatory compliance such as Microsoft Compliance Manager, which can reduce overhead costs of this function.

Team composition and key relationships

Cloud security compliance management frequently interacts with:

- Security operations
- IT operations
- Organizational compliance/risk management teams
- Audit and legal teams
- Key business leaders or their representatives

Next steps

Review the function of people security.

People security functions in the cloud

5/13/2021 • 2 minutes to read • Edit Online

People security protects the organization from risk of inadvertent human mistakes and malicious insider actions.

Modernization

Modernization of this function includes:

- **Increase positive engagement** with users using gamification and positive reinforcement / education rather than relying solely on negative reinforcement approaches like traditional "phish and punish" solutions.
- **High quality human engagement:** Security awareness communications and training should be high quality productions that drive empathy and emotional engagement to connect with the human side of employees and the organizations mission.
- **Realistic expectations:** Accept that users will sometimes open phishing emails, and instead focus success metrics on reducing the rate versus expecting to stop 100 percent of opening.
- **Organizational culture change:** Organizational leadership must drive an intentional culture change to make security a priority for each member of the organization.
- **Increased insider risk focus** to help organizations protect valuable trade secrets and other data with highly profitable illicit use cases (such as customer locations or communication records).
- **Improved insider risk detection** which takes advantage of cloud capabilities for activity logging, behavior analytics, and machine learning.

Team composition and key relationships

People security commonly partners with the following types of roles:

- Audit and legal teams
- Human resources
- Privacy team
- Data security
- Communications teams, for user awareness
- Security operations, for insider risk
- Physical security, for insider risk

Next steps

Review the function of application security and DevSecOps.

Application security and DevSecOps functions

3/6/2021 • 2 minutes to read • Edit Online

The objective of application security and DevSecOps is to integrate security assurances into development processes and custom line of business (LOB) applications.

Modernization

Application development is rapidly being reshaped in multiple aspects simultaneously including the DevOps team model, DevOps rapid release cadence, and the technical composition of applications via cloud services and APIs. See how the cloud is changing security relationships and responsibilities to understand these changes.

This modernization of antiquated development models presents both opportunity and a requirement to modernize security of applications and development processes. The fusion of security into DevOps processes is often referred to as DevSecOps and drives changes including:

- **Security is integrated, not outside approval:** The rapid pace of change in application development makes classic arms-length "scan and report" approaches obsolete. These legacy approaches can't keep up with releases without grinding development to a halt and creating time-to-market delays, developer underutilization, and growth of issue backlog.
  - **Shift left** to engage security earlier in application development processes as fixing issues earlier is cheaper, faster, and more effective. If you wait until after the cake is baked, it is harder to change the shape.
  - **Native integration:** Security practices must be integrated seamlessly to avoid unhealthy friction in development workflows and continuous integration/continuous deployment (CI/CD) processes. For more information about the GitHub approach, see Securing software, together.
  - **High-quality security:** Security must provide high-quality findings and guidance that enable developers to fix issues fast and don't waste developer time with false positives.
  - **Converged culture:** Security, development, and operations roles should contribute key elements into a shared culture, shared values, and shared goals and accountabilities.
- **Agile security:** Shift security from a "must be perfect to ship" approach to an agile approach that starts with minimum viable security for applications (and for the processes to develop them) that is continuously improved incrementally.
- **Embrace cloud-native** infrastructure and security features to streamline development processes while integrating security.
- **Supply chain risk management:** Take a zero-trust approach to open-source software (OSS) and third-party components that validate their integrity and ensure that bug fixes and updates are applied to these components.
- **Continuous learning:** The rapid release pace of developer services, sometimes called platform as a service (PaaS) services, and changing composition of applications means that dev, ops, and security team members will be constantly learning new technology.
- **Programmatic approach** to application security to ensure continuous improvement of the agile approach happens.

For additional context, see Microsoft secure development lifecycle.

Team composition and key relationships

Application security and DevSecOps functions are ideally performed by security aware developers and

operations teams (with the support of security subject matter experts).

This function commonly interacts with other functions and experts including:

- Security architecture and operations
- Infrastructure security
- Communications (training and tooling)
- People security
- Identity and keys
- Compliance/risk management teams
- Key business leaders or their representatives

Next steps

Review the function of data security.

Function of cloud data security

3/6/2021 • 2 minutes to read • Edit Online

The main objective for a data security team is to provide security protections and detective controls for sensitive enterprise data in any format in any location.

Modernization

Data security strategies are being shaped primarily by:

- **Data sprawl:** Sensitive data is being generated and stored on a nearly limitless variety of devices and cloud services where people creatively collaborate.
- **New model:** The cloud enables new models of "phone home for key" to supplement and replace classic data loss protection (DLP) models that "catch it on the way out the door"
- **Regulations** like General Data Protection Regulation (GDPR) are requiring organizations to closely track private data and how applications are using it.

Team composition and key relationships

Data security commonly interacts with following roles in the organization:

- Key business leaders or their representatives
- Records management teams
- Policy and standards teams
- Privacy teams
- IT architecture and operations
- Security architecture and operations
- Legal teams
- Insider risk team
- Compliance/risk management teams

Next steps

Review the function of cloud infrastructure and endpoint security.

Function of cloud infrastructure and endpoint security

3/6/2021 • 2 minutes to read • Edit Online

A cloud security team working on infrastructure and endpoint security provides security protections, detective, and response controls for infrastructure and network components used by enterprise applications and users.

Modernization

Software defined datacenters and other cloud technologies are helping solve longstanding challenges with infrastructure and endpoint security including:

- **Inventory and configuration error discovery** are much more reliable for cloud hosted assets as they're all immediately visible (vs. A physical datacenter).
- **Vulnerability management** evolving into a critical part of overall security posture management.
- **Addition of container technologies** to be managed and secured by infrastructure and network teams as the organization adopts this technology broadly. See Container security in Security Center for an example.
- **Security agent consolidation** and tool simplification to reduce the maintenance and performance overhead of security agents and tools.
- **Allow-listing of applications** and internal network filtering is becoming much easier to configure and deploy for cloud hosted servers (using machine learning generated rule sets). See Adaptive application controls and adaptive network hardening for Azure examples.
- **Automated templates** for configuring infrastructure and security are much easier with software defined datacenters in the cloud. Azure example is Azure Blueprints
- **Just in time (JIT) and just enough access (JEA)** enable practical application of least privilege principles to privileged access for servers and endpoints.
- **User experience** becomes critical as users increasingly can choose or purchase their endpoint devices.
- **Unified endpoint management** allows managing security posture of all endpoint devices including mobile and traditional PCs as well as providing critical device integrity signals for zero trust access control solutions.
- **Network security architectures** and controls are partially diminished with the shift to cloud application architectures, but they remain a fundamental security measure. For more information, see Network security and containment.

Team composition and key relationships

Cloud infrastructure and endpoint security commonly interacts with the following roles:

- IT architecture and operations
- Security architecture
- Security operations center (SOC)
- Compliance team
- Audit team

Next steps

Review the function of threat intelligence.

Function of identity and key management in the cloud

3/6/2021 • 2 minutes to read • Edit Online

The main objective of a security team working on identity management is to provide authentication and authorization of humans, services, devices, and applications. Key and certification management provides secure distribution and access to key material for cryptographic operations (which often support similar outcomes as identity management).

Modernization

Data identity and key management modernization is being shaped by:

- Identity and key/certification management disciplines are coming closer together as they both provide assurances for authentication and authorization to enable secure communications.
- Identity controls are emerging as a primary security perimeter for cloud applications
- Key-based authentication for cloud services is being replaced with identity management because of the difficulty of storing and securely providing access to those keys.
- Critical importance of carrying positive lessons learned from on-premises identity architectures such as single identity, single sign-on (SSO), and native application integration.
- Critical importance of avoiding common mistakes of on-premises architectures that often overcomplicated them, making support difficult and attacks easier. These include:
  - Sprawling groups and organizational units (OUs).
  - Sprawling set of third-party directories and identity management systems.
  - Lack of clear standardization and ownership of application identity strategy.
- Credential theft attacks remain a high impact and high likelihood threat to mitigate.
- Service accounts and application accounts remaining a top challenge, but becoming easier to solve. Identity teams should actively embrace the cloud capabilities that are beginning to solve this like Azure AD managed identities.

Team composition and key relationships

Identity and key management teams need to build strong relationships with the following roles:

- IT architecture and operations
- Security architecture and operations
- Development teams
- Data security teams
- Privacy teams
- Legal teams
- Compliance/risk management teams

Next steps

Review the function of infrastructure and endpoint security

Function of cloud threat intelligence

3/6/2021 • 2 minutes to read • Edit Online

Security threat intelligence provides context and actionable insights on active attacks and potential threats to enable decision making by security teams, technical teams, and organizational leaders.

Modernization

Threat intelligence teams are emerging and evolving to meet the needs of the security operations center (SOC) and others managing security risk for the organization.

These teams should focus on a strategy that includes:

- **Strategic threat intelligence** tailored to executive audiences increases awareness of cybersecurity risk, funding requirements, and supports sound risk decision making by organizational leadership.
- **Incremental program growth** to provide quick wins with direct incident support and evolving into a threat intelligence platform to track and inform stakeholders.
- **Tactical and operational threat intelligence** to guide decision making during incident investigation and threat detections.

Team composition and key relationships

Cloud threat intelligence is commonly provided by the following types of roles.

- Security posture management
- Organizational executive leadership
- Key business leaders or their representatives
- Security architecture and operations
- IT architecture and operations
- Risk management teams

Next steps

Review the function of cloud security posture management.

Function of cloud security posture management

5/18/2021 • 2 minutes to read • Edit Online

The main objective for a cloud security team working on posture management is to continuously report on and improve the security posture of the organization by focusing on disrupting a potential attacker's return on investment (ROI).

Modernization

Posture management is a set of new functions that realize many previously imagined or attempted ideas that were difficult, impossible, or extremely manual before the advent of the cloud. Some of elements of posture management can be traced to zero trust, deperimeterization, continuous monitoring, and manual scoring of risk by expert consultancies.

Posture management introduces a structured approach to modernization, using the following:

- **Zero trust-based access control:** That considers active threat level during access control decisions.
- **Real-time risk scoring:** To provide visibility into top risks.
- **Threat and vulnerability management (TVM)** to establish a holistic view of the organizations attack surface and risk and integrate it into operations and engineering decision making.
- **Discover sharing risks:** To understand the data exposure of enterprise intellectual property on both sanctioned and unsanctioned cloud services.
- **Cloud security posture management** to take advantage of cloud instrumentation to monitor and prioritize security improvements.
- **Technical policy:** Apply guardrails to audit and enforce the organizations standards and policies to technical systems. For more information, see Azure Policy and Azure Blueprints.
- **Threat modeling** systems and architectures, as well as specific applications.

Emerging discipline: Security posture management will disrupt many norms of the security organization in a healthy way with these new capabilities and may shift responsibilities among roles or create new roles.

Team composition and key relationships

Security posture management is an evolving function, so it might be a dedicated team, or it might be provided by other teams.

Security posture management should work closely with the following teams:

- Threat intelligence team
- Information technology
- Compliance and risk management teams
- Business leaders and SMEs
- Security architecture and operations
- Audit team

Next steps

Review the function of cloud security incident preparation.

Function of cloud security incident preparation

5/18/2021 • 2 minutes to read • Edit Online

The primary objective for an incident preparation team is to build process maturity and muscle memory for responding to major incidents throughout the organization. This includes helping prepare security, executive leadership and many outside of security.

Modernization

Practice exercises have become powerful tools to ensure stakeholders are informed and familiar with their role in a major security incident. Participants of these exercises should include:

- **Executive leadership and board of directors** to make strategic risk decisions and provide oversight.
- **Communications and public relations** to ensure internal users, customers, and other external stakeholders are informed of relevant and appropriate information.
- **Internal stakeholders** to provide legal counsel and other business advice
- **Incident management** to coordinate activities and communications.
- **Technical team members** to investigate and remediate incident.
- **Business continuity integration** with organizational functions that own crisis management, disaster recovery, and business continuity plans.

Microsoft has published lessons learned and recommendations in the Incident Response Reference Guide.

Team composition and key relationships

Critical partners for security incident preparation are:

- Security operations center (SOC).
- External counsel as needed.
- Media and communication training.
- External partners and government agencies, if applicable.

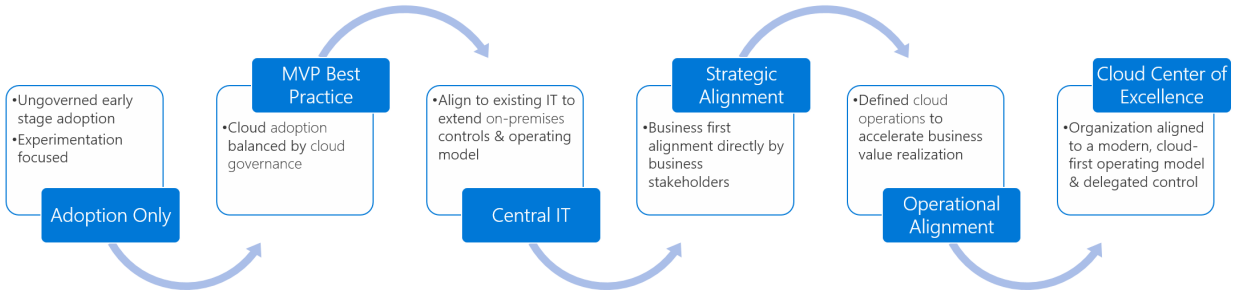
Mature team structures

3/6/2021 • 5 minutes to read • Edit Online

Every cloud function is provided by someone during every cloud adoption effort. These assignments and team structures can develop organically, or they can be intentionally designed to match a defined team structure.

As adoption needs grow, so does the need for balance and structure. Watch this video to get an overview of common team structures at various stages of organizational maturity.

The following graphic and list outline those structures based on typical maturation stages. Use these examples to find the organizational structure that best aligns with your operational needs.



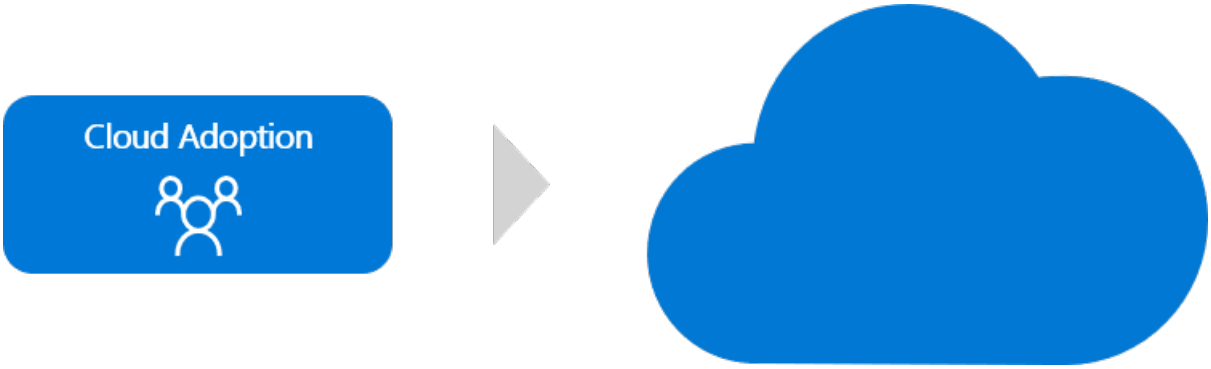
Organizational structures tend to move through the common maturity model that's outlined here:

1. Cloud adoption team only
2. MVP best practice
3. Central IT team
4. Strategic alignment
5. Operational alignment
6. Cloud center of excellence (CCoE)

Most companies start with little more than a *cloud adoption team*. But we recommend that you establish an organizational structure that more closely resembles the MVP best practice structure.

Cloud adoption team only

The nucleus of all cloud adoption efforts is the cloud adoption team. This team drives the technical changes that enable adoption. Depending on the objectives of the adoption effort, this team might include a diverse range of team members who handle a broad set of technical and business tasks.



For small-scale or early-stage adoption efforts, this team might be as small as one person. In larger-scale or

late-stage efforts, it's common to have several cloud adoption teams, each with around six engineers. Regardless of size or tasks, the consistent aspect of any cloud adoption team is that it provides the means to onboarding solutions into the cloud. For some organizations, this might be a sufficient organizational structure. The cloud adoption team article provides more insight into the structure, composition, and function of the cloud adoption team.

> **WARNING**
> Operating with **only** a cloud adoption team (or multiple cloud adoption teams) is considered an antipattern and should be avoided. At a minimum, consider the MVP best practice.

Best practice: Minimum viable product (MVP)



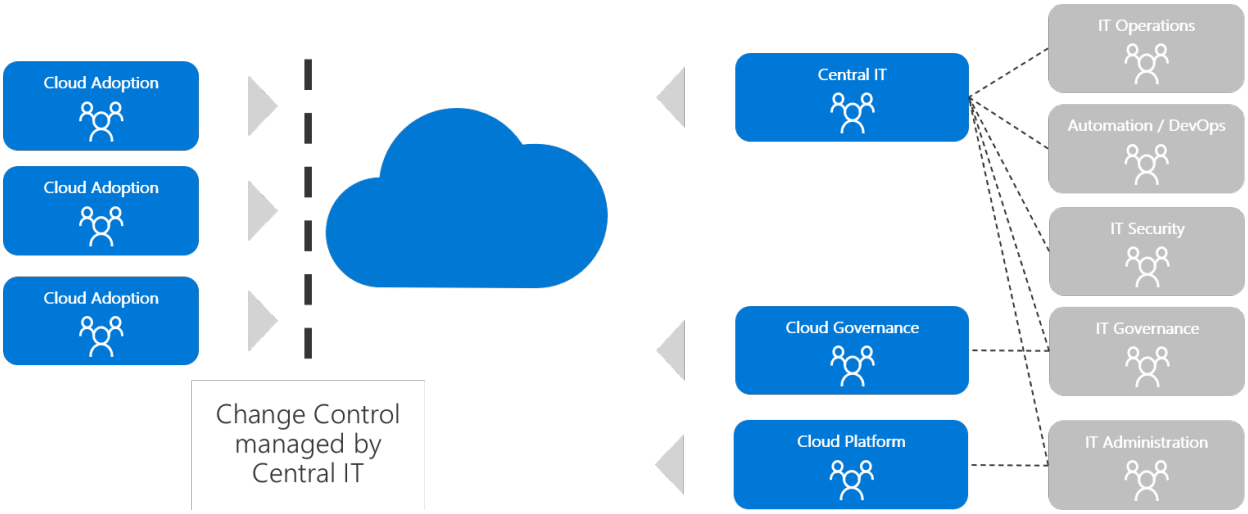
We recommend that you have two teams to create balance across cloud adoption efforts. These two teams are responsible for various functions throughout the adoption effort.

- **Cloud adoption team:** This team is accountable for technical solutions, business alignment, project management, and operations for solutions that are adopted.
- **Cloud governance team:** To balance the cloud adoption team, a cloud governance team is dedicated to ensuring excellence in the solutions that are adopted. The cloud governance team is accountable for platform maturity, platform operations, governance, and automation.

This proven approach is considered an MVP because it might not be sustainable. Each team is wearing many hats, as outlined in the *responsible, accountable, consulted, informed* (RACI) charts.

The following sections describe a fully staffed, proven organizational structure along with approaches to aligning the appropriate structure to your organization.

Central IT team



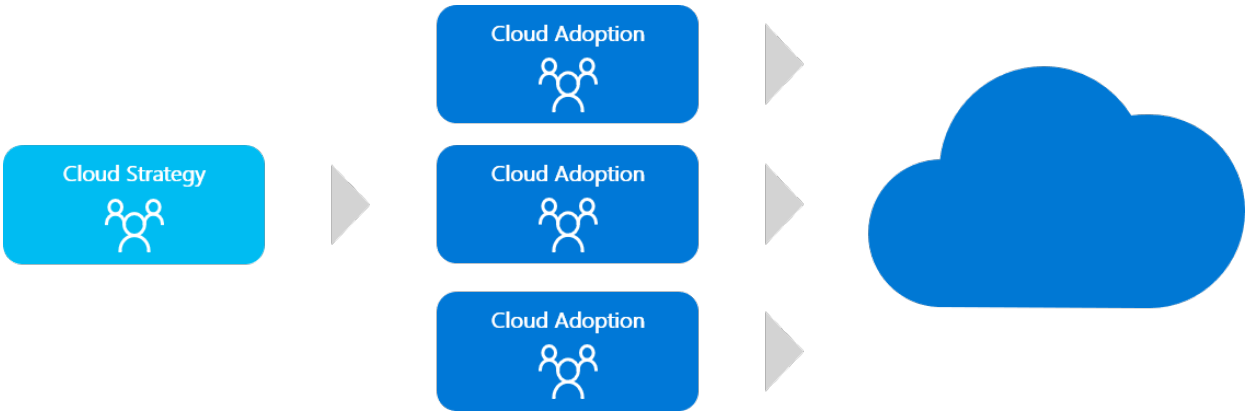
As adoption scales, the cloud governance team might struggle to keep pace with the flow of innovation from multiple cloud adoption teams. This is especially true in environments that have heavy compliance, operations, or security requirements. At this stage, it is common for companies to shift cloud responsibilities to an existing

central IT team. If that team can reassess tools, processes, and people to better support cloud adoption at scale, then including the central IT team can add significant value. Bringing in subject matter experts from operations, automation, security, and administration to modernize the central IT team can drive effective operational innovations.

Unfortunately, the central IT team phase can be one of the riskiest phases of organizational maturity. The central IT team must come to the table with a strong growth mindset. If the team views the cloud as an opportunity to grow and adapt, then it can provide great value throughout the process. But if the central IT team views cloud adoption primarily as a threat to their existing model, then the central IT team becomes an obstacle to the cloud adoption teams and the business objectives they support. Some central IT teams have spent months or even years attempting to force the cloud into alignment with on-premises approaches, with only negative results. The cloud doesn't require that everything change within the central IT team, but it does require significant change. If resistance to change is prevalent within the central IT team, this phase of maturity can quickly become a cultural antipattern.

Cloud adoption plans heavily focused on platform as a service (PaaS), DevOps, or other solutions that require less operations support are less likely to see value during this phase of maturity. On the contrary, these types of solutions are the most likely to be hindered or blocked by attempts to centralize IT. A higher level of maturity, like a cloud center of excellence (CCoE), is more likely to yield positive results for those types of transformational efforts. To understand the differences between centralized IT in the cloud and a CCoE, see Cloud center of excellence.

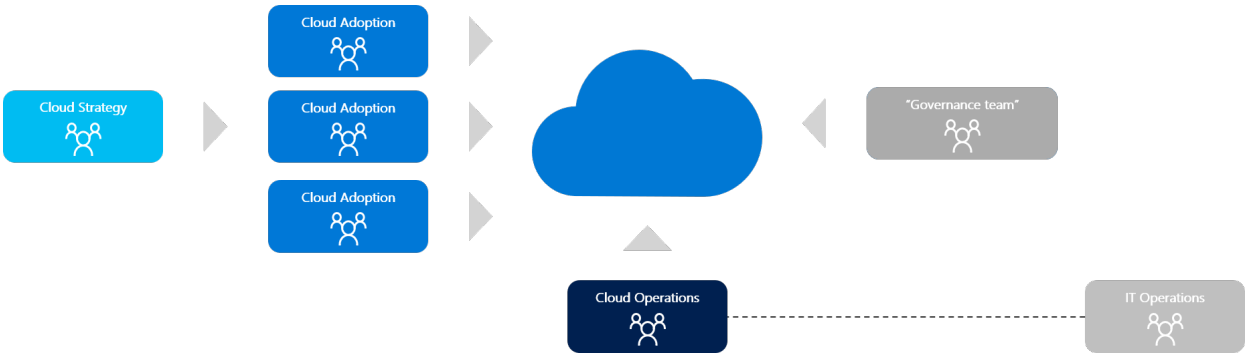
Strategic alignment



As the investment in cloud adoption grows and business values are realized, business stakeholders often become more engaged. A defined cloud strategy team, as the following image illustrates, aligns those business stakeholders to maximize the value realized by cloud adoption investments.

When maturity happens organically, as a result of IT-led cloud adoption efforts, strategic alignment is usually preceded by a governance or central IT team. When cloud adoption efforts are lead by the business, the focus on operating model and organization tends to happen earlier. Whenever possible, business outcomes and the cloud strategy team should both be defined early in the process.

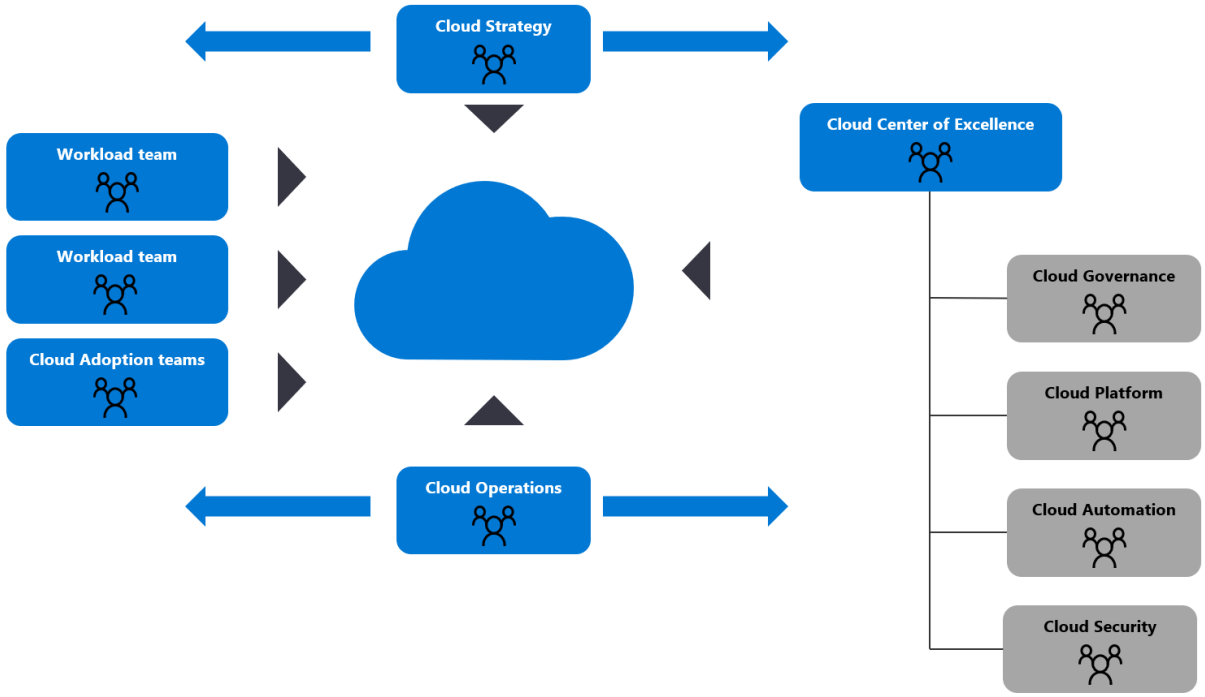
Operational alignment



Realizing business value from cloud adoption efforts requires stable operations. Operations in the cloud might require new tools, processes, or skills. When stable IT operations are required to achieve business outcomes, it's important to add a defined cloud operations team, as shown here.

Cloud operations can be delivered by the existing IT operations roles. But it's not uncommon for cloud operations to be delegated to other parties outside of IT operations. Managed service providers, DevOps teams, and business unit IT often assume the responsibilities associated with cloud operations, with support and guardrails provided by IT operations. This is increasingly common for cloud adoption efforts that focus heavily on DevOps or PaaS deployments.

Cloud center of excellence



At the highest state of maturity, a cloud center of excellence aligns teams around a modern cloud-first operating model. This approach provides centralized IT functions like governance, security, platform, and automation.

The primary difference between this structure and the central IT team structure above is a strong focus on self-service and democratization. The teams in this structure organize with the intent of delegating control as much as possible. Aligning governance and compliance practices to cloud-native solutions creates guardrails and protection mechanisms. Unlike the central IT team model, the cloud-native approach maximizes innovation and minimizes operational overhead. For this model to be adopted, mutual agreement to modernize IT processes will be required from business and IT leadership. This model is unlikely to occur organically and often requires executive support.

Next steps

After aligning to a certain stage of organizational structure maturity, you can use RACI charts to align accountability and responsibility across each team.

Align the appropriate RACI chart

Align responsibilities across teams

5/24/2021 • 3 minutes to read • Edit Online

Learn to align responsibilities across teams by developing a cross-team matrix that identifies *responsible, accountable, consulted, and informed (RACI)* parties. This article provides an example RACI matrix for the organizational structures described in Establish team structures:

- Cloud adoption team only
- MVP best practice
- Central IT team
- Strategic alignment
- Operational alignment
- Cloud center of excellence (CCoE)

To track organizational structure decisions over time, download and modify the RACI template.

The examples in this article specify these RACI constructs:

- The one team that is *accountable* for a function.
- The teams that are *responsible* for the outcomes.
- The teams that should be *consulted* during planning.
- The teams that should be *informed* when work is completed.

The last row of each table (except the first) contains a link to the most-aligned cloud capability for additional information.

Cloud adoption team only

TEAM	SOLUTION DELIVERY	BUSINESS ALIGNMENT	CHANGE MANAGEMENT	SOLUTION OPERATIONS	GOVERNANCE	PLATFORM MATURITY	PLATFORM OPERATIONS	PLATFORM AUTOMATION
Cloud adoption team	Accountable	Accountable	Accountable	Accountable	Accountable	Accountable	Accountable	Accountable

Best practice: Minimum viable product (MVP)

TEAM	SOLUTION DELIVERY	BUSINESS ALIGNMENT	CHANGE MANAGEMENT	SOLUTION OPERATIONS	GOVERNANCE	PLATFORM MATURITY	PLATFORM OPERATIONS	PLATFORM AUTOMATION
Cloud adoption team	Accountable	Accountable	Accountable	Accountable	Consulted	Consulted	Consulted	Informed
Cloud governance team	Consulted	Informed	Informed	Informed	Accountable	Accountable	Accountable	Accountable

TEAM	SOLUTION DELIVERY	BUSINESS ALIGNMENT	CHANGE MANAGEMENT	SOLUTION OPERATIONS	GOVERNANCE	PLATFORM MATURITY	PLATFORM OPERATIONS	PLATFORM AUTOMATION
Aligned cloud capability	Cloud adoption	Cloud strategy	Cloud strategy	Cloud operations	CCoE and cloud governance	CCoE and cloud platform	CCoE and cloud platform	CCoE and cloud automation

Central IT team

TEAM	SOLUTION DELIVERY	BUSINESS ALIGNMENT	CHANGE MANAGEMENT	SOLUTION OPERATIONS	GOVERNANCE	PLATFORM MATURITY	PLATFORM OPERATIONS	PLATFORM AUTOMATION
Cloud adoption team	Accountable	Accountable	Responsible	Responsible	Informed	Informed	Informed	Informed
Cloud governance team	Consulted	Informed	Informed	Informed	Accountable	Consulted	Responsible	Informed
Central IT team	Consulted	Informed	Accountable	Accountable	Responsible	Accountable	Accountable	Accountable
Aligned cloud capability	Cloud adoption	Cloud strategy	Cloud strategy	Cloud operations	Cloud governance	Central IT team	Central IT team	Central IT team

Strategic alignment

TEAM	SOLUTION DELIVERY	BUSINESS ALIGNMENT	CHANGE MANAGEMENT	SOLUTION OPERATIONS	GOVERNANCE	PLATFORM MATURITY	PLATFORM OPERATIONS	PLATFORM AUTOMATION
Cloud strategy team	Consulted	Accountable	Accountable	Consulted	Consulted	Informed	Informed	Informed
Cloud adoption team	Accountable	Consulted	Responsible	Accountable	Informed	Informed	Informed	Informed
CCoE model RACI	Consulted	Informed	Informed	Informed	Accountable	Accountable	Accountable	Accountable
Aligned cloud capability	Cloud adoption	Cloud strategy	Cloud strategy	Cloud operations	CCoE and cloud governance	CCoE and cloud platform	CCoE and cloud platform	CCoE and cloud automation

Operational alignment

TEAM	SOLUTION DELIVERY	BUSINESS ALIGNMENT	CHANGE MANAGEMENT	SOLUTION OPERATIONS	GOVERNANCE	PLATFORM MATURITY	PLATFORM OPERATIONS	PLATFORM AUTOMATION
Cloud strategy team	Consulted	Accountable	Accountable	Consulted	Consulted	Informed	Informed	Informed
Cloud adoption team	Accountable	Consulted	Responsible	Consulted	Informed	Informed	Informed	Informed
Cloud operations team	Consulted	Consulted	Responsible	Accountable	Consulted	Informed	Accountable	Consulted
CCoE model RACI	Consulted	Informed	Informed	Informed	Accountable	Accountable	Responsible	Accountable
Aligned cloud capability	Cloud adoption	Cloud strategy	Cloud strategy	Cloud operations	CCoE and cloud governance	CCoE and cloud platform	CCoE and cloud platform	CCoE and cloud automation

Cloud center of excellence (CCoE)

TEAM	SOLUTION DELIVERY	BUSINESS ALIGNMENT	CHANGE MANAGEMENT	SOLUTION OPERATIONS	GOVERNANCE	PLATFORM MATURITY	PLATFORM OPERATIONS	PLATFORM AUTOMATION
Cloud strategy team	Consulted	Accountable	Accountable	Consulted	Consulted	Informed	Informed	Informed
Cloud adoption team	Accountable	Consulted	Responsible	Consulted	Informed	Informed	Informed	Informed
Cloud operations team	Consulted	Consulted	Responsible	Accountable	Consulted	Informed	Accountable	Consulted
Cloud governance team	Consulted	Informed	Informed	Consulted	Accountable	Consulted	Responsible	Informed
Cloud platform team	Consulted	Informed	Informed	Consulted	Consulted	Accountable	Responsible	Responsible

TEAM	SOLUTION DELIVERY	BUSINESS ALIGNMENT	CHANGE MANAGEMENT	SOLUTION OPERATIONS	GOVERNANCE	PLATFORM MATURITY	PLATFORM OPERATIONS	PLATFORM AUTOMATION
Cloud automation team	Consulted	Informed	Informed	Informed	Consulted	Responsible	Responsible	Accountable
Aligned cloud capability	Cloud adoption	Cloud strategy	Cloud strategy	Cloud operations	CCoE and cloud governance	CCoE and cloud platform	CCoE and cloud platform	CCoE and cloud automation

Next steps

To track decisions about organization structure over time, download and modify the RACI template. Copy and modify the most closely aligned sample from the RACI matrices in this article.

Download the RACI template

Build technical skills

3/6/2021 • 3 minutes to read • Edit Online

Organizational and environmental (technical) readiness can require new skills for technical and nontechnical contributors. The following information can help your organization build the necessary skills.

Organizational readiness learning paths

Depending on the motivations and business outcomes that are associated with a cloud-adoption effort, leaders may need to establish new organizational structures or virtual teams to facilitate various functions. The following articles can help your organization develop the necessary skills to structure those teams to meet the desired outcomes:

- Align your organization: Discover approaches to establishing the proper organizational structures.
- Organization alignment exercises: Get an overview of alignment and team structures to help meet specific goals.
- Establish teams: Learn how to establish teams within your organization that are responsible for delivering cloud functionality.
- Break down silos and fiefdoms: Learn about two common organizational antipatterns and ways to guide the teams to productive collaboration.

Environmental (technical) readiness learning paths

During the readiness phase, technical staff have to create a migration landing zone to host, operate, and govern workloads that they migrate to the cloud. Use the following paths to accelerate development of the necessary skills:

- Create an Azure account: The first step to using Azure is to create an account. Your account holds the Azure services that you provision and handles your personal settings, like identity, billing, and preferences.
- Azure portal: Tour the Azure portal features and services, and customize the portal.
- Introduction to Azure: Get started with Azure. Create and configure your first virtual machine in the cloud.
- Introduction to security in Azure: Learn the basic concepts to protect your infrastructure and data in the cloud. Understand what responsibilities are yours and what Azure handles.
- Manage resources in Azure: Learn how to work through the Azure CLI and web portal to create, manage, and control cloud-based resources.
- Create a VM: Use the Azure portal to create a virtual machine.
- Azure network services: Learn Azure networking basics and how to improve resiliency and reduce latency.
- Azure compute options: Review the Azure compute services.
- Secure resources with Azure RBAC: Use Azure role-based access control (Azure RBAC) to secure resources.
- Azure Storage options: Learn about the benefits of Azure data storage.

During the readiness phase, architects have to design solutions that span all Azure environments. The following resources can prepare them for these tasks:

- Foundations for Cloud Architecture: A Pluralsight course to help architect the right foundational solutions.
- Microsoft Azure architecture: A Pluralsight course to ground architects in Azure architecture.
- Designing migrations for Microsoft Azure: A Pluralsight course to help architects design a migration solution.

Deeper skills exploration

The following information describes resources for additional learning.

Typical mappings of cloud IT roles

Microsoft and partners offer various options for all audiences to develop skills with Azure services.

- Map roles and skills: A resource for mapping your cloud career path. Learn about your cloud role and suggested skills. Follow a learning curriculum at your own pace to build the skills that you need most to stay relevant.
- Explore Azure certification training and exams to gain official recognition for your Azure knowledge.

Microsoft Learn

Microsoft Learn is a new approach to learning. Readiness for the new responsibilities that come with cloud adoption doesn't come easily. Microsoft Learn provides a rewarding approach to hands-on learning that helps you achieve your goals faster. Earn points, reach new levels, and achieve more.

The following are a few examples of role-specific learning paths on Microsoft Learn:

- Business users may experience a steep learning curve when they help plan, test, and adopt cloud-based technology. Microsoft Learn modules focus on adopting cloud models and tools for better managing business through cloud-based services.
- Solution architects can access hundreds of modules and learning paths. The available topics range from core infrastructure services to advanced data transformation.
- Administrators have access to modules that focus on Azure fundamentals, configuring containers, and even advanced administration in the cloud.
- Developers can use Microsoft Learn resources to help during architecture, governance, modernization activities.

Learn more

For additional learning paths, browse the Microsoft Learn catalog. Use the roles filter to align learning paths with your role.

Build a cost-conscious organization

3/6/2021 • 6 minutes to read • Edit Online

As outlined in Motivations: why are we moving to the cloud?, there are many sound reasons for a company to adopt the cloud. When cost reduction is a primary driver, it's important to create a cost-conscious organization.

Ensuring cost consciousness is not a one-time activity. Like other cloud-adoption topics, it's iterative. The following diagram outlines this process to focus on three interdependent activities: *visibility*, *accountability*, and *optimization*. These processes play out at macro and micro levels, which we describe in detail in this article.



Figure 1: Outline of the cost-conscious organization.

General cost-conscious processes

- **Visibility:** For an organization to be conscious of costs, it needs visibility into those costs. Visibility in a cost-conscious organization requires consistent reporting for the teams adopting the cloud, finance teams who manage budgets, and management teams who are responsible for the costs. This visibility is accomplished by establishing:
  - The right reporting scope.
  - Proper resource organization (management groups, resource groups, subscriptions).
  - Clear tagging strategies.
  - Proper access controls (Azure RBAC).
- **Accountability:** Accountability is as important as visibility. Accountability starts with clear budgets for adoption efforts. Budgets should be well established, clearly communicated, and based on realistic expectations. Accountability requires an iterative process and a growth mindset to drive the right level of accountability.
- **Optimization:** Optimization is the action that creates cost reductions. During optimization, resource allocations are modified to reduce the cost of supporting various workloads. This process requires iteration and experimentation. Each reduction in cost reduces performance. Finding the right balance between cost control and end-user performance expectations demands input from multiple parties.

The following sections describe the roles that various teams play in developing a cost-conscious organization.

Cloud strategy team

Building cost consciousness into cloud-adoption efforts starts at the leadership level. To be effective in the long term, the cloud strategy team should include a member of the finance team. If your financial structure holds business managers accountable for solution costs, they should be invited to join the team as well. In addition to the core activities that are typically assigned to the cloud strategy team, all members of the cloud strategy team should also be responsible for:

- **Visibility:** The cloud strategy team and cloud governance team need to know the actual costs of the cloud-adoption efforts. Given the executive-level view of this team, they should have access to multiple cost scopes to analyze spending decisions. Typically, an executive needs visibility into the total costs across all cloud "spend." But as active members of the cloud strategy team, they should also be able to view costs per business unit or per billing unit to validate showback, chargeback, or other cloud accounting models.
- **Accountability:** Budgets should be established between the cloud strategy, cloud governance, and cloud adoption teams based on expected adoption activities. When deviations from budget occur, the cloud strategy team and the cloud governance team must partner to quickly determine the best course of action to remediate the deviations.
- **Optimization:** During optimization efforts, the cloud strategy team can represent the investment and return value of specific workloads. If a workload has strategic value or financial impact on the business, cost-optimization efforts should be monitored closely. If there's no strategic impact on the organization and no inherent cost for poor performance of a workload, the cloud strategy team may approve over-optimization. To drive these decisions, the team must be able to view costs on a per-project scope.

Cloud adoption team

The cloud adoption team is at the center of all adoption activities. So, they're the first line of defense against overspending. This team has an active role in all three phases of cost-consciousness.

- **Visibility:**
  - **Awareness:** It's important for the cloud adoption team to have visibility into the cost-saving goals of the effort. Simply stating that the cloud-adoption effort will help reduce costs is a recipe for failure. *Specific* visibility is important. For example, if the goal is to reduce datacenter TCO by 3 percent or annual operating expenses by 7 percent, disclose those targets early and clearly.
  - **Telemetry:** This team needs visibility into the impact of their decisions. During migration or innovation activities, their decisions have a direct effect on costs and performance. The team needs to balance these two competing factors. Performance monitoring and cost monitoring that's scoped to the team's active projects are important to provide the necessary visibility.
- **Accountability:** The cloud adoption team needs to be aware of any preset budgets that are associated with their adoption efforts. When real costs don't align with the budget, there's an opportunity to create accountability. Accountability doesn't equate to penalizing the adoption team for exceeding budget, because budget excess can result from necessary performance decisions. Instead, accountability means educating the team about the goals and how their decisions affect those goals. Additionally, accountability includes providing a dialog in which the team can communicate about decisions that led to overspending. If those decisions are misaligned with the goals of the project, this effort provides a good opportunity to partner with the cloud strategy team to make better decisions.
- **Optimization:** This effort is a balancing act, as optimization of resources can reduce the performance of the workloads that they support. Sometimes anticipated or budgeted savings can't be realized for a workload because the workload doesn't perform adequately with the budgeted resources. In those cases,

the cloud adoption team has to make wise decisions and report changes to the cloud strategy team and the cloud governance team so that budgets or optimization decisions can be corrected.

Cloud governance team

Generally, the cloud governance team is responsible for cost management across the entire cloud-adoption effort. As outlined in the Cost Management discipline topic of the Cloud Adoption Framework's Govern methodology, cost management is the first of the Five Disciplines of Cloud Governance. Those articles outline a-series of deeper responsibilities for the cloud governance team.

This effort focuses on the following activities that are related to the development of a cost-conscious organization:

- **Visibility:** The cloud governance team works as a peer of the cloud strategy team to plan cloud-adoption budgets. These two teams also work together to regularly review actual expenses. The cloud governance team is responsible for ensuring consistent, reliable cost reporting and performance telemetry.
- **Accountability:** When budget deviations occur, the cloud strategy team and the cloud governance team must partner to quickly determine the best course of action to remediate the deviations. Generally, the cloud governance team will act on those decisions. Sometimes the action may be simple retraining for the affected cloud adoption team. The cloud governance team can also help optimize deployed assets, change discounting options, or even implement automated cost-control options like blocking deployment of unplanned assets.
- **Optimization:** After assets are migrated to or created in the cloud, you can employ monitoring tools to assess performance and utilization of those assets. Proper monitoring and performance data can identify assets that should be optimized. The cloud governance team is responsible for ensuring that the monitoring and cost-reporting tools are consistently deployed. They can also help the adoption teams identify opportunities to optimize based on performance and cost telemetry.

Cloud center of excellence

While not typically responsible for cost management, the CCoE can have a significant impact on cost-conscious organizations. Many foundational IT decisions affect costs at scale. When the CCoE does their part, costs can be reduced for multiple cloud-adoption efforts.

- **Visibility:** Any management group or resource group that houses core IT assets should be visible to the CCoE team. The team can use this data to farm opportunities to optimize.
- **Accountability:** While not typically accountable for cost, the CCoE can hold itself accountable for creating repeatable solutions that minimize cost and maximize performance.
- **Optimization:** Given the CCoE's visibility to multiple deployments, the team is in an ideal position to suggest optimization tips and to help adoption teams better tune assets.

Next steps

Practicing these responsibilities at each level of the business helps drive a cost-conscious organization. To begin acting on this guidance, review the organizational readiness introduction to help identify the right team structures.

Identify the right team structures

Cloud organizational antipatterns

7/15/2021 • 5 minutes to read • Edit Online

Customers often experience cloud adoption antipatterns within their organizational structure. Many factors can cause these problems:

- Toolsets
- Partners
- Engineers
- Misaligned IT departments

It's important to understand the role of these factors in a successful cloud adoption scenario.

Antipattern: Treat IT as a cost center

Many companies treat IT departments as cost centers. This approach can lead to the perception that IT doesn't add value to the company. When employees view IT as a provider rather than an enabler, they can become discouraged. It's also hard for the company to attract the right talent. Reduced motivation and long lifecycle times result. The quality of work from IT can suffer, and silos and fiefdoms can develop.

Example: Treat IT as a cost center

A corporation manages its IT department as a cost center responsible to the chief financial officer (CFO). The managing board perceives IT as a slow service provider that's one of the company's biggest cost drivers. The managing board doesn't realize that the mobility business unit is consuming most of the assets that the IT department ordered. IT purchases a datacenter for all business units to use, but the mobility business unit gets this oversized asset. The board doesn't view IT as an enabler or a partner.

Preferred outcome: View IT as an enabler

Instead of managing your IT department as a cost center, consider one of these approaches:

- Chargeback: Business units treat IT costs like operating expenses in their budgets.
- Showback or awareness-back: IT functions as an agent. In reports back to the business, IT attributes any direct costs to relevant business units.

Use the cloud as a tool to increase cost and business transparency. For instance, implement Cost Management discipline to increase cost transparency. Then you'll be more aware of the cost of different business units. You'll view the IT department as an enabler for those units.

To improve transparency, focus on visibility, accountability, and optimization when moving to the cloud. For more information, see Build a cost-conscious organization.

Antipattern: Invest in new technology without involving the business

IT departments often invest significant human and financial resources in building and deploying robust platforms and toolsets. But, sometimes IT fails to consider business units and their needs during design and development phases. This omission leads to new platforms with minimal relevance for business units. Employees are then hesitant to accept the new technology. Poor or slow adoption can result. Frustration also builds within IT when business units don't use its platforms.

Example: Set up a platform without involving business units

The IT department of a data analysis firm sets up and customizes an Azure platform without involving any

business units. While using the platform, business unit developers:

- Realize that they don't have the permissions that they need for deployment.
- Can only use a restricted number of services.
- Issue support tickets, which lengthen approval cycles.
- Begin to doubt the new platform.

In the end, some developers purchase an Azure subscription by themselves to avoid the hassle of IT rules and regulations. Shadow IT appears. Since the firm has little control over the shadow IT, high security risks emerge.

Preferred outcome: Involve business units in decision making

Avoid creating IT silos when deploying an enterprise-ready cloud platform. Involve developers and technical decision makers (TDMs) from business units in design and development processes. To improve platform adoption, listen to business unit input.

Refer to Start with Cloud Adoption Framework enterprise-scale landing zones for Azure best practices and design principles that increase adoption speed and are tailored toward developers. Strike the right balance between compliance and flexibility. For instance, find ways to satisfy governance and security policies while keeping development environments agile.

Antipattern: Outsource core business functions

Consulting partners and managed service providers (MSPs) can play an important role in a cloud journey. But, companies should take care that the partners' and MSPs' work doesn't provide the most value in their business. Companies that outsource responsibilities to MSPs or cloud consultants shouldn't become dependent on these providers.

Example: Outsource cloud adoption and migration

A research institute has a time-critical cloud migration project. To shorten the cloud adoption journey, it hires an MSP to build up the Azure foundation and implement the migration. Instead of learning about the cloud adoption phase and building up skills, the institute chooses to hand over all Azure responsibility to the MSP. Since the institute has no cloud or Azure knowledge, the MSP takes the lead on all decisions, making the institute dependent on the MSP.

Preferred outcome: Make critical design areas the company's responsibility

Keep outsourcing in mind as a good cost-cutting strategy. But, make decisions within your company when they involve these critical design areas:

- Governance
- Risk
- Compliance
- Identity

Keep responsibility inside the company for these and other areas that are critical to your security estate. Use external partners to speed up the adoption journey. But, to avoid becoming dependent on providers, don't outsource everything.

Antipattern: Hire technical decision makers instead of developing cloud engineers

Companies place importance on finding the right personnel. As a result, they often hire or build up TDMs during initial cloud adoption phases. Successful cloud journeys rely on TDMs. But more importantly, cloud adoptions need engineers with all-hands-on-deck mentalities and deep technical skills.

Example: Hire TDMs only

A research institute hires several TDMs to lead its cloud journey. After the initial high-level-concept phase ends, the implementation phase starts. The institute then realizes that cloud deployments behave differently than on-premises deployments. It needs extra cloud engineering effort to properly implement infrastructure as code (IaC) concepts and policy-driven governance.

Preferred outcome: Use cloud engineers for the implementation phase

Remember that engineers are essential for properly implementing cloud automation and landing zone concepts. Responsibilities and tasks can shift significantly when you adopt service models. By shifting responsibilities to a cloud provider, you can go into production faster. You can also use TDMs for decision making, but use capable cloud engineers for tasks that require deep engineering knowledge. Then you'll realize the advantages that the cloud provides.

Next steps

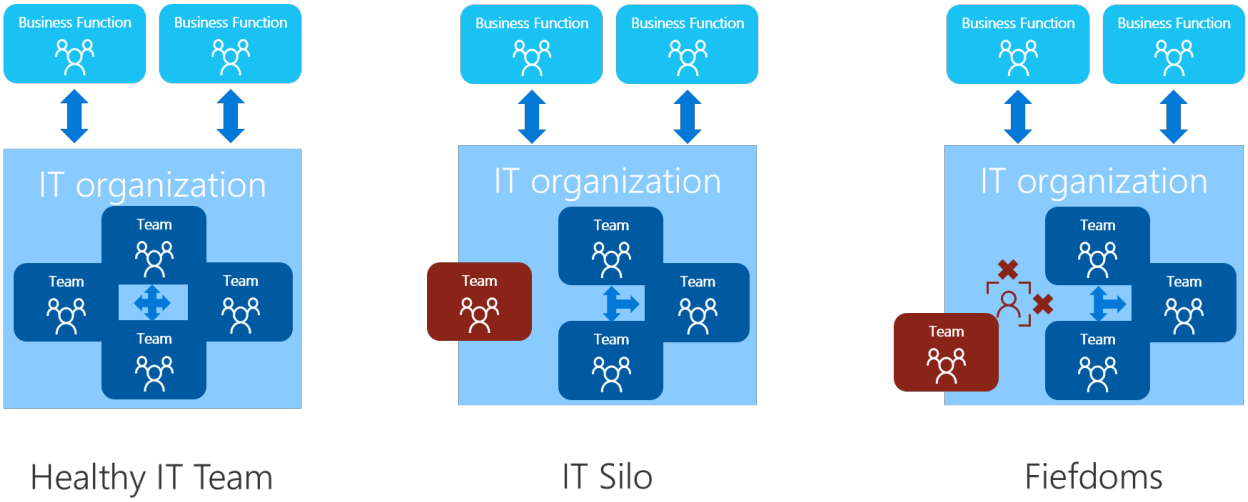
- Align responsibilities across teams
- Organizational antipatterns: silos and fiefdoms
- Build a cost-conscious organization

Organizational antipatterns: Silos and fiefdoms

3/6/2021 • 13 minutes to read • Edit Online

Success in any major change to business practices, culture, or technology operations requires a growth mindset. At the heart of the growth mindset is an acceptance of change and the ability to lead in spite of ambiguity.

Some antipatterns can block a growth mindset in organizations that want to grow and transform, including micromanagement, biased thinking, and exclusionary practices. Many of these blockers are personal challenges that create personal growth opportunities for everyone. But two common antipatterns in IT require more than individual growth or maturity: silos and fiefdoms.



Healthy IT Team IT Silo Fiefdoms

These antipatterns are a result of organic changes within various teams, which result in unhealthy organizational behaviors. To address the resistance caused by each antipattern, it's important to understand the root cause of this formation.

Healthy, organic IT teams

It's natural to create a division of labor across IT. It's healthy to establish teams that have similar expertise, shared processes, a common objective, and an aligned vision. It's also natural for those teams to have their own microculture, shared norms, and perspectives.

Healthy IT teams focus on partnering with other teams to promote the successful completion of their duties. Healthy IT teams seek to understand the business goals that their technology contribution is designed to support. The details and fiscal impact might be fuzzy, but the team's value contribution is often understood within the team.

Although healthy IT teams have a passion for the technology that they support, they're open to change and willing to try new things. Such teams are often the earliest and strongest contributors to cloud center of excellence (CCoE) efforts. Their contribution should be heavily encouraged.

Natural resistance to change

At times, the microcultures within healthy IT teams might react poorly to executive or top-down decisions to drive change. This reaction is natural, as collectives of human beings with shared norms often cooperate to overcome external threats.

Changes that affect the team's day-to-day jobs, sense of security, or autonomy can be viewed as a risk to the collective. Signs of resistance are often an early indicator that the team members don't feel like they're part of the decision-making process.

When cloud architects and other leaders invest in abolishing personal biases and driving for inclusion of the existing IT teams, this resistance is likely to lessen quickly and dissolve over time. One tool available to cloud architects and leaders to create inclusive decision making is the formation of a CCoE.

Healthy friction

It's easy to confuse resistance with friction. Existing IT teams can be knowledgeable regarding past mistakes, tangible risks, tribal knowledge about solutions, and undocumented technical debt. Unfortunately, even the healthiest IT teams can fall in the trap of describing these important data points as part of a specific technical solution, which shouldn't be changed. This approach to communication masks the teams' knowledge, creating a perception of resistance.

Providing these teams with a mechanism for communicating in future-looking terminology will add data points, identify gaps, and create healthy friction around the proposed solutions. That extra friction will sand down rough edges on the solutions and drive longer-term values. Simply changing the conversation can create clarity around complex topics and generate energy to deliver more successful solutions.

The guidance on defining corporate policy is aimed at facilitating risk-based conversations with business stakeholders. But this same model can be used to facilitate conversations with teams that are perceived as cloud resistant. When the perception of resistance is widespread, it might be wise to include resistance resolution practices in the charter for a cloud governance team.

Antipatterns

The organic and responsive growth within IT that creates healthy IT teams can also result in antipatterns that block transformation and cloud adoption. IT silos and fiefdoms are different from the natural microcultures within healthy IT teams. In either pattern, the team focus tends to be directed toward protecting their "turf". When team members are confronted with an opportunity to drive change and improve operations, they will invest more time and energy into blocking the change than finding a positive solution.

As mentioned earlier, healthy IT teams can create natural resistance and positive friction. Silos and fiefdoms are a different challenge. There is no documented leading indicator for either antipattern. These antipatterns tend to be identified after months of cloud center of excellence and cloud governance team efforts. They're discovered as the result of ongoing resistance.

Even in toxic cultures, the efforts of the CCoE and the cloud governance team should help drive cultural growth and technical progress. After months of effort, a few teams might still show no signs of inclusive behaviors and stand firm in their resistance to change. These teams are likely operating in one of the following antipattern models: silos and fiefdoms. Although these models have similar symptoms, the root cause and approaches to addressing resistance is radically different between them.

IT silos

Team members in an IT silo are likely to define themselves through their alignment to a small number of IT vendors or an area of technical specialization. But don't confuse silos with IT fiefdoms. Silos tend to be driven by comfort and passion, and silos are often easier to overcome than the fear-driven motives behind fiefdoms.

This antipattern often emerges from of a common passion for a specific solution. IT silos are then reinforced by the team's advanced skills as a result of the investment in that specific solution. This superior skill can be an accelerator to cloud adoption efforts if the resistance to change can be overcome. It can also become a major blocker if the silos are broken down or if the team members can't accurately evaluate options. Fortunately, IT silos can often be overcome without any significant changes to the organizational chart.

Address resistance from IT silos

IT silos can be addressed through the following approaches. The best approach will depend on the root cause of

the resistance.

Create virtual teams: The Organizational readiness section of the Cloud Adoption Framework describes a multilayered structure for integrating and defining four virtual teams. One benefit of this structure is cross-organization visibility and inclusion. Introducing a cloud center of excellence creates a high-profile aspirational team that top engineers will want to participate in. This helps create new cross-solution alignments that aren't bound by organizational-chart constraints, and will drive inclusion of top engineers who have been sheltered by IT silos.

Introduction of a cloud strategy team will create immediate visibility to IT contributions regarding cloud adoption efforts. When IT silos fight for separation, this visibility can help motivate IT and business leaders to properly support those resistant team members. This process is a quick path to stakeholder engagement and support.

Consider experimentation and exposure: Team members in an IT silo have likely been constrained to think a certain way for some time. Breaking the one-track mind is a first step to addressing resistance.

Experimentation and exposure are powerful tools for breaking down barriers in silos. The team members might be resistant to competing solutions, so it's not wise to put them in charge of an experiment that competes with their existing solution. But as part of a first workload test of the cloud, the organization should implement competing solutions. The siloed team should be invited to participate as an input and review source, but not as a decision maker. This should be clearly communicated to the team, along with a commitment to engage the team more deeply as a decision maker before moving into production solutions.

During review of the competing solution, use the practices outlined in Define corporate policy to document tangible risks of the experiment and establish policies that help the siloed team become more comfortable with the future state. This will expose the team to new solutions and harden the future solution.

Be "boundary-less": The teams that drive cloud adoption find it easy to push boundaries by exploring exciting, new cloud-native solutions. This is one half of the approach to removing boundaries. But that thinking can further reinforce IT silos. Pushing for change too quickly and without respect to existing cultures can create unhealthy friction and lead to natural resistance.

When IT silos start to resist, it's important to be "boundary-less" in your own solutions. Be mindful of one simple truth: cloud-native isn't always the best solution. Consider hybrid solutions that might provide an opportunity to extend the existing investments of the IT silo into the future.

Also consider cloud-based versions of the solution that the IT silo team uses now. Experiment with those solutions and expose yourself to the viewpoint of those living in the IT silo. At a minimum, you will gain a fresh perspective. In many situations, you might earn enough of the IT silo's respect to lessen resistance.

Invest in education: Many people living in an IT silo became passionate about the current solution as a result of expanding their own education. Investing in the education of these teams is seldom misplaced. Allocate time for these individuals to engage in self-learning, classes, or even conferences to break the day-to-day focus on the current solution.

For education to be an investment, some return must come as a result of the expense. In exchange for the investment, the team might demonstrate the proposed solution to the rest of the teams involved in cloud adoption. They might also provide documentation of the tangible risks, risk management approaches, and desired policies in adopting the proposed solution. Each will engage these teams in the solution and help take advantage of their tribal knowledge.

Turn roadblocks into speed bumps: IT silos can slow or stop any transformation. Experimentation and iteration will find a way, but only if the project keeps moving. Focus on turning roadblocks into merely speed bumps. Define policies that everyone can be temporarily comfortable with in exchange for continued progression.

For instance, if IT security is the roadblock because its security solution can't monitor compromises of protected data in the cloud, establish data classification policies. Prevent deployment of classified data into the cloud until an agreeable solution can be found. Invite IT security into experimentation with hybrid or cloud-native solutions to monitor protected data.

If the network team operates as a silo, identify workloads that are self-contained and don't have network dependencies. In parallel, experiment, expose, and educate the network team while working on hybrid or alternative solutions.

Be patient and be inclusive: It's tempting to move on without support of an IT silo. But this decision will cause disruptions and roadblocks down the road. Changing minds in members of the IT silo can take time. Be patient of their natural resistance--convert it to value. Be inclusive and invite healthy friction to improve the future solution.

Never compete: The IT silo exists for a reason. It persists for a reason. There is an investment in maintaining the solution that the team members are passionate about. Directly competing with the solution or the IT silo will distract from the real goal of achieving business outcomes. This trap has blocked many transformation projects.

Stay focused on the goal, as opposed to a single component of the goal. Help accentuate the positive aspects of the IT silo's solution and help the team members make wise decisions about the best solutions for the future. Don't insult or degrade the current solution, because that would be counterproductive.

Partner with the business: If the IT silo isn't blocking business outcomes, why do you care? There is no perfect solution or perfect IT vendor. Competition exists for a reason; each has its own benefits.

Embrace diversity and include the business by supporting and aligning to a strong cloud strategy team. When an IT silo supports a solution that blocks business outcomes, it will be easier to communicate that roadblock without the noise of technical squabbles. Supporting nonblocking IT silos will show an ability to partner for the desired business outcomes. These efforts will earn more respect and greater support from the business when an IT silo presents a legitimate blocker.

IT fiefdoms

Team members in an IT fiefdom are likely to define themselves through their alignment to a specific process or area of responsibility. The team operates under an assumption that external influence on its area of responsibility will lead to problems. Fiefdoms tend to be a fear-driven antipattern, which will require significant leadership support to overcome.

Fiefdoms are especially common in organizations that have experienced IT downsizing, frequent turbulence in IT staff, or poor IT leadership. When the business sees IT purely as a cost center, fiefdoms are much more likely to arise.

Generally, fiefdoms are the result of a line manager who fears loss of the team and the associated power base. These leaders often have a sense of duties to their team and feel a need to protect their subordinates from negative consequences. Phrases like "shelter the team from change" and "protect the team from process disruption" can be indicators of an overly guarded manager who might need more support from leadership.

Address resistance from IT fiefdoms

IT fiefdoms can demonstrate some growth by following the approaches to addressing IT silo resistance. Before you try to address resistance from an IT fiefdom, we recommend that you treat the team like an IT silo first. If those types of approaches fail to yield any significant change, the resistant team might be suffering from an IT fiefdom antipattern. The root cause of IT fiefdoms is a little more complex to address, because that resistance tends to come from the direct line manager (or a leader higher up the organizational chart). Challenges that are IT silo-driven are typically simpler to overcome.

When continued resistance from IT fiefdoms blocks cloud adoption efforts, it might be wise for a combined

effort to evaluate the situation with existing IT leaders. IT leaders should carefully consider insights from the cloud strategy team, cloud center of excellence, and the cloud governance team before making decisions.

> **NOTE**
>
> IT leaders should never take changes to the organizational chart lightly. They should also validate and analyze feedback from each of the supporting teams. But transformational efforts like cloud adoption tend to magnify underlying issues that have gone unnoticed or unaddressed long before this effort. When fiefdoms are preventing the company's success, leadership changes are a likely necessity.
>
> Fortunately, removing the leader of a fiefdom doesn't often end in termination. These strong, passionate leaders can often move into a management role after a brief period of reflection. With the right support, this change can be healthy for the leader of the fiefdom and the current team.

Caution

For managers of IT fiefdoms, protecting the team from risk is a clear leadership value. But there's a fine line between protection and isolation. When the team is blocked from participating in driving changes, it can have psychological and professional consequences on the team. The urge to resist change might be strong, especially during times of visible change.

The manager of any isolated team can best demonstrate a growth mindset by experimenting with the guidance associated with healthy IT teams in the preceding sections. Active and optimistic participation in governance and CCoE activities can lead to personal growth. Managers of IT fiefdoms are best positioned to change stifling mindsets and help the team develop new ideas.

IT fiefdoms can be a sign of systemic leadership issues. To overcome an IT fiefdom, IT leaders need the ability to make changes to operations, responsibilities, and occasionally even the people who provide line management of specific teams. When those changes are required, it's wise to approach those changes with clear and defensible data points.

Alignment with business stakeholders, business motivations, and business outcomes might be required to drive the necessary change. Partnership with the cloud strategy team, cloud center of excellence, and the cloud governance team can provide the data points needed for a defensible position. When necessary, these teams should be involved in a group escalation to address challenges that can't be addressed with IT leadership alone.

Next steps

Disrupting organizational antipatterns is a team effort. To act on this guidance, review the organizational readiness introduction to identify the right team structures and participants:

Identify the right team structures and participants

Tools and templates

9/1/2021 • 4 minutes to read • Edit Online

The Cloud Adoption Framework includes tools that help you quickly implement technical change. Use these tools, templates, and assessments to accelerate cloud adoption. The following resources can help you in each phase of adoption. Some of the tools and templates can be used in multiple phases.

Strategy

RESOURCE	DESCRIPTION
Cloud journey tracker	Identify your cloud adoption path based on the needs of your business.
Strategy and plan template	Document decisions as you execute your cloud adoption strategy and plan.

Plan

RESOURCE	DESCRIPTION
Strategic migration assessment and readiness tool	From business planning to training to security and governance, find out what you need to do to prepare for your Microsoft Azure migration with the strategic migration assessment and readiness tool (SMART).
Cloud journey tracker	Identify your cloud adoption path based on the needs of your business.
Strategy and plan template	Document decisions, as you execute your cloud adoption strategy and plan.
Cloud adoption plan generator	Standardize processes by deploying a backlog to Azure Boards using a template.
Using the `Strategy-Plan-Ready-Govern` Azure DevOps template	Standardize processes by deploying a backlog to Azure Boards using a template.

Ready

RESOURCE	DESCRIPTION
Readiness checklist	Use this checklist to prepare your environment for adoption. This preparation includes preparing your first migration landing zone, personalizing the blueprint, and expanding it.
Naming and tagging conventions tracking template	Document decisions about naming and tagging standards to ensure consistency and reduce onboarding time.

RESOURCE	DESCRIPTION
CAF Foundation blueprint	Use a lightweight implementation of an initial governance foundation to provide practical experience with governance tools in Azure.
CAF Migration landing zone blueprint	Provision and prepare to host workloads being migrated from an on-premises environment into Azure. For more information about this blueprint, see Deploy a migration landing zone.
Terraform modules	Open-source code base for the Terraform version of the CAF Azure landing zones.
Terraform registry	The Terraform registry website, filtered to list all of the Cloud Adoption Framework modules needed to create a landing zone via Terraform.
Enterprise-scale landing zone	Open-source code base for the enterprise-scale implementation of the CAF Azure landing zones.
Data landing zone	Expand your landing zone with data. Data landing zone shared services, such as data storage, ingestion services, and management services.

Govern

RESOURCE	DESCRIPTION
Governance benchmark assessment	Identify gaps between your current state and business priorities, and get the right resources to help you address those gaps.
CAF Foundation blueprint	Lightweight implementation of an initial governance foundation to provide practical experience with governance tools in Azure.
Governance discipline template	Define the basic set of governance processes used to enforce each governance discipline.
Cost Management discipline template	Define the policy statements and design guidance to mature the cloud governance in your organization. Focuses on cost management.
Deployment Acceleration discipline template	Define the policy statements and design guidance to mature the cloud governance in your organization. Focuses on deployment acceleration.
Identity Baseline discipline template	Define the policy statements and design guidance to mature the cloud governance in your organization. Focuses on identity requirements.
Resource Consistency discipline template	Define the policy statements and design guidance to mature the cloud governance in your organization. Focuses on resource consistency.

RESOURCE	DESCRIPTION
Security Baseline discipline template	Define the policy statements and design guidance to mature the cloud governance in your organization. Focuses on the security baseline.
Azure Security Benchmark	The Azure Security Benchmark (ASB) provides prescriptive best practices and recommendations to help improve the security of workloads, data, and services on Azure.
Azure governance visualizer	The Azure governance visualizer is a PowerShell script that iterates through an Azure tenant's management group hierarchy down to the subscription level. It captures data from the most relevant Azure governance capabilities such as Azure Policy, Azure role-based access control (Azure RBAC), and Azure Blueprints. From the collected data, the visualizer shows your hierarchy map, creates a tenant summary, and builds granular scope insights about your management groups and subscriptions.

Migrate

RESOURCE	DESCRIPTION
Datacenter migration discovery checklist	Review this checklist for information that helps identify workloads, servers, and other assets in your datacenter. Use this information to help plan your migration.
Migration templates	In the Azure DevOps generator, we've built several templates that you can use to help streamline your projects. Templates have been created for Azure Virtual Desktop, server migration, SQL migration, and AKS deployments.

Innovate

RESOURCE	DESCRIPTION
Deployment templates	The Azure DevOps generator has project checklists to help you build your knowledge mining, modern data warehouse, and industry focused accelerators like the retail recommender using AI and modern IoT. The templates give you a structured approach and checklist to follow. Each template and work item has links to required content, assets, and code samples that can help you get started.
Solution Accelerators	The Cloud Adoption Framework Solution Accelerators are intended to help you rapidly deploy applications and workloads to address business challenges. Today we have solution accelerators for, Retail Recommendation, Demand Forecasting, Modern Finance and Workplace Intelligence for IT.

Manage

RESOURCE	DESCRIPTION
Microsoft Azure Well-Architected Review	This online assessment aids in defining workload-specific architectures and operations options.
Best practices source code	This deployable source code complements and accelerates adoption of best practices for Azure server management services. Use this source code to quickly enable operations management and establish an operations baseline.
Operations management workbook	Document decisions about operations management in the cloud, and facilitate conversations with the business to ensure alignment regarding SLAs, investment in resiliency, and budget allocation related to operations.

Organize

RESOURCE	DESCRIPTION
Cross-team responsible, accountable, consulted, and informed (RACI) diagram	Download and modify the RACI spreadsheet template to track organizational structure decisions over time.

Secure

RESOURCE	DESCRIPTION
Deploy STIG compliant Windows VM	This quickstart shows how to deploy a STIG-compliant Windows virtual machine (Preview) on Azure or Azure Government using the corresponding portal.
Deploy STIG compliant Linux VM	This quickstart shows how to deploy a STIG-compliant Linux virtual machine (Preview) on Azure or Azure Government using the corresponding portal.

Architectural decision guides

3/6/2021 • 2 minutes to read • Edit Online

The architectural decision guides in the Cloud Adoption Framework describe patterns and models that help when creating cloud governance design guidance. Each decision guide focuses on one core infrastructure component of cloud deployments and lists patterns and models that can support specific cloud deployment scenarios.

When you begin to establish cloud governance for your organization, actionable governance journeys provide a baseline roadmap. These journeys make assumptions about requirements and priorities that might not reflect those of your organization.

These decision guides supplement the sample governance journeys by providing alternative patterns and models that help you align the architectural design choices made in the example design guidance with your own requirements.

Decision guidance categories

The following categories represent foundational technologies for all cloud deployments. The sample governance journeys make design decisions related to these technologies based on the needs of example businesses, and some of these decisions might not match your organization's needs. The following sections discuss alternative options for each category, allowing you to choose a pattern or model better suited to your requirements.

Subscriptions: Plan your cloud deployment's subscription design and account structure to match your organization's ownership, billing, and management capabilities.

Identity: Integrate cloud-based identity services with your existing identity resources to support authorization and access control within your IT environment.

Policy enforcement: Define and enforce organizational policy rules for cloud-deployed resources and workloads that align with your governance requirements.

Resource consistency: Ensure that deployment and organization of your cloud-based resources align to enforce resource management and policy requirements.

Resource tagging: Organize your cloud-based resources to support billing models, cloud accounting approaches, management, and to optimize resource utilization and cost. Resource tagging requires a consistent and well-organized naming and metadata scheme.

Software Defined Networking: Deploy secure workloads to the cloud using rapid deployment and modification of virtualized networking capabilities. Software-defined networks can support agile workflows, isolate resources, and integrate cloud-based systems with your existing IT infrastructure.

Encryption: Secure your sensitive data using encryption to align with your organization's compliance and security policy requirements.

Logging and reporting: Monitor log data generated by cloud-based resources. Analyzing data provides health-related insights into the operations, maintenance, and compliance status of workloads.

Next steps

Learn how subscriptions and accounts serve as the base of a cloud deployment.

Subscriptions design

Subscription decision guide

8/25/2021 • 3 minutes to read • Edit Online

Effective subscription design helps organizations establish a structure to organize and manage assets in Azure during cloud adoption. This guide will help you decide when to create additional subscriptions and expand your management group hierarchy to support your business priorities.

Prerequisites

Adopting Azure begins by creating an Azure subscription, associating it with an account, and deploying resources like virtual machines and databases to the subscription. For an overview of these concepts, see Azure fundamental concepts.

- Create your initial subscriptions.
- Create additional subscriptions to scale your Azure environment.
- Organize and manage your subscriptions using Azure management groups.

Model your organization

Because every organization is different, Azure management groups are designed to be flexible. Modeling your cloud estate to reflect your organization's hierarchy helps you define and apply policies at higher levels of the hierarchy, and rely on inheritance to ensure that those policies are automatically applied to management groups lower in the hierarchy. Although subscriptions can be moved between different management groups, it's helpful to design an initial management group hierarchy that reflects your anticipated organizational needs.

Before finalizing your subscription design, also consider how resource consistency considerations might influence your design choices.

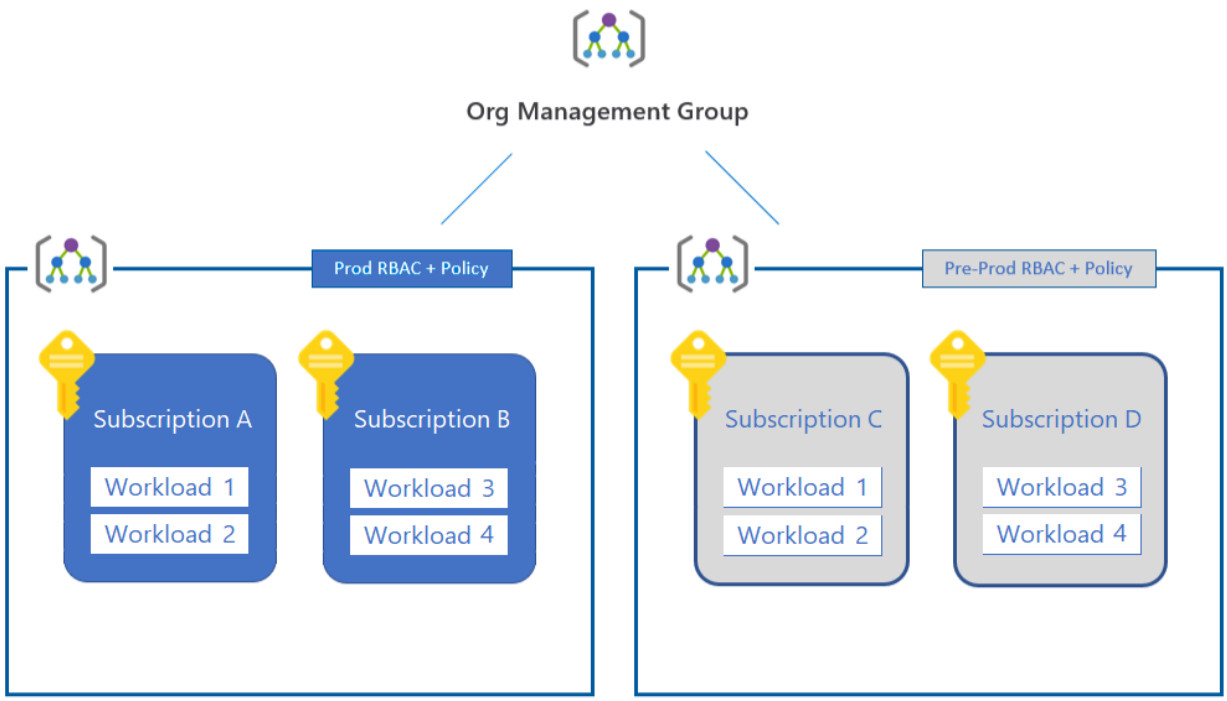
> **NOTE**
> An Azure Enterprise Agreement (EA) allows you to define another organizational hierarchy for billing purposes. This hierarchy is distinct from your management group hierarchy, which focuses on providing an inheritance model for easily applying suitable policies and access control to your resources.

Subscription design strategies

Consider the following subscription design strategies to address your business priorities.

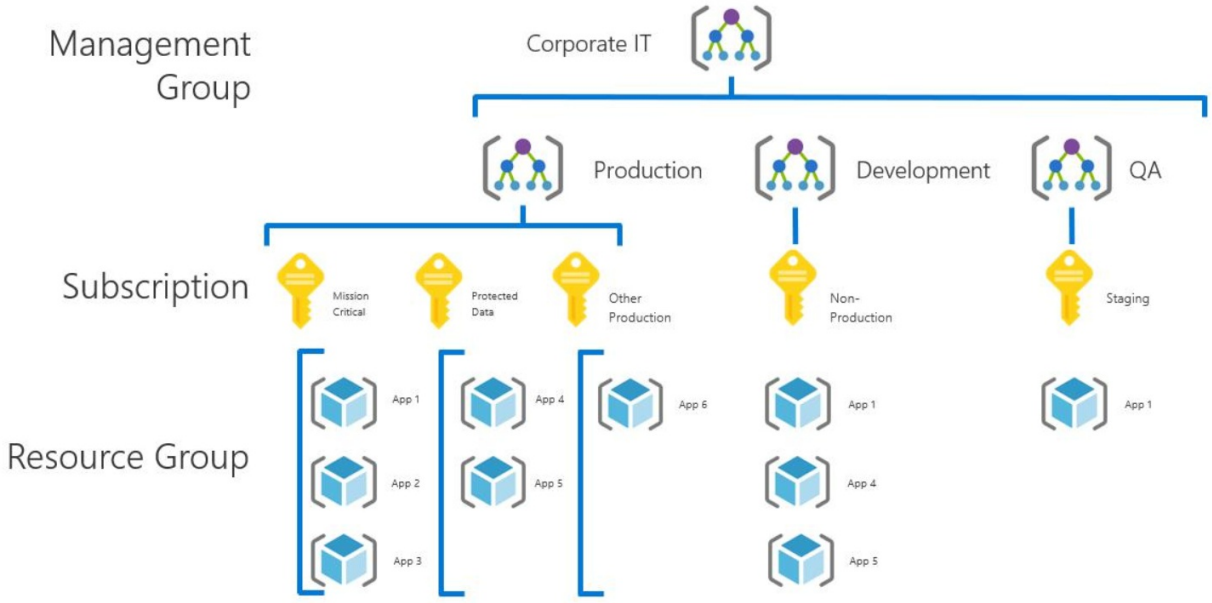
Workload separation strategy

As an organization adds new workloads to the cloud, different ownership of subscriptions or basic separation of responsibility may result in multiple subscriptions in both the production and nonproduction management groups. While this approach does provide basic workload separation, it doesn't take significant advantage of the inheritance model to automatically apply policies across a subset of your subscriptions.



Application category strategy

As an organization's cloud footprint grows, additional subscriptions are typically created to support applications with fundamental differences in business criticality, compliance requirements, access controls, or data protection needs. Building from the initial production and nonproduction subscriptions, the subscriptions supporting these application categories are organized under either the production or nonproduction management group as applicable. These subscriptions are typically owned and administered by the operations staff of a central IT team.



Each organization will categorize their applications differently, often separating subscriptions based on specific applications or services or along the lines of application archetypes. This categorization is often designed to support workloads that are likely to consume most of the resource limits of a subscription, or separate mission-critical workloads to ensure they don't compete with other workloads under these limits. Some workloads that might justify a separate subscription include:

- Mission-critical workloads.
- Applications that are part of cost of goods sold (COGS) within your company. For example, every widget manufactured by a company contains an Azure IoT module that sends telemetry. This may require a dedicated subscription for accounting or governance purposes to be part of COGS.
- Applications subject to regulatory requirements such as HIPAA or FedRAMP.

Functional strategy

The functional strategy organizes subscriptions and accounts along functional lines, such as finance, sales, or IT support, using a management group hierarchy.

Business unit strategy

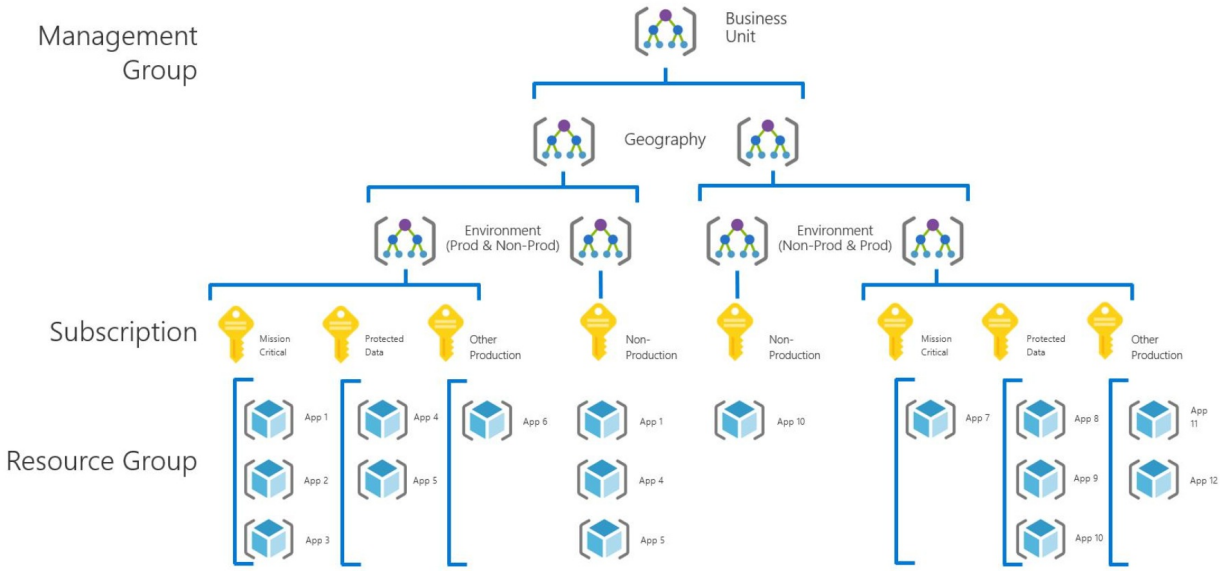
The business unit strategy groups subscriptions and accounts based on profit and loss category, business unit, division, profit center, or similar business structure using a management group hierarchy.

Geographic strategy

For organizations with global operations, the geographic strategy groups subscriptions and accounts based on geographic regions using a management group hierarchy.

Mix subscription strategies

Management group hierarchies can be up to six levels deep. This provides you with the flexibility to create a hierarchy that combines several of these strategies to meet your organizational needs. For example, the following diagram shows an organizational hierarchy that combines a business unit strategy with a geographic strategy.



Related resources

- Resource access management in Azure
- Multiple layers of governance in large enterprises
- Multiple geographic regions

Next steps

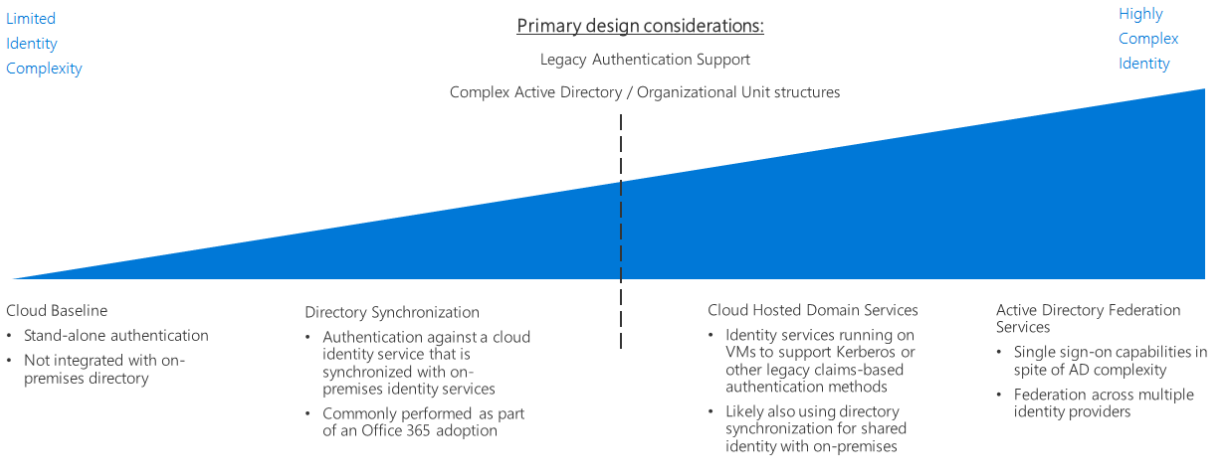
Subscription design is just one of the core infrastructure components requiring architectural decisions during a cloud adoption process. Visit the architectural decision guides overview to learn about additional strategies used when making design decisions for other types of infrastructure.

Architectural decision guides

Identity decision guide

5/18/2021 • 7 minutes to read • Edit Online

In any environment, whether on-premises, hybrid, or cloud-only, IT needs to control which administrators, users, and groups have access to resources. Identity and access management (IAM) services enable you to manage access control in the cloud.



Jump to: Determine identity integration requirements | Cloud baseline | Directory synchronization | Cloud-hosted domain services | Active Directory Federation Services | Learn more

Several options are available for managing identity in a cloud environment. These options vary in cost and complexity. A key factor in structuring your cloud-based identity services is the level of integration required with your existing on-premises identity infrastructure.

Azure Active Directory (Azure AD) provides a base level of access control and identity management for Azure resources. If your organization's on-premises Active Directory infrastructure has a complex forest structure or customized organizational units (OUs), your cloud-based workloads might require directory synchronization with Azure AD for a consistent set of identities, groups, and roles between your on-premises and cloud environments. Additionally, support for applications that depend on legacy authentication mechanisms might require the deployment of Active Directory Domain Services (AD DS) in the cloud.

Cloud-based identity management is an iterative process. You could start with a cloud-native solution with a small set of users and corresponding roles for an initial deployment. As your migration matures, you might need to integrate your identity solution using directory synchronization or add domains services as part of your cloud deployments. Revisit your identity strategy in every iteration of your migration process.

Determine identity integration requirements

QUESTION	CLOUD BASELINE	DIRECTORY SYNCHRONIZATION	CLOUD-HOSTED DOMAIN SERVICES	ACTIVE DIRECTORY FEDERATION SERVICES
Do you currently lack an on-premises directory service?	Yes	No	No	No

QUESTION	CLOUD BASELINE	DIRECTORY SYNCHRONIZATION	CLOUD-HOSTED DOMAIN SERVICES	ACTIVE DIRECTORY FEDERATION SERVICES
Do your workloads need to use a common set of users and groups between the cloud and on-premises environment?	No	Yes	No	No
Do your workloads depend on legacy authentication mechanisms, such as Kerberos or NTLM?	No	No	Yes	Yes
Do you require single sign-on across multiple identity providers?	No	No	No	Yes

As part of planning your migration to Azure, you will need to determine how best to integrate your existing identity management and cloud identity services. The following are common integration scenarios.

Cloud baseline

Azure AD is the native identity and access management (IAM) system for granting users and groups access to management features on the Azure platform. If your organization lacks a significant on-premises identity solution, and you plan to migrate workloads to be compatible with cloud-based authentication mechanisms, you should begin developing your identity infrastructure using Azure AD as a base.

Cloud baseline assumptions: Using a purely cloud-native identity infrastructure assumes the following:

- Your cloud-based resources will not have dependencies on on-premises directory services or Active Directory servers, or workloads can be modified to remove those dependencies.
- The application or service workloads being migrated either support authentication mechanisms compatible with Azure AD or can be modified easily to support them. Azure AD relies on internet-ready authentication mechanisms such as SAML, OAuth, and OpenID Connect. Existing workloads that depend on legacy authentication methods using protocols such as Kerberos or NTLM might need to be refactored before migrating to the cloud using the cloud baseline pattern.

> **TIP**
>
> Completely migrating your identity services to Azure AD eliminates the need to maintain your own identity infrastructure, significantly simplifying your IT management.
>
> But Azure AD is not a full replacement for a traditional on-premises Active Directory infrastructure. Directory features such as legacy authentication methods, computer management, or group policy might not be available without deploying additional tools or services to the cloud.
>
> For scenarios where you need to integrate your on-premises identities or domain services with your cloud deployments, see the directory synchronization and cloud-hosted domain services patterns discussed below.

Directory synchronization

For organizations with existing on-premises Active Directory infrastructure, directory synchronization is often the best solution for preserving existing user and access management while providing the required IAM

capabilities for managing cloud resources. This process continuously replicates directory information between Azure AD and on-premises directory services, allowing common credentials for users and a consistent identity, role, and permission system across your entire organization.

> **NOTE**
> Organizations that have adopted Microsoft 365 might have already implemented directory synchronization between their on-premises Active Directory infrastructure and Azure Active Directory.

Directory synchronization assumptions: Using a synchronized identity solution assumes the following:

- You need to maintain a common set of user accounts and groups across your cloud and on-premises IT infrastructure.
- Your on-premises identity services support replication with Azure AD.

> **TIP**
> Any cloud-based workloads that depend on legacy authentication mechanisms provided by on-premises Active Directory servers and that are not supported by Azure AD will still require either connectivity to on-premises domain services or virtual servers in the cloud environment providing these services. Using on-premises identity services also introduces dependencies on connectivity between the cloud and on-premises networks.

Cloud-hosted domain services

If you have workloads that depend on claims-based authentication using legacy protocols such as Kerberos or NTLM, and those workloads cannot be refactored to accept modern authentication protocols such as SAML or OAuth and OpenID Connect, you might need to migrate some of your domain services to the cloud as part of your cloud deployment.

This pattern involves deploying virtual machines running Active Directory to your cloud-based virtual networks to provide Active Directory Domain Services (AD DS) for resources in the cloud. Any existing applications and services migrating to your cloud network should be able to use these cloud-hosted directory servers with minor modifications.

It's likely that your existing directories and domain services will continue to be used in your on-premises environment. In this scenario, you should also use directory synchronization to provide a common set of users and roles in both the cloud and on-premises environments.

Cloud-hosted domain services assumptions: Performing a directory migration assumes the following:

- Your workloads depend on claims-based authentication using protocols like Kerberos or NTLM.
- Your workload virtual machines need to be domain-joined for management or application of Active Directory group policy purposes.

> **TIP**
> While a directory migration coupled with cloud-hosted domain services provides great flexibility when migrating existing workloads, hosting virtual machines within your cloud virtual network to provide these services does increase the complexity of your IT management tasks. As your cloud migration experience matures, examine the long-term maintenance requirements of hosting these servers. Consider whether refactoring existing workloads for compatibility with cloud identity providers such as Azure Active Directory can reduce the need for these cloud-hosted servers.

Active Directory Federation Services

Identity federation establishes trust relationships across multiple identity management systems to allow common authentication and authorization capabilities. You can then support single sign-on capabilities across

multiple domains within your organization or identity systems managed by your customers or business partners.

Azure AD supports federation of on-premises Active Directory domains using Active Directory Federation Services (AD FS). For more information about how this can be implemented in Azure, see Extend AD FS to Azure.

Learn more

For more information about identity services in Azure, see:

- Azure AD. Azure AD provides cloud-based identity services. It allows you to manage access to your Azure resources and control identity management, device registration, user provisioning, application access control, and data protection.
- Azure AD Connect. The Azure AD Connect tool allows you to connect Azure AD instances with your existing identity management solutions, allowing synchronization of your existing directory in the cloud.
- Azure role-based access control (Azure RBAC). Azure RBAC efficiently and securely manages access to resources in the management plane. Jobs and responsibilities are organized into roles, and users are assigned to these roles. Azure RBAC allows you to control who has access to a resource along with which actions a user can perform on that resource.
- Azure AD Privileged Identity Management (PIM). PIM lowers the exposure time of resource access privileges and increases your visibility into their use through reports and alerts. It limits users to taking on their privileges "just in time" (JIT), or by assigning privileges for a shorter duration, after which privileges are revoked automatically.
- Integrate on-premises Active Directory domains with Azure Active Directory. This reference architecture provides an example of directory synchronization between on-premises Active Directory domains and Azure AD.
- Extend Active Directory Domain Services (AD DS) to Azure. This reference architecture provides an example of deploying AD DS servers to extend domain services to cloud-based resources.
- Extend Active Directory Federation Services (AD FS) to Azure. This reference architecture configures Active Directory Federation Services (AD FS) to perform federated authentication and authorization with your Azure AD directory.

Next steps

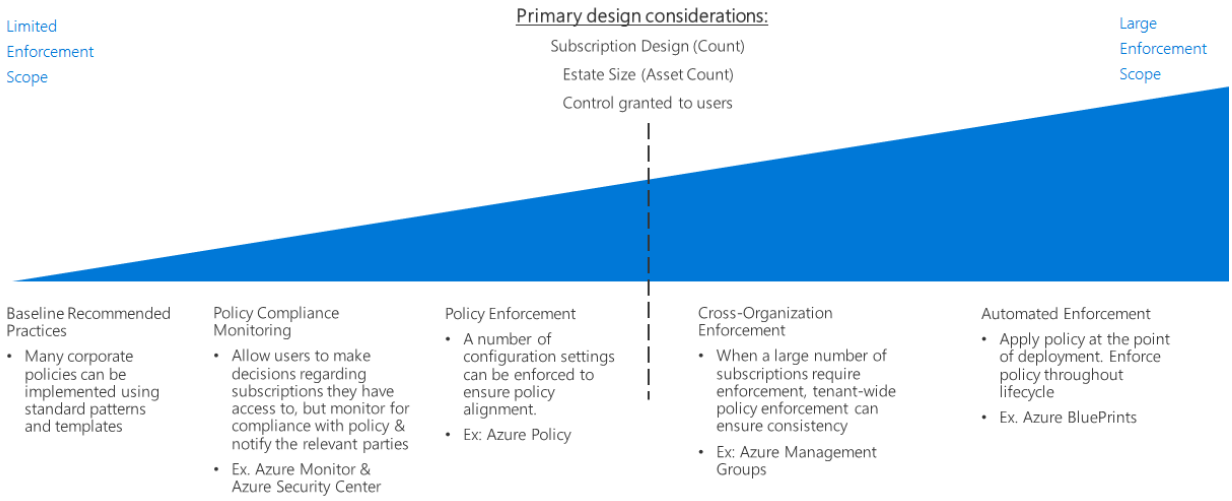
Identity is just one of the core infrastructure components requiring architectural decisions during a cloud adoption process. To learn about alternative patterns or models used when making design decisions for other types of infrastructure, see the architectural decision guides overview.

Architectural decision guides overview

Policy enforcement decision guide

3/6/2021 • 3 minutes to read • Edit Online

Defining organizational policy is not effective unless it can be enforced across your organization. A key aspect of planning any cloud migration is determining how best to combine tools provided by the cloud platform with your existing IT processes to maximize policy compliance across your entire cloud estate.



Jump to: Baseline best practices | Policy compliance monitoring | Policy enforcement | Cross-organization policy | Automated enforcement

As your cloud estate grows, you will be faced with a corresponding need to maintain and enforce policy across a larger array of resources, and subscriptions. As your estate gets larger and your organization's policy requirements increase, the scope of your policy enforcement processes needs to expand to ensure consistent policy adherence and fast violation detection.

Platform-provided policy enforcement mechanisms at the resource or subscription level are usually sufficient for smaller cloud estates. Larger deployments justify a larger enforcement scope and may need to take advantage of more sophisticated enforcement mechanisms involving deployment standards, resource grouping and organization, and integrating policy enforcement with your logging and reporting systems.

The primary factors in determining the scope of your policy enforcement processes is your organization's cloud governance requirements, the size and nature of your cloud estate, and how your organization is reflected in your subscription design. An increase in size of your estate or a greater need to centrally manage policy enforcement can both justify an increase in enforcement scope.

Baseline best practices

For single subscription and simple cloud deployments, many corporate policies can be enforced using features that are native to resources and subscriptions in Azure. The consistent use of the patterns discussed throughout the Cloud Adoption Framework decision guides can help establish a baseline level of policy compliance without specific investment in policy enforcement. These features include:

- Deployment templates can provision resources with standardized structure and configuration.
- Tagging and naming standards can help organize operations and support accounting and business requirements.
- Traffic management and networking restrictions can be implemented through Software Defined Networking.
- Azure role-based access control can secure and isolate your cloud resources.

Start your cloud policy enforcement planning by examining how the application of the standard patterns discussed throughout these guides can help meet your organizational requirements.

Policy compliance monitoring

A first step beyond simply relying on the policy enforcement mechanisms provided by the Azure platform is ensuring ability to verify cloud-based applications and services comply with organizational policy. This includes implementing notification capabilities for alerting responsible parties if a resource becomes noncompliant. Effective logging and reporting of the compliance status of your cloud workloads is a critical part of a corporate policy enforcement strategy.

As your cloud estate grows, additional tools such as Azure Security Center can provide integrated security and threat detection, and help apply centralized policy management and alerting for both your on-premises and cloud assets.

Policy enforcement

In Azure, you can apply configuration settings and resource creation rules at the management group, subscription, or resource group level to help ensure policy alignment.

Azure Policy is an Azure service for creating, assigning, and managing policies. These policies enforce different rules and effects over your resources, so those resources stay compliant with your corporate standards and service-level agreements. Azure Policy evaluates your resources for noncompliance with assigned policies. For example, you might want to limit the SKU size of virtual machines in your environment. After implementing a corresponding policy, new and existing resources are evaluated for compliance. With the right policy, existing resources can be brought into compliance.

Cross-organization policy

As your cloud estate grows to span many subscriptions that require enforcement, you will need to focus on a cloud-estate-wide enforcement strategy to ensure policy consistency.

Your subscription design must account for policy in relation to your organizational structure. In addition to helping support complex organization within your subscription design, Azure management groups can be used to assign Azure Policy rules across multiple subscriptions.

Automated enforcement

While standardized deployment templates are effective at a smaller scale, Azure Blueprints allows large-scale standardized provisioning and deployment orchestration of Azure solutions. Workloads across multiple subscriptions can be deployed with consistent policy settings for any resources created.

For IT environments integrating cloud and on-premises resources, you may need use logging and reporting systems to provide hybrid monitoring capabilities. Your third-party or custom operational monitoring systems may offer additional policy enforcement capabilities. For larger or more mature cloud estates, consider how best to integrate these systems with your cloud assets.

Next steps

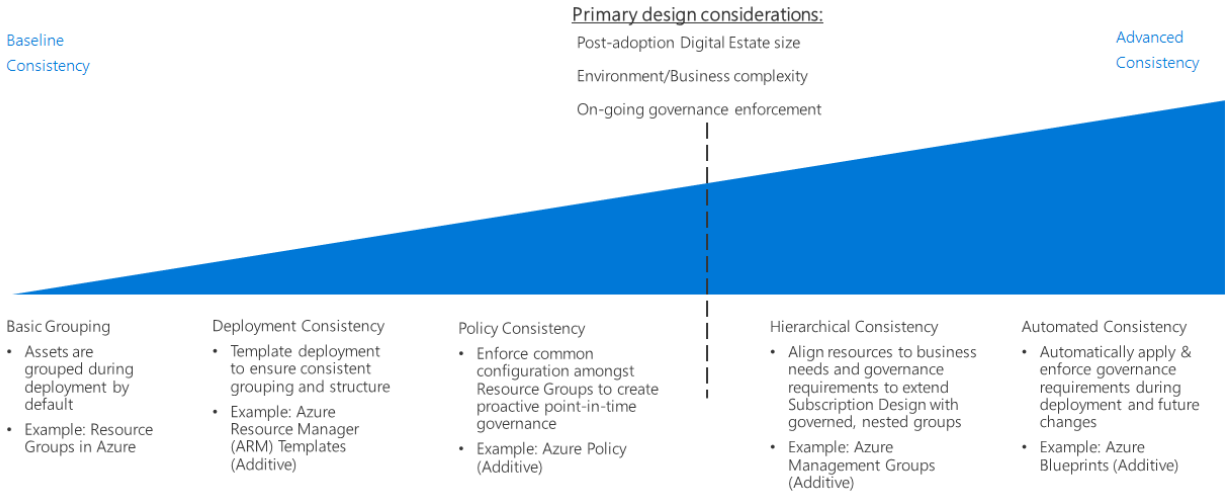
Policy enforcement is just one of the core infrastructure components requiring architectural decisions during a cloud adoption process. Visit the architectural decision guides overview to learn about alternative patterns or models used when making design decisions for other types of infrastructure.

Architectural decision guides

Resource consistency decision guide

3/6/2021 • 5 minutes to read • Edit Online

Azure subscription design defines how you organize your cloud assets in relation to your organization's structure, accounting practices, and workload requirements. In addition to this level of structure, addressing your organizational governance policy requirements across your cloud estate requires the ability to consistently organize, deploy, and manage resources within a subscription.



Jump to: Basic grouping | Deployment consistency | Policy consistency | Hierarchical consistency | Automated consistency

Decisions regarding the level of your cloud estate's resource consistency requirements are primarily driven by these factors: post-migration digital estate size, business or environmental requirements that don't fit neatly within your existing subscription design approaches, or the need to enforce governance over time after resources have been deployed.

As these factors increase in importance, the benefits of ensuring consistent deployment, grouping, and management of cloud-based resources becomes more important. Achieving more advanced levels of resource consistency to meet increasing requirements requires more effort spent in automation, tooling, and consistency enforcement, and this results in additional time spent on change management and tracking.

Basic grouping

In Azure, resource groups are a core resource organization mechanism to logically group resources within a subscription.

Resource groups act as containers for resources with a common lifecycle as well as shared management constraints such as policy or Azure role-based access control (Azure RBAC) requirements. Resource groups can't be nested, and resources can only belong to one resource group. All control plane actions act on all resources in a resource group. For example, deleting a resource group also deletes all resources within that group. The preferred pattern for resource group management is to consider:

1. Are the contents of the resource group developed together?
2. Are the contents of the resource group managed, updated, and monitored together and done so by the same people or teams?
3. Are the contents of the resource group retired together?

If you answered **no** to any of the above points, the resource in question should be placed elsewhere, in another resource group.

> **IMPORTANT**
>
> Resource groups are also region specific; however, it's common for resources to be in different regions within the same resource group because they're managed together as described above. For more information about region selection, see Multiple regions.

Deployment consistency

Building on top of the base resource grouping mechanism, the Azure platform provides a system for using templates to deploy your resources to the cloud environment. You can use templates to create consistent organization and naming conventions when deploying workloads, enforcing those aspects of your resource deployment and management design.

Azure Resource Manager templates allow you to repeatedly deploy your resources in a consistent state using a predetermined configuration and resource group structure. Resource Manager templates help you define a set of standards as a basis for your deployments.

For example, you can have a standard template for deploying a web server workload that contains two virtual machines as web servers combined with a load balancer to distribute traffic between the servers. You can then reuse this template to create structurally identical set of virtual machines with a load balancer whenever this type of workload is needed, only changing the deployment name and IP addresses involved.

You can also programmatically deploy these templates and integrate them with your CI/CD systems.

Policy consistency

To ensure that governance policies are applied when resources are created, part of resource grouping design involves using a common configuration when deploying resources.

By combining resource groups and standardized Resource Manager templates, you can enforce standards for what settings are required in a deployment and what Azure Policy rules are applied to each resource group or resource.

For example, you may have a requirement that all virtual machines deployed within your subscription connect to a common subnet managed by your central IT team. You can create a standard template for deploying workload VMs to create a separate resource group for the workload and deploy the required VMs there. This resource group would have a policy rule to only allow network interfaces within the resource group to be joined to the shared subnet.

For a more in-depth discussion of enforcing your policy decisions within a cloud deployment, see Policy enforcement.

Hierarchical consistency

Resource groups allow you to support additional levels of hierarchy within your organization within the subscription, applying Azure Policy rules and access controls at a resource group level. As the size of your cloud estate grows, you may need to support more complicated cross-subscription governance requirements than can be supported using the Azure Enterprise Agreement's enterprise/department/account/subscription hierarchy.

Azure management groups allow you to organize subscriptions into more sophisticated organizational structures by grouping subscriptions in a hierarchy distinct from your Enterprise Agreement's hierarchy. This alternate hierarchy allows you to apply access control and policy enforcement mechanisms across multiple

subscriptions and the resources they contain. Management group hierarchies can be used to match your cloud estate's subscriptions with operations or business governance requirements. For more information, see the subscription decision guide.

Automated consistency

For large cloud deployments, global governance becomes both more important and more complex. It's crucial to automatically apply and enforce governance requirements when deploying resources, as well as meet updated requirements for existing deployments.

Azure Blueprints enable organizations to support global governance of large cloud estates in Azure. Blueprints move beyond the capabilities provided by standard Azure Resource Manager templates to create complete deployment orchestrations capable of deploying resources and applying policy rules. Blueprints support versioning, the ability to update all subscriptions where the blueprint was used, and the ability to lock down deployed subscriptions to avoid the unauthorized creation and modification of resources.

These deployment packages allow IT and development teams to rapidly deploy new workloads and networking assets that comply with changing organizational policy requirements. Blueprints can also be integrated into CI/CD pipelines to apply revised governance standards to deployments as they're updated.

Next steps

Resource consistency is just one of the core infrastructure components requiring architectural decisions during a cloud adoption process. Visit the architectural decision guides overview to learn about alternative patterns or models used when making design decisions for other types of infrastructure.

Architectural decision guides

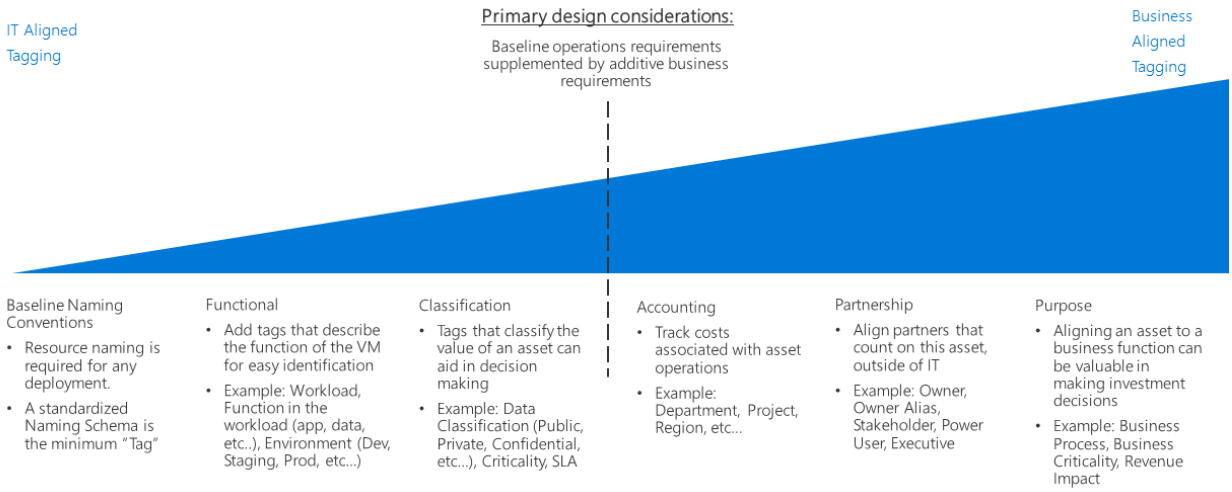
Resource naming and tagging decision guide

8/25/2021 • 5 minutes to read • Edit Online

Organizing cloud-based resources is a crucial task for IT, unless you only have simple deployments. Use naming and tagging standards to organize your resources for these reasons:

- **Resource management:** Your IT teams will need to quickly locate resources associated with specific workloads, environments, ownership groups, or other important information. Organizing resources is critical to assigning organizational roles and access permissions for resource management.
- **Cost management and optimization:** Making business groups aware of cloud resource consumption requires IT to understand the resources and workloads each team is using. The following topics are supported by cost-related tags:
  - Cloud accounting models
  - ROI calculations
  - Cost tracking
  - Budgets
  - Alerts
  - Recurring spend tracking and reporting
  - Post-implementation optimizations
  - Cost-optimization tactics
- **Operations management:** Visibility for the operations management team regarding business commitments and SLAs is an important aspect of ongoing operations. For operations to be managed well, tagging for mission criticality is required.
- **Security:** Classification of data and security impact is a vital data point for the team, when breaches or other security issues arise. To operate securely, tagging for data classification is required.
- **Governance and regulatory compliance:** Maintaining consistency across resources helps identify deviation from agreed-upon policies. Prescriptive guidance for resource tagging demonstrates how one of the following patterns can help when deploying governance practices. Similar patterns are available to evaluate regulatory compliance using tags.
- **Automation:** In addition to making resources easier for IT to manage, a proper organizational scheme allows you to take advantage of automation as part of resource creation, operational monitoring, and the creation of DevOps processes.
- **Workload optimization:** Tagging can help identify patterns and resolve broad issues. Tag can also help identify the assets required to support a single workload. Tagging all assets associated with each workload enables deeper analysis of your mission-critical workloads to make sound architectural decisions.

Tagging decision guide



Jump to: Baseline naming conventions | Resource tagging patterns | Learn more

Your tagging approach can be simple or complex, with the emphasis ranging from supporting IT teams managing cloud workloads to integrating information relating to all aspects of the business.

An IT-aligned tagging focus, such as tagging based on workload, application, function, or environment, reduces the complexity of monitoring assets and simplifies making management decisions based on operational requirements.

Tagging schemes that include a business-aligned focus, such as accounting, business ownership, or business criticality may require a larger time investment to create tagging standards that reflect business interests and maintain those standards over time. This investment yields a tagging system that provides improved accounting for costs and value of IT assets to the overall business. This association of an asset's business value to its operational cost is one of the first steps in changing the cost center perception of IT within your wider organization.

Baseline naming conventions

A standardized naming convention is the starting point for organizing your cloud-hosted resources. A properly structured naming system allows you to quickly identify resources for both management and accounting purposes. If you have existing IT naming conventions in other parts of your organization, consider whether your cloud naming conventions should align with them or if you should establish separate cloud-based standards.

> **NOTE**
> Naming rules and restrictions vary per Azure resource. Your naming conventions must comply with these rules.

Resource tagging patterns

For more sophisticated organization than a consistent naming convention only can provide, cloud platforms support the ability to tag resources.

Tags are metadata elements attached to resources. Tags consist of pairs of key/value strings. The values you include in these pairs is up to you, but the application of a consistent set of global tags, as part of a comprehensive naming and tagging policy, is a critical part of an overall governance policy.

As part of your planning process, use the following questions to help determine the kind of information your resource tags need to support:

- Does your naming and tagging policies need to integrate with existing naming and organizational policies within your company?

- Will you implement a chargeback or showback accounting system? Will you need to associate resources with accounting information for departments, business groups, and teams in more detail than a simple subscription-level breakdown allows?
- Does tagging need to represent details such regulatory compliance requirements for a resource? What about operational details such as uptime requirements, patching schedules, or security requirements?
- What tags will be required for all resources based on centralized IT policy? What tags will be optional? Are individual teams allowed to implement their own custom tagging schemes?

The common tagging patterns listed below provide examples of how tagging can be used to organize cloud assets. These patterns are not meant to be exclusive and can be used in parallel, providing multiple ways of organizing assets based on your company's needs.

TAG TYPE	EXAMPLES	DESCRIPTION

Functional	`app` = `catalogsearch1`
	`tier` = `web`
	`webserver` = `apache`
	`env` = `prod`
	`env` = `staging`
	`env` = `dev`
		Categorize resources in relation to their purpose within a workload, what environment they've been deployed to, or other functionality and operational details.

Classification	`confidentiality` = `private` `SLA` = `24hours`	Classifies a resource by how it's used and what policies apply to it.

Accounting	`department` = `finance`
	`program` = `business-initiative`
	`region` = `northamerica`
		Allows a resource to be associated with specific groups within an organization for billing purposes.

Partnership	`owner` = `jsmith`
	`contactalias` = `catsearchowners`
	`stakeholders` = `user1;user2;user3`
		Provides information about what people (outside of IT) are related or otherwise affected by the resource.

Purpose	`businessprocess` = `support`
	`businessimpact` = `moderate`
	`revenueimpact` = `high`
		Aligns resources to business functions to better support investment decisions.

Learn more

For more information about naming and tagging in Azure, see:

- Naming conventions for Azure resources. Refer to this guidance for recommended naming conventions for Azure resources.
- Use tags to organize your Azure resources and management hierarchy. You can apply tags in Azure at both the resource group and individual resource level, giving you flexibility in the granularity of any accounting reports based on applied tags.

Next steps

Resource tagging is just one of the core infrastructure components requiring architectural decisions during a cloud adoption process. Visit the architectural decision guides overview to learn about alternative patterns or models used when making design decisions for other types of infrastructure.

Architectural decision guides

Encryption decision guide

7/15/2021 • 7 minutes to read • Edit Online

Encrypting data protects it against unauthorized access. Properly implemented encryption policy provides additional layers of security for your cloud-based workloads and guards against attackers and other unauthorized users from both inside and outside your organization and networks.

Jump to: Key management | Data encryption | Learn more

Cloud encryption strategy focuses on corporate policy and compliance mandates. Encrypting resources is desirable, and many Azure services such as Azure Storage and Azure SQL Database enable encryption by default. But encryption has costs that can increase latency and overall resource usage.

For demanding workloads, striking the correct balance between encryption and performance, and determining how data and traffic is encrypted can be essential. Encryption mechanisms can vary in cost and complexity, and both technical and policy requirements can influence your decisions on how encryption is applied and how you store and manage critical secrets and keys.

Corporate policy and third-party compliance are the biggest drivers when planning an encryption strategy. Azure provides multiple standard mechanisms that can meet common requirements for encrypting data, whether at rest or in transit. For policies and compliance requirements that demand tighter controls, such as standardized secrets and key management, encryption in-use, or data-specific encryption, you'll need to develop a more sophisticated encryption strategy to support these requirements.

Key management

Encryption of data in the cloud depends on the secure storage, management, and operational use of encryption keys. A key management system is critical to your organization's ability to create, store, and manage cryptographic keys, as well important passwords, connection strings, and other IT confidential information.

Modern key management systems such as Azure Key Vault support storage and management of software protected keys for dev/test usage and hardware security module (HSM) protected keys for maximum protection of production workloads or sensitive data.

When planning a cloud migration, the following table can help you decide how to store and manage encryption keys, certificates, and secrets that are critical for creating secure and manageable cloud deployments:

QUESTION	CLOUD-NATIVE	BRING YOUR OWN KEY	HOLD YOUR OWN KEY
Does your organization lack centralized key and secret management?	Yes	No	No
Will you need to limit the creation of keys and secrets to devices to your on-premises hardware, while using these keys in the cloud?	No	Yes	No

QUESTION	CLOUD-NATIVE	BRING YOUR OWN KEY	HOLD YOUR OWN KEY
Does your organization have rules or policies in place that would prevent keys from being stored offsite?	No	No	Yes

Cloud-native

With cloud-native key management, all keys and secrets are generated, managed, and stored in a cloud-based vault such as Azure Key Vault. This approach simplifies many IT tasks related to key management, such as key backup, storage, and renewal.

Cloud-native assumptions: Using a cloud-native key management system includes these assumptions:

- You trust the cloud key management solution with creating, managing, and hosting your organization's secrets and keys.
- You enable all on-premises applications and services that rely on accessing encryption services or secrets to access the cloud key management system.

Bring your own key (BYOK)

With a BYOK approach, you generate keys on dedicated HSM hardware within your on-premises environment, then securely transferring these keys to a cloud-based management system such as Azure Key Vault for use with your cloud-hosted resources.

Bring-your-own-key assumptions: Generating keys on-premises and using them with a cloud-based key management system includes these assumptions:

- You trust the underlying security and access control infrastructure of the cloud platform for hosting and using your keys and secrets.
- Your cloud-hosted applications or services can access and use keys and secrets in a robust and secure way.
- You're required by regulatory or organizational policy to keep the creation and management of your organization's secrets and keys on-premises.

On-premises (hold your own key)

Certain scenarios might have regulatory, policy, or technical reasons prohibiting the storage of keys on a cloud-based key management system. If so, you must generate keys using on-premises hardware, store and manage them using an on-premises key management system, and establish a way for cloud-based resources to access these keys for encryption purposes. Note that holding your own key might not be compatible with all Azure-based services.

On-premises key management assumptions: Using an on-premises key management system includes these assumptions:

- You're required by regulatory or organizational policy to keep the creation, management, and hosting of your organization's secrets and keys on-premises.
- Any cloud-based applications or services that rely on accessing encryption services or secrets can access the on-premises key management system.

Data encryption

Consider several different states of data with different encryption needs when planning your encryption policy:

DATA STATE	DATA
Data in transit	Internal network traffic, internet connections, connections between datacenters or virtual networks
Data at rest	Databases, files, virtual drives, PaaS storage
Data in use	Data loaded in RAM or in CPU caches

Data in transit

Data in transit is data moving between resources on the internal network, between datacenters or external networks, or over the internet.

Data in transit is usually encrypted by requiring SSL/TLS protocols for network traffic. Always encrypt traffic between your cloud-hosted resources and external networks or the public internet. PaaS resources typically enforce SSL/TLS encryption by default. Your cloud adoption teams and workload owners should consider enforcing encryption for traffic between IaaS resources hosted inside your virtual networks.

Assumptions about encrypting data in transit: Implementing proper encryption policy for data in transit assumes the following:

- All publicly accessible endpoints in your cloud environment will communicate with the public internet using SSL/TLS protocols.
- When connecting cloud networks with on-premises or other external network over the public internet, use encrypted VPN protocols.
- When connecting cloud networks with on-premises or other external network using a dedicated WAN connection such as ExpressRoute, you will use a VPN or other encryption appliance on-premises paired with a corresponding virtual VPN or encryption appliance deployed to your cloud network.
- If you have sensitive data that shouldn't be included in traffic logs or other diagnostics reports visible to IT staff, you will encrypt all traffic between resources in your virtual network.

Data at rest

Data at rest represents any data not being actively moved or processed, including files, databases, virtual machine drives, PaaS storage accounts, or similar assets. Encrypting stored data protects virtual devices or files against unauthorized access either from external network penetration, rogue internal users, or accidental releases.

PaaS storage and database resources generally enforce encryption by default. IaaS resources can be secured by encrypting data at the virtual disk level or by encrypting the entire storage account hosting your virtual drives. All of these assets can make use of either Microsoft-managed or customer-managed keys stored in Azure Key Vault.

Encryption for data at rest also encompasses more advanced database encryption techniques, such as column-level and row level encryption, providing much more control over exactly what data is being secured.

Your overall policy and compliance requirements, the sensitivity of the data being stored, and the performance requirements of your workloads should determine which assets require encryption.

Assumptions about encrypting data at rest

Encrypting data at rest assumes the following:

- You're storing data that is not meant for public consumption.
- Your workloads can accept the added latency cost of disk encryption.

Data in use

Encryption for data in use involves securing data in nonpersistent storage, such as RAM or CPU caches. Use of technologies such as full memory encryption, enclave technologies, such as Intel's Secure Guard Extensions (SGX). This also includes cryptographic techniques, such as homomorphic encryption that can be used to create secure, trusted execution environments.

Assumptions about encrypting data in use: Encrypting data in use assumes the following:

- You're required to maintain data ownership separate from the underlying cloud platform at all times, even at the RAM and CPU level.

Learn more

For more information about encryption and key management in Azure, see:

- Azure encryption overview: A detailed description of how Azure uses encryption to secure both data at rest and data in transit.
- Azure Key Vault: Key Vault is the primary key management system for storing and managing cryptographic keys, secrets, and certificates within Azure.
- Azure data security and encryption best practices. A discussion of Azure data security and encryption best practices.
- Confidential computing in Azure: Azure's confidential computing initiative provides tools and technology to create trusted execution environments or other encryption mechanisms to secure data in use.

Next steps

Encryption is just one of the core infrastructure components requiring architectural decisions during a cloud adoption process. To learn about alternative patterns or models used when making design decisions for other types of infrastructure, see the architectural decision guides overview.

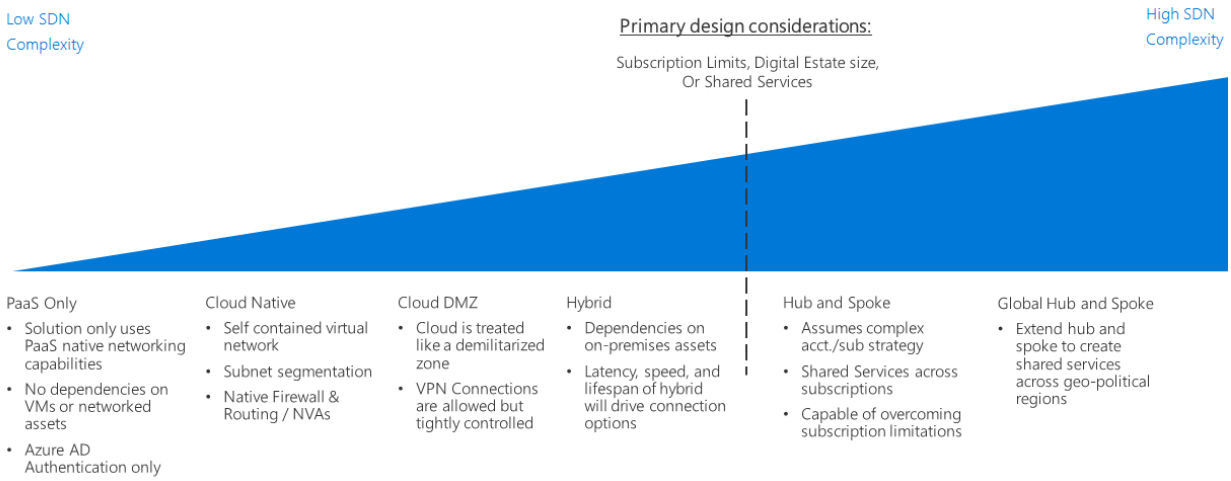
Architectural decision guides overview

Software Defined Networking decision guide

5/18/2021 • 3 minutes to read • Edit Online

Software Defined Networking (SDN) is a network architecture designed to allow virtualized networking functionality that can be centrally managed, configured, and modified through software. SDN enables the creation of cloud-based networks using the virtualized equivalents to physical routers, firewalls, and other networking devices used in on-premises networks. SDN is critical to creating secure virtual networks on public cloud platforms such as Azure.

Networking decision guide



Jump to: PaaS only | Cloud-native | Cloud DMZ | Hybrid | Hub and spoke model | Learn more

SDN provides several options with varying degrees of pricing and complexity. The above discovery guide provides a reference to quickly personalize these options to best align with specific business and technology strategies.

The inflection point in this guide depends on several key decisions that your cloud strategy team has made before making decisions about networking architecture. Most important among these are decisions involving your digital estate definition and subscription design, which may also require inputs from decisions made related to your cloud accounting and global markets strategies.

Small single-region deployments of fewer than 1,000 VMs are less likely to be significantly affected by this inflection point. Conversely, large adoption efforts with more than 1,000 VMs, multiple business units, or multiple geopolitical markets, could be substantially affected by your SDN decision and this key inflection point.

Choose the right virtual networking architectures

This section expands on the decision guide to help you choose the right virtual networking architectures.

There are many ways to implement SDN technologies to create cloud-based virtual networks. How you structure the virtual networks used in your migration and how those networks interact with your existing IT infrastructure will depend on a combination of the workload requirements and your governance requirements.

When planning which virtual networking architecture or combination of architectures to consider when planning your cloud migration, consider the following questions to help determine what's right for your organization:

QUESTION	PAAS-ONLY	CLOUD-NATIVE	CLOUD DMZ	HYBRID	HUB AND SPOKE
Will your workload only use PaaS services and not require networking capabilities beyond those provided by the services themselves?	Yes	No	No	No	No
Does your workload require integration with on-premises applications?	No	No	Yes	Yes	Yes
Have you established mature security policies and secure connectivity between your on-premises and cloud networks?	No	No	No	Yes	Yes
Does your workload require authentication services not supported through cloud identity services, or do you need direct access to on-premises domain controllers?	No	No	No	Yes	Yes
Will you need to deploy and manage a large number of VMs and workloads?	No	No	No	No	Yes
Will you need to provide centralized management and on-premises connectivity while delegating control over resources to individual workload teams?	No	No	No	No	Yes

Virtual networking architectures

Learn more about the primary Software Defined Networking architectures:

- PaaS-only: Most platform as a service (PaaS) products support a limited set of built-in networking features and may not require an explicitly defined software defined network to support workload requirements.
- Cloud-native: A cloud-native architecture supports cloud-based workloads using virtual networks built on the cloud platform's default Software Defined Networking capabilities, without reliance on on-premises or other external resources.
- Cloud DMZ: Supports limited connectivity between your on-premises and cloud networks, secured through the implementation of a perimeter network tightly controlling traffic between the two environments.
- Hybrid: The hybrid cloud network architecture allows virtual networks in trusted cloud environments to access your on-premises resources and vice versa.
- Hub and spoke: The hub and spoke architecture allows you to centrally manage external connectivity and shared services, isolate individual workloads, and overcome potential subscription limits.

Learn more

For more information about Software Defined Networking in Azure, see:

- Azure Virtual Network. On Azure, the core SDN capability is provided by Azure Virtual Network, which acts as a cloud analog to physical on-premises networks. Virtual networks also act as a default isolation boundary between resources on the platform.
- Azure best practices for network security. Recommendations from the Azure security team on how to configure your virtual networks to minimize security vulnerabilities.

Next steps

Software Defined Networking is just one of the core infrastructure components requiring architectural decisions during a cloud adoption process. Visit the architectural decision guides overview to learn about alternative patterns or models used when making design decisions for other types of infrastructure.

Architectural decision guides

Software Defined Networking: PaaS-only

3/6/2021 • 2 minutes to read • Edit Online

When you implement a platform as a service (PaaS) resource, the deployment process automatically creates an assumed underlying network with a limited number of controls over that network, including load balancing, port blocking, and connections to other PaaS services.

In Azure, several PaaS resource types can be deployed into a virtual network or connected to a virtual network, integrating these resources with your existing virtual networking infrastructure. Other services, such as App Service Environment, Azure Kubernetes Service (AKS), and Service Fabric must be deployed within a virtual network. In many cases, a PaaS-only networking architecture, relying solely on the default native networking capabilities provided by PaaS resources, is sufficient to meet a workload's connectivity and traffic management requirements.

If you're considering a PaaS only networking architecture, be sure you validate that the required assumptions align with your requirements.

PaaS-only assumptions

Deploying a PaaS-only networking architecture assumes the following:

- The application being deployed is a standalone application or depends only on other PaaS resources that do not require a virtual network.
- Your IT operations teams can update their tools, training, and processes to support management, configuration, and deployment of standalone PaaS applications.
- The PaaS application is not part of a broader cloud migration effort that will include IaaS resources.

These assumptions are minimum qualifiers aligned to deploying a PaaS-only network. While this approach may align with the requirements of a single application deployment, each cloud adoption team should consider these long-term questions:

- Will this deployment expand in scope or scale to require access to other non-PaaS resources?
- Are other PaaS deployments planned beyond the current solution?
- Does the organization have plans for other future cloud migrations?

The answers to these questions would not preclude a team from choosing a PaaS only option but should be considered before making a final decision.

Software Defined Networking: Cloud-native

3/6/2021 • 2 minutes to read • Edit Online

A cloud-native virtual network is required when deploying IaaS resources such as virtual machines to a cloud platform. Access to virtual networks from external sources, similar to the web, need to be explicitly provisioned. These types of virtual networks support the creation of subnets, routing rules, and virtual firewall and traffic management devices.

A cloud-native virtual network has no dependencies on your organization's on-premises or other non-cloud resources to support the cloud-hosted workloads. All required resources are provisioned either in the virtual network itself or by using managed PaaS offerings.

Cloud-native assumptions

Deploying a cloud-native virtual network assumes the following:

- The workloads you deploy to the virtual network have no dependencies on applications or services that are accessible only from inside your on-premises network. Unless they provide endpoints accessible over the public internet, applications and services hosted internally on-premises are not usable by resources hosted on a cloud platform.
- Your workload's identity management and access control depends on the cloud platform's identity services or IaaS servers hosted in your cloud environment. You will not need to directly connect to identity services hosted on-premises or other external locations.
- Your identity services do not need to support single sign-on (SSO) with on-premises directories.

Cloud-native virtual networks have no external dependencies. This makes them simple to deploy and configure, and as a result this architecture is often the best choice for experiments or other smaller self-contained or rapidly iterating deployments.

Additional issues your cloud adoption teams should consider when discussing a cloud-native virtual networking architecture include:

- Existing workloads designed to run in an on-premises datacenter may need extensive modification to take advantage of cloud-based functionality, such as storage or authentication services.
- Cloud-native networks are managed solely through the cloud platform management tools, and therefore may lead to management and policy divergence from your existing IT standards as time goes on.

Next steps

For more information about cloud-native virtual networking in Azure, see:

- Azure Virtual Network guides: Newly created virtual networks are cloud-native by default. Use these guides to help plan the design and deployment of your virtual networks.
- Azure networking limits: Each virtual network and connected resources exists in a single subscription. These resources bound by subscription limits.

Software Defined Networking: Cloud DMZ

3/6/2021 • 2 minutes to read • Edit Online

The Cloud DMZ network architecture allows limited access between your on-premises and cloud-based networks, using a virtual private network (VPN) to connect the networks. Although a DMZ model is commonly used when you want to secure external access to a network, the Cloud DMZ architecture discussed here is intended specifically to secure access to the on-premises network from cloud-based resources and vice versa.

This architecture is designed to support scenarios where your organization wants to start integrating cloud-based workloads with on-premises workloads but may not have fully matured cloud security policies or acquired a secure dedicated WAN connection between the two environments. As a result, cloud networks should be treated like a DMZ to ensure on-premises services are secure.

The DMZ deploys network virtual appliances (NVAs) to implement security functionality such as firewalls and packet inspection. Traffic passing between on-premises and cloud-based applications or services must pass through the DMZ where it can be audited. VPN connections and the rules determining what traffic is allowed through the DMZ network are strictly controlled by IT security teams.

Cloud DMZ assumptions

Deploying a Cloud DMZ includes the following assumptions:

- Your security teams have not fully aligned on-premises and cloud-based security requirements and policies.
- Your cloud-based workloads require access to limited subset of services hosted on your on-premises or third-party networks, or users or applications in your on-premises environment need limited access to cloud-hosted resources.
- Implementing a VPN connection between your on-premises networks and cloud provider is not prevented by corporate policy, regulatory requirements, or technical compatibility issues.
- Your workloads either do not require multiple subscriptions to bypass subscription resource limits, or they involve multiple subscriptions but don't require central management of connectivity or shared services used by resources spread across multiple subscriptions.

Your cloud adoption teams should consider the following issues when looking at implementing a Cloud DMZ virtual networking architecture:

- Connecting on-premises networks with cloud networks increases the complexity of your security requirements. Even though connections between cloud networks and the on-premises environment are secured, you still need to ensure cloud resources are secured. Any public IPs created to access cloud-based workloads need to be properly secured using a public-facing DMZ or Azure Firewall.
- The Cloud DMZ architecture is commonly used as a stepping stone while connectivity is further secured and security policy aligned between on-premises and cloud networks, allowing a broader adoption of a full-scale hybrid networking architecture. It may also apply to isolated deployments with specific security, identity, and connectivity needs that the Cloud DMZ approach satisfies.

Learn more

For more information about implementing a Cloud DMZ in Azure, see:

- Implement a DMZ between Azure and your on-premises datacenter. This article discusses how to implement

a secure hybrid network architecture in Azure.

Software Defined Networking: Hybrid network

3/6/2021 • 2 minutes to read • Edit Online

The hybrid cloud network architecture allows virtual networks to access your on-premises resources and services and vice versa, using a dedicated WAN connection such as ExpressRoute or other connection method to directly connect the networks.

Building on the cloud-native virtual network architecture, a hybrid virtual network is isolated when initially created. Adding connectivity to the on-premises environment grants access to and from the on-premises network, although all other inbound traffic targeting resources in the virtual network need to be explicitly allowed. You can secure the connection using virtual firewall devices and routing rules to limit access or you can specify exactly what services can be accessed between the two networks using cloud-native routing features or deploying network virtual appliances (NVAs) to manage traffic.

Although the hybrid networking architecture supports VPN connections, dedicated WAN connections like ExpressRoute are preferred due to higher performance and increased security.

Hybrid assumptions

Deploying a hybrid virtual network includes the following assumptions:

- Your IT security teams have aligned on-premises and cloud-based network security policy to ensure cloud-based virtual networks can be trusted to communicated directly with on-premises systems.
- Your cloud-based workloads require access to storage, applications, and services hosted on your on-premises or third-party networks, or your users or applications in your on-premises need access to cloud-hosted resources.
- You need to migrate existing applications and services that depend on on-premises resources, but don't want to expend the resources on redevelopment to remove those dependencies.
- Connecting your on-premises networks to cloud resources over VPN or dedicated WAN is not prevented by corporate policy, data sovereignty requirements, or other regulatory compliance issues.
- Your workloads either do not require multiple subscriptions to bypass subscription resource limits, or your workloads involve multiple subscriptions but do not require central management of connectivity or shared services used by resources spread across multiple subscriptions.

Your cloud adoption teams should consider the following issues when looking at implementing a hybrid virtual networking architecture:

- Connecting on-premises networks with cloud networks increases the complexity of your security requirements. Both networks must be secured against external vulnerabilities and unauthorized access from both sides of the hybrid environment.
- Scaling the number and size of workloads within a hybrid cloud environment can add significant complexity to routing and traffic management.
- You will need to develop compatible management and access control policies to maintain consistent governance throughout your organization.

Learn more

For more information about hybrid networking in Azure, see:

- Hybrid network reference architecture. Azure hybrid virtual networks use either an ExpressRoute circuit or Azure VPN to connect your virtual network with your organization's existing IT assets not hosted in Azure. This article discusses the options for creating a hybrid network in Azure.

Software Defined Networking: Hub and spoke

3/6/2021 • 2 minutes to read • Edit Online

The hub and spoke networking model organizes your Azure-based cloud network infrastructure into multiple connected virtual networks. This model allows you to more efficiently manage common communication or security requirements and deal with potential subscription limitations.

In the hub and spoke model, the *hub* is a virtual network that acts as a central location for managing external connectivity and hosting services used by multiple workloads. The *spokes* are virtual networks that host workloads and connect to the central hub through virtual network peering.

All traffic passing in or out of the workload spoke networks is routed through the hub network where it can be routed, inspected, or otherwise managed by centrally managed IT rules or processes.

This model aims to address the following concerns:

- **Cost savings and management efficiency.** Centralizing services that can be shared by multiple workloads, such as network virtual appliances (NVAs) and DNS servers, in a single location allows IT to minimize redundant resources and management effort across multiple workloads.
- **Overcoming subscription limits.** Large cloud-based workloads may require the use of more resources than are allowed within a single Azure subscription. Peering workload virtual networks from different subscriptions to a central hub can overcome these limits. For more information, see Azure networking limits.
- **Separation of concerns.** The ability to deploy individual workloads between central IT teams and workload teams.

The following diagram shows an example hub and spoke architecture including centrally managed hybrid connectivity.



The hub and spoke architecture is often used alongside the hybrid networking architecture, providing a centrally managed connection to your on-premises environment shared between multiple workloads. In this scenario, all traffic traveling between the workloads and on-premises passes through the hub where it can be managed and secured.

Hub and spoke assumptions

Implementing a hub and spoke virtual networking architecture assumes the following:

- Your cloud deployments will involve workloads hosted in separate working environments, such as development, test, and production, that all rely on a set of common services such as DNS or directory services.
- Your workloads do not need to communicate with each other but have common external communications and shared services requirements.
- Your workloads require more resources than are available within a single Azure subscription.
- You need to provide workload teams with delegated management rights over their own resources while maintaining central security control over external connectivity.

Global hub and spoke

Hub and spoke architectures are commonly implemented with virtual networks deployed to the same Azure region to minimize latency between networks. Large organizations with a global presence might need to deploy

workloads across multiple regions for availability, disaster recovery, or regulatory requirements. The hub and spoke model can use of Azure global virtual network peering to extend centralized management and shared services across regions and support workloads distributed across the world.

Learn more

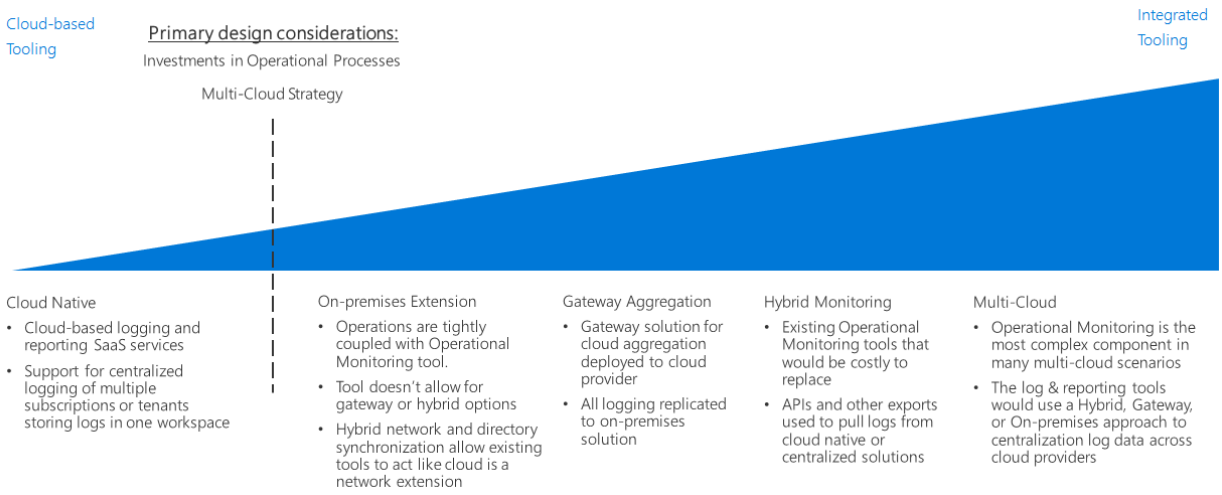
For reference architectures showing how to implement hub and spoke networks on Azure, see:

- Implement a hub and spoke network topology in Azure
- Implement a hub and spoke network topology with shared services in Azure

Logging and reporting decision guide

3/6/2021 • 7 minutes to read • Edit Online

All organizations need mechanisms for notifying IT teams of performance, uptime, and security issues before they become serious problems. A successful monitoring strategy allows you to understand how the individual components that make up your workloads and networking infrastructure are performing. Within the context of a public cloud migration, integrating logging and reporting with any of your existing monitoring systems, while surfacing important events and metrics to the appropriate IT staff, is critical in ensuring your organization is meeting uptime, security, and policy compliance goals.



Jump to: Planning your monitoring infrastructure | Cloud-native | On-premises extension | Gateway aggregation | Hybrid monitoring (on-premises) | Hybrid monitoring (cloud-based) | Multicloud | Learn more

The inflection point when determining a cloud logging and reporting strategy is based primarily on existing investments your organization has made in operational processes, and to some degree any requirements you have to support a multicloud strategy.

Activities in the cloud can be logged and reported in multiple ways. Cloud-native and centralized logging are two common managed service options that are driven by the subscription design and the number of subscriptions.

Plan your monitoring infrastructure

When planning your deployment, you need to consider where logging data is stored and how you will integrate cloud-based reporting and monitoring services with your existing processes and tools.

QUESTION	CLOUD-NATIVE	ON-PREMISES EXTENSION	HYBRID MONITORING	GATEWAY AGGREGATION
Do you have an existing on-premises monitoring infrastructure?	No	Yes	Yes	No

QUESTION	CLOUD-NATIVE	ON-PREMISES EXTENSION	HYBRID MONITORING	GATEWAY AGGREGATION
Do you have requirements preventing storage of log data on external storage locations?	No	Yes	No	No
Do you need to integrate cloud monitoring with on-premises systems?	No	No	Yes	No
Do you need to process or filter telemetry data before submitting it to your monitoring systems?	No	No	No	Yes

Cloud-native

If your organization currently lacks established logging and reporting systems, or if your planned deployment does not need to be integrated with existing on-premises or other external monitoring systems, a cloud-native SaaS solution such as Azure Monitor, is the simplest choice.

In this scenario, all log data is recorded and stored in the cloud, while the logging and reporting tools that process and surface information to IT staff are provided by the Azure platform and Azure Monitor.

Custom logging solutions based on Azure Monitor can be implemented as needed for each subscription or workload in smaller or experimental deployments. These solutions are organized centrally to monitor log data across your entire cloud estate.

Cloud-native assumptions: Using a cloud-native logging and reporting system assumes the following:

- You do not need to integrate the log data from you cloud workloads into existing on-premises systems.
- You will not be using your cloud-based reporting systems to monitor on-premises systems.

On-premises extension

It might require substantial redevelopment effort for applications and services migrating to the cloud to use cloud-based logging and reporting solutions such as Azure Monitor. In these cases, consider allowing these workloads to continue sending telemetry data to existing on-premises systems.

To support this approach, your cloud resources must communicate directly with your on-premises systems through a combination of hybrid networking and cloud-hosted domain services. With this in place, the cloud virtual network functions as a network extension of the on-premises environment. Therefore, cloud-hosted workloads can communicate directly with your on-premises logging and reporting system.

This approach capitalizes on your existing investment in monitoring tooling with limited modification to any cloud-deployed applications or services. This is often the fastest approach to support monitoring during a lift and shift migration. But it won't capture log data produced by cloud-based PaaS and SaaS resources, and it will omit any VM-related logs generated by the cloud platform itself such as VM status. As a result, this pattern should be a temporary solution until a more comprehensive hybrid monitoring solution is implemented.

On-premises-only assumptions:

- You need to maintain log data only in your on-premises environment only, either in support of technical requirements or due to regulatory or policy requirements.

- Your on-premises systems do not support hybrid logging and reporting or gateway aggregation solutions.
- Your cloud-based applications can submit telemetry directly to your on-premises logging systems or monitoring agents that submit to on-premises can be deployed to workload VMs.
- Your workloads don't depend on PaaS or SaaS services that require cloud-based logging and reporting.

Gateway aggregation

For scenarios where the amount of cloud-based telemetry data is large or existing on-premises monitoring systems need log data modified before it can be processed, a log data gateway aggregation service might be required.

A gateway service is deployed to your cloud provider. Then, relevant applications and services are configured to submit telemetry data to the gateway instead of a default logging system. The gateway can then process the data: aggregating, combining, or otherwise formatting it before then submitting it to your monitoring service for ingestion and analysis.

Also, a gateway can be used to aggregate and preprocess telemetry data bound for cloud-native or hybrid systems.

Gateway aggregation assumptions:

- You expect large volumes of telemetry data from your cloud-based applications or services.
- You need to format or otherwise optimize telemetry data before submitting it to your monitoring systems.
- Your monitoring systems have APIs or other mechanisms available to ingest log data after processing by the gateway.

Hybrid monitoring (on-premises)

A hybrid monitoring solution combines log data from both your on-premises and cloud resources to provide an integrated view into your IT estate's operational status.

If you have an existing investment in on-premises monitoring systems that would be difficult or costly to replace, you might need to integrate the telemetry from your cloud workloads into preexisting on-premises monitoring solutions. In a hybrid on-premises monitoring system, on-premises telemetry data continues to use the existing on-premises monitoring system. Cloud-based telemetry data is either sent to the on-premises monitoring system directly, or the data is sent to Azure Monitor then compiled and ingested into the on-premises system at regular intervals.

On-premises hybrid monitoring assumptions: Using an on-premises logging and reporting system for hybrid monitoring assumes the following:

- You need to use existing on-premises reporting systems to monitor cloud workloads.
- You need to maintain ownership of log data on-premises.
- Your on-premises management systems have APIs or other mechanisms available to ingest log data from cloud-based systems.

> **TIP**
> As part of the iterative nature of cloud migration, transitioning from distinct cloud-native and on-premises monitoring to a partial hybrid approach is likely as the integration of cloud-based resources and services into your overall IT estate matures.

Hybrid monitoring (cloud-based)

If you do not have a compelling need to maintain an on-premises monitoring system, or you want to replace on-premises monitoring systems with a centralized cloud-based solution, you can also choose to integrate on-premises log data with Azure Monitor to provide centralized cloud-based monitoring system.

Mirroring the on-premises centered approach, in this scenario cloud-based workloads would submit telemetry direct to Azure Monitor, and on-premises applications and services would either submit telemetry directly to Azure Monitor, or aggregate that data on-premises for ingestion into Azure Monitor at regular intervals. Azure Monitor would then serve as your primary monitoring and reporting system for your entire IT estate.

Cloud-based hybrid monitoring assumptions: Using cloud-based logging and reporting systems for hybrid monitoring assumes the following:

- You don't depend on existing on-premises monitoring systems.
- Your workloads do not have regulatory or policy requirements to store log data on-premises.
- Your cloud-based monitoring systems have APIs or other mechanisms available to ingest log data from on-premises applications and services.

Multicloud

Integrating logging and reporting capabilities across a multiple-cloud platform can be complicated. Services offered between platforms are often not directly comparable, and logging and telemetry capabilities provided by these services differ as well.

Multicloud logging support often requires the use of gateway services to process log data into a common format before submitting data to a hybrid logging solution.

Learn more

Azure Monitor is the default reporting and monitoring service for Azure. It provides:

- A unified platform for collecting application telemetry, host telemetry (such as VMs), container metrics, Azure platform metrics, and event logs.
- Visualization, queries, alerts, and analytical tools. It can provide insights into virtual machines, guest operating systems, virtual networks, and workload application events.
- REST APIs for integration with external services and automation of monitoring and alerting services.
- Integration with many popular third-party vendors.

Next steps

Logging and reporting is just one of the core infrastructure components requiring architectural decisions during a cloud adoption process. Visit the architectural decision guides overview to learn about alternative patterns or models used when making design decisions for other types of infrastructure.

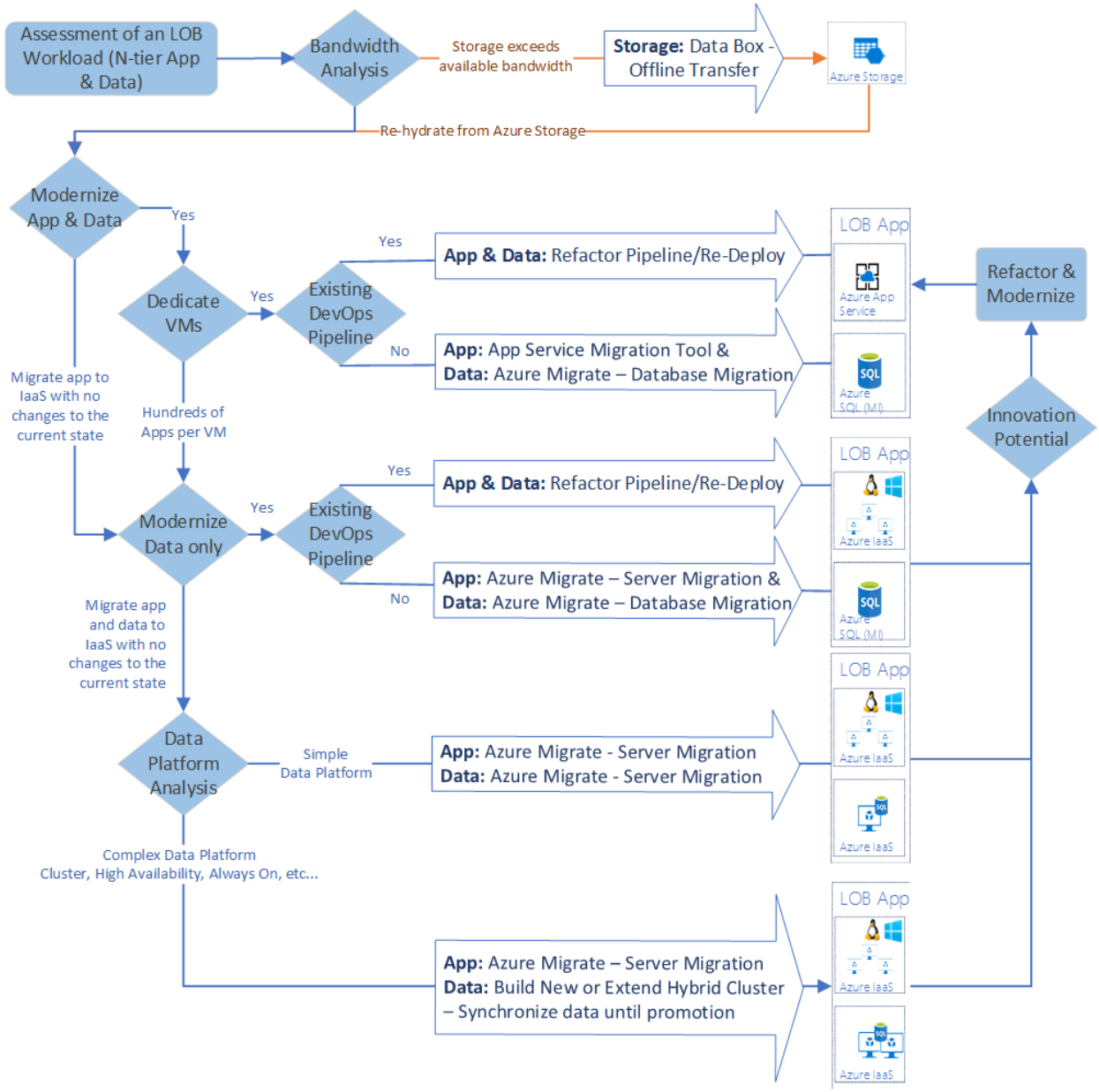
Architectural decision guides

Migration tools decision guide

5/18/2021 • 4 minutes to read • Edit Online

The strategy and tools you use to migrate an application to Azure will largely depend on your business motivations, technology strategies, and timelines, as well as a deep understanding of the actual workload and assets (infrastructure, apps, and data) being migrated. The following decision tree serves as high-level guidance for selecting the best tools to use based on migration decisions. Treat this decision tree as a starting point.

The choice to migrate using platform as a service (PaaS) or infrastructure as a service (IaaS) technologies is driven by the balance between cost, time, existing technical debt, and long-term returns. IaaS is often the fastest path to the cloud with the least amount of required change to the workload. PaaS could require modifications to data structures or source code, but produces substantial long-term returns in the form of reduced operating costs and greater technical flexibility. In the following diagram, the term *modernize* is used to reflect a decision to modernize an asset during migration and migrate the modernized asset to a PaaS platform.



Key questions

Answering the following questions will allow you to make decisions based on the above tree.

- **Would modernization of the application platform during migration prove to be a wise investment of time, energy, and budget?** PaaS technologies such as Azure App Service or Azure Functions can increase deployment flexibility and reduce the complexity of managing virtual machines to host applications. Applications may require refactoring before they can take advantage of these cloud-native capabilities, potentially adding significant time and cost to a migration effort. If your application can migrate to PaaS technologies with a minimum of modifications, it's likely a good candidate for modernization. If extensive refactoring would be required, a migration using IaaS-based virtual machines may be a better choice.
- **Would modernization of the data platform during migration prove to be a wise investment of time, energy, and budget?** As with application migration, Azure PaaS managed storage options, such as Azure SQL Database, Azure Cosmos DB, and Azure Storage, offer significant management and flexibility benefits, but migrating to these services may require refactoring of existing data and the applications that use that data. Data platforms typically require less refactoring than the application platform would. Therefore, it's common for the data platform to be modernized, even though the application platform remains the same. If your data can be migrated to a managed data service with minimal changes, it's a good candidate for modernization. Data that would require extensive time or cost to be refactored to use these PaaS services may be better migrated using IaaS-based virtual machines to better match existing hosting capabilities.
- **Is your application currently running on dedicated virtual machines or sharing hosting with other applications?** Application running on dedicated virtual machines may be more easily migrated to PaaS hosting options than applications running on shared servers.
- **Will your data migration exceed your network bandwidth?** Network capacity between your on-premises data sources and Azure can be a bottleneck on data migration. If the data you need to transfer faces bandwidth limitations that prevent efficient or timely migration, you may need to look into alternative or offline transfer mechanisms. The Cloud Adoption Framework's article on migration replication discusses how replication limits can affect migration efforts. As part of your migration assessment, consult your IT teams to verify your local and WAN bandwidth is capable of handling your migration requirements. Also see the migration scenario for handling storage requirements that exceed network capacity during a migration.
- **Does your application make use of an existing DevOps pipeline?** In many cases, Azure Pipelines can be easily refactored to deploy applications to cloud-based hosting environments.
- **Does your data have complex data storage requirements?** Production applications usually require data storage that is highly available, offers always-on functionality and similar service uptime and continuity features. Azure PaaS-based managed database options, such as Azure SQL Database, Azure Database for MySQL, and Azure Cosmos DB all offer 99.99 percent uptime service-level agreements. Conversely, IaaS-based SQL Server on Azure VMs offers single-instance service-level agreements of 99.95 percent. If your data cannot be modernized to use PaaS storage options, guaranteeing higher IaaS uptime will involve more complex data storage scenarios such as running SQL Server Always On clusters and continuously syncing data between instances. This can involve significant hosting and maintenance costs, so balancing uptime requirements, modernization effort, and overall budgetary impact is important when considering your data migration options.

Innovation and migration

In line with the Cloud Adoption Framework's emphasis on incremental migration efforts, an initial decision on migration strategy and tooling does not rule out future innovation efforts to update an application to take advantage of opportunities presented by the Azure platform. While an initial migration effort might focus primarily on rehosting using an IaaS approach, you should plan to revisit your cloud-hosted application portfolio regularly to investigate optimization opportunities.

Learn more

- Cloud fundamentals: Overview of Azure compute options: Provides information on the capabilities of Azure IaaS and PaaS compute options.
- Cloud fundamentals: Choose the right data store: Discusses PaaS storage options available on the Azure platform.
- Migration best practices: Data requirements exceed network capacity during a migration effort: Discusses alternative data migration mechanisms for scenarios where data migration is hindered by available network bandwidth.
- SQL Database: Choose the right SQL Server option in Azure: Discussion of the options and business justifications for choosing to host your SQL Server workloads in a managed infrastructure (IaaS) or a managed service (PaaS) environment.

Cloud Operating Model is now part of the Microsoft Cloud Adoption Framework for Azure

5/18/2021 • 2 minutes to read • Edit Online

In early 2018, Microsoft released the Cloud Operating Model (COM). The COM was a guide that helped customers understand the *what* and the *why* of digital transformation. This helped customers get a sense of all the areas that needed to be addressed: business strategy, culture strategy, and technology strategy. What was not included in the COM were the specific *how-to* steps, which left customers wondering, "Where do we go from here?"

The Microsoft Cloud Adoption Framework for Azure, is designed to help you understand the **what** and **why** and provide unified guidance on the **how** to help accelerate your cloud adoption efforts.

Using Cloud Operating Model practices within the Cloud Adoption Framework

For an approach that's similar to the COM, begin with one of the following:

- Get started: Accelerate migration
- Get started: Build and innovate in the cloud
- Enable success with a sound operating model

Azure enterprise scaffold is now the Microsoft Cloud Adoption Framework for Azure

5/4/2021 • 2 minutes to read • Edit Online

The Azure enterprise scaffold has been integrated into the Microsoft Cloud Adoption Framework for Azure. The goals of the enterprise scaffold are now addressed in the Ready methodology of the Cloud Adoption Framework. The enterprise scaffold content has been deprecated.

To begin using the Cloud Adoption Framework, see:

- Ready overview
- Azure landing zones
- Landing zone considerations.

If you need to review the deprecated content, see the Azure enterprise scaffold.

Azure Virtual Datacenter

5/4/2021 • 2 minutes to read • Edit Online

A more robust platform architecture and implementation have been created to build on the prior Azure Virtual Datacenter (VDC) approach. Enterprise-scale landing zones in the Microsoft Cloud Adoption Framework for Azure are now the recommended approach for larger cloud-adoption efforts.

The following guidance serves as a significant part of the foundation for the Ready methodology and the Govern methodology in the Cloud Adoption Framework. To support customers making this transition, the following resources are archived and maintained in a separate GitHub repository.

- Azure Virtual Datacenter: This e-book shows you how to deploy enterprise workloads to the Azure cloud platform while respecting your existing security and networking policies.
- Azure Virtual Datacenter lift-and-shift guide: This white paper discusses the process that enterprise IT staff and decision makers can use to identify and plan the migration of applications and servers to Azure using a lift-and-shift approach while minimizing other development costs and optimizing cloud hosting options.