

Component	Common characteristics
Applications	Programming languages and associated metadata
Application-specific roadmaps	Decisions about whether to maintain, enhance, migrate, or eliminate applications
Data structures	File types, datasets, and database technologies
Dependencies	Inter-application and data dependencies, backups, and archival strategies
Hardware	MIPS (million instructions per second), MSUs (million service units), and LPARs (logical partitions)
Integration types	Methods such as file transfers, message queues, sockets, APIs, replication, and change data capture (CDC)
Security	Implemented protocols such as RACF, ACF2, and top secret
Support	Skillsets within a technical team for maintaining the migrated environment

As you prepare for the POC, you should review case studies that mirror your workload's environment. Assess the methodologies that were used. Determine time frame estimates and costs associated with each potential solution to create a comparative analysis. Include common characteristics that are in the previous table.

## Define criteria to select a workload

When you define workloads for a POC, consider the viability and the effect of the resulting migration. Choose workloads that demonstrate the immediate benefits of a POC and can also transition into full-scale migration. Treat a POC as a pilot, so you can extract value in terms of cost, business functionality, and technical advancement. Take advantage of a POC so you can quickly and efficiently migrate to the cloud.

## Define business criteria

To select a workload, define the business criteria. Identify the key business factors and requirements that influence workloads for a POC. This step ensures that the workload you select aligns with your strategic business goals and drivers, such as cost reduction or critical-system modernization. Evaluate business criteria to determine workloads that offer the highest value and lowest risk and to effectively use resources during the migration process.

## Understand business drivers

Determine the underlying motives that drive the need for modernizing mainframe workloads. These motives include cost efficiencies, operational improvements, and strategic objectives. Align business drivers to demonstrate clear benefits, which helps secure stakeholder buy-in and justify investments in the migration project.

**Understand the criticality of workloads.** Workload criticality is a measure of the importance of a workload to the business. Understand the criticality of workloads to help build a roadmap for modernization. Carefully determine the areas that you need to validate in a workload. For example, you might select a workload for a POC, and the initial phase of migration might draw several dependencies to mission-critical workloads. This new information might require you to include the dependencies in the scope, which drastically increases the complexity and risk of the migration.

**Prioritize workloads by the degree of associated risk.** Prioritize workloads by the use risks or current constraints of the environment. For example, retiring workload support resources or unsupported technical components are driving factors to prioritize workloads for migration and reduce risk.

**Prepare personnel.** It's important to identify the right support resources and make them available to personnel so you can validate the success of the POC. Group workloads by department to help limit the number of teams that are involved in the initial scope.

## Establish POC goals

To establish POC goals, set clear objectives for the POC in terms of technical validation, business benefits, and strategic alignment. Objectives help to guide the decision-making process. Choose workloads that demonstrate the POC's objectives and provide measurable outcomes that support the business case for migration. Consider the following examples of POC goals.

- *Reduce or exit a datacenter:* When you downsize or exit a datacenter, it's crucial to assess how the existing distributed and midrange systems interface with the mainframe. Use the resulting information to help pinpoint which workloads are

best-suited for the POC. Choose workloads that are distributed and sensitive to latency so you can optimize and potentially accelerate the modernization process.

- *Reduce mainframe consumption:* Map workload usage for the POC to decrease the mainframe's operational footprint. This method helps to prevent further investment in capacity and sets the foundation for a cost-effective, streamlined operational environment.
- *Optimize the procurement model:* A procurement model refers to the specific approach and terms under which an organization acquires and pays for its mainframe resources, both fixed and variable costs. Cost savings depend on the procurement model that's in place for the mainframe. The mainframe is a mix of fixed and variable costs, so it's essential to understand where variability lies within the model so you can save money.
- *Align with the mainframe refresh schedule:* Coordinate your POC workload selection with your organization's mainframe refresh schedule. Prioritize workloads that reduce mainframe usage. This strategy aims to sync with hardware updates for optimal financial and operational efficiency.

## Evaluate technical partners

Choosing the right partners for participation in the POC and any following migration phases is a critical step that significantly affects the project's duration, effort, and cost.

**Ensure continuity with your implementation partners.** It's important to work with the same partners from start to finish. They can be from your company, or they can be outside vendors. Maintain the same team so you can use their POC findings for later stages. Changing partners can cause problems because the new team isn't familiar with the completed work.

**Use institutional knowledge.** Work with teams that already know your systems. These people, your employees or employees from other companies, know your system best. They can help plan the POC, pick the right tests, and determine the success criteria. Their knowledge helps ensure that the POC works for your business.

## Define technical criteria

From a technical standpoint, it's important to identify the technical components in a mainframe ecosystem that present complexity in a migration. Define a scope that represents those complexities. It's important to address complexity at the POC stage. You can eliminate risk in the subsequent modernization stages and identify solution

components for optimal design and architecture. When you select workloads for a migration POC, consider the following technical factors.

**Choose the right technology mix.** Choose workloads or subsets that include various technologies. For instance, assembler programs might constitute a small part of the total codebase but they might be crucial because of their intensive usage. In this example, the solution must cater to the unique demands of assembler language and its usage.

**Select various compilation parameters.** Select programs with diverse compilation parameters, such as COBOL and PL/I, to ensure minimal compatibility issues. These parameters test the POC against potential compatibility issues. Rehosting, refactoring, or other modernization strategies are scrutinized for tool or platform efficacy and customization options.

**Identify capability usage.** Pay special attention to batch processes and job control language (JCL) use cases. Identify and understand the specific functions of mainframe utilities, like sort editors or file editors, that might not have direct equivalents in modern solutions. Identify these utilities to help avoid unexpected challenges. Workload programmers often take advantage of the various capabilities that these utilities provide. Not all capabilities provide a direct map to a modernized solution.

**List integration patterns to test.** It's important to test integration patterns so you can ensure that the modernized workload performs the same as or better than the legacy environment. Determine the most important factors to validate in the POC phase based on the usage, criticality, and unique nature of the integration. Test the crucial integration patterns, such as patterns that are used among various workloads, datasets, and systems, including MQ with CICS, batch processes, and secure file transfers.

**Understand utilities and packaged software.** You need to migrate essential mainframe utilities and packaged software to Azure-based equivalents. Determine first-party or third-party solutions that can replicate or replace current mainframe functionalities. For example, the mainframe workloads and data might adhere to security rules that are set up within Resource Access Control Facility (RACF) and Access Control Facility 2 (ACF2), among others. When these workloads are hosted on Azure, the security requirements need to be adapted to adhere to Microsoft Entra ID. Consider the following utilities and services:

- Azure DevOps
- Azure Monitor
- Content storage
- Distribution
- Microsoft Entra ID

- Printers
- Reports
- Schedulers

**Define deliverables.** Define the deliverables to provide at the end of the POC. We recommend the following deliverables.

- *An architecture overview:* Provide a detailed architecture model with a deployment view. Ideally, you should align the model with established Azure architecture patterns. For an example overview, see [Azure mainframe and midrange architecture concepts and patterns](#).
- *Azure landing zone artifacts:* Include scripts and other resources for the Azure landing zone that you can repurpose for broader use.
- *Modernized code and setup:* Host the updated source code on GitHub. Include the setup scripts, configuration details, and parameters.
- *Supporting documentation:* Deliver comprehensive documentation that includes nonfunctional requirements, design blueprints, and test plans. You should customize these factors to fit the scope of the POC.

## Design the target environment

As you prepare to migrate mainframe workloads, data, and utilities to Azure, carefully select and design the appropriate Azure services. Azure offers a comprehensive yet flexible environment that you can use to tailor your target architecture. Implement a consumption-based model to dynamically scale your workload and meet business demands.

When you adopt an Azure pay-as-you-go model, consider the following factors.

**Performance.** Mainframe workloads typically have specific performance requirements. The performance requirements include processing large volumes of data within a specific time frame and accommodating a high volume of concurrent users. Define these performance benchmarks in your POC to ensure that the Azure solution meets or exceeds the legacy system's performance and user experience. For instance, validate the handling of heavy batch processing, simultaneous user access, and low-latency transactions.

**Scale.** In the POC phase, verify that the proposed solution can scale vertically (increasing the power of existing instances) and horizontally (adding more instances). Scaling

enables you to evaluate how the solution manages increased loads and transaction volumes in conjunction with the solution patterns and products that you consider.

**File-processing patterns.** Identify and test various file-processing patterns during the POC. You should assess the file volume, size, and types, such as the virtual storage access method (VSAM), generation data groups (GDGs), physical sequential datasets, and key-sequenced datasets (KSDSs). Also assess file operations, like sequential and keyed reads. For example, determine how the Azure solution manages and catalogs GDG files in a cloud environment. Similarly, if you migrate databases, test for database operation patterns and capabilities.

**Microsoft support.** The Microsoft Legacy Modernization team has Azure Core Engineering (ACE) resources, such as Global Black Belts, Data Ninjas, and Cloud Solution Architects. For a solution platform design, you can use the [mainframe landing zone accelerator](#). The Microsoft Mainframe Modernization team can also help to define an architecture that's specific to a solution path. Use their guidance to accelerate deployment and improve the solution in Azure.

## Define POC exit criteria

It's important to define success criteria for a POC so you can determine whether the POC achieves its goals and objectives. Success criteria are measurable and specific indicators that you use to evaluate the effectiveness of the POC.

Define success criteria to identify key performance indicators (KPIs). These criteria help to gauge the effectiveness of the POC. They also serve as a benchmark to compare the project's alignment with business goals and stakeholder expectations. To design your success and exit criteria, consider the following guidance.

- *Verify deliverables:* Review, finalize, and accept all source code, configuration files, setup parameters, and documentation. Examples include the projected architecture, Azure landing zone scripts, design documents, nonfunctional requirements, and test plans.
- *Finalize the Azure target environment:* Confirm that the Azure environment is fully defined and prepared for the migration and implementation process.
- *Validate the assessment roadmap:* Review the recommendations from the assessment phase. Adjust the recommendations as necessary based on the POC results.
- *Establish workload-specific KPIs:* Set explicit success criteria for workload performance, including response times, job execution durations, and other critical

technical KPIs.

- *Document migration insights:* Record all findings, challenges, and strategies that are related to an effective data migration, based on the lessons learned from the POC.
- *Evaluate project parameters:* Reassess and confirm the project's estimation, timeline, and the composition of the teams, including the customers, system integrators, partners, and cloud providers. Amend the business case accordingly.
- *Review technical resources:* Update a list of necessary software, tools, and accelerators in accordance with the initial assessment phase recommendations and any new insights that are gained during the POC.
- *Present an executive summary:* Prepare and deliver a comprehensive presentation of the POC outcomes and lessons learned. Set expectations about the upcoming phases for senior management or the steering committee.

## Contributors

*This article is maintained by Microsoft. It was originally written by the following contributors.*

Principal authors:

- [Shelby Howard](#) | Senior Specialist, Global Black Belt Mainframe and Midrange Modernization Solutions
- [Venkataraman Ramakrishnan](#) | Senior Technical Program Manager, Azure Core Engineering Mainframe and Midrange Modernization Solutions

*To see non-public LinkedIn profiles, sign in to LinkedIn.*

## Next steps

- [Mainframe application migration strategies](#)
- To provide feedback or receive assistance, contact the [Microsoft Legacy Modernization ACE team](#). Along with Microsoft Certified Partners, the Legacy Modernization ACE team works with the world's top public and private sector organizations to modernize their legacy platforms.

For more information about partners, see [Migration partners](#).

## Related resources

- [Azure mainframe and midrange architecture concepts and patterns](#)

# Mainframe rehosting on Azure virtual machines

Article • 04/25/2022

Migrating workloads from mainframe environments to the cloud enables you to modernize your infrastructure and often save on costs. Many workloads can be transferred to Azure with only minor code changes, such as updating the names of databases.

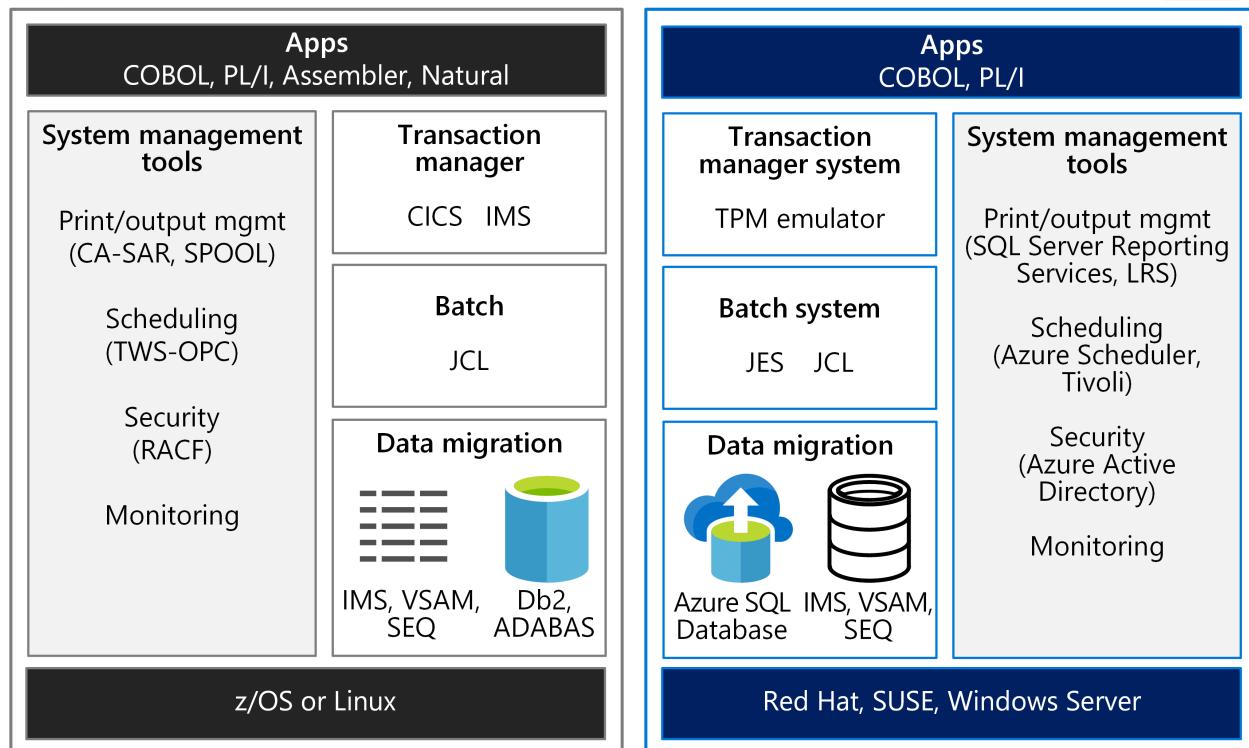
Generally speaking, the term *mainframe* means a large computer system. Specifically, the vast majority currently in use are IBM System Z servers or IBM plug-compatible systems that run MVS, DOS, VSE, OS/390, or z/OS.

An Azure virtual machine (VM) is used to isolate and manage the resources for a specific application on a single instance. Mainframes such as IBM z/OS use Logical Partitions (LPARS) for this purpose. A mainframe might use one LPAR for a CICS region with associated COBOL programs, and a separate LPAR for IBM Db2 database. A typical [n-tier application on Azure](#) deploys Azure VMs into a virtual network that can be segmented into subnets for each tier.

Azure VMs can run mainframe emulation environments and compilers that support lift-and-shift scenarios. Development and testing are often among the first workloads to migrate from a mainframe to an Azure dev/test environment. Common server components that you can emulate include online transaction process (OLTP), batch, and data ingestion systems as the following figure shows.



Mainframe



Some mainframe workloads can be migrated to Azure with relative ease, while others can be rehosted on Azure using a partner solution. For detailed guidance about choosing a partner solution, the [Azure Mainframe Migration center](#) can help.

## Mainframe migration

Rehost, rebuild, replace, or retire? IaaS or PaaS? To determine the right migration strategy for your mainframe application, see the [Mainframe migration](#) guide in the Azure Architecture Center.

## Micro Focus rehosting platform

Micro Focus Enterprise Server is one of the largest mainframe rehosting platforms available. You can use it run your z/OS workloads on a less expensive x86 platform on Azure.

To get started:

- [Install Enterprise Server and Enterprise Developer on Azure](#)
- [Set up CICS BankDemo for Enterprise Developer on Azure](#)
- [Run Enterprise Server in a Docker Container on Azure](#)

## TmaxSoft OpenFrame on Azure

TmaxSoft OpenFrame is a popular mainframe rehosting solution used in lift-and-shift scenarios. An OpenFrame environment on Azure is suitable for development, demos, testing, or production workloads.

To get started:

- [Get started with TmaxSoft OpenFrame](#)
- [Download the ebook ↗](#)

## IBM zD&T 12.0

IBM Z Development and Test Environment (IBM zD&T) sets up a non-production environment on Azure that you can use for development, testing, and demos of z/OS-based applications.

The emulation environment on Azure can host different kinds of Z instances through Application Developers Controlled Distributions (ADCDs). You can run zD&T Personal Edition, zD&T Parallel Sysplex, and zD&T Enterprise Edition on Azure and Azure Stack.

To get started:

- [Set up IBM zD&T 12.0 on Azure](#)
- [Set up ADCD on zD&T](#)

## IBM DB2 pureScale on Azure

The IBM DB2 pureScale environment provides a database cluster for Azure. It's not identical to the original environment, but it delivers similar availability and scale as IBM DB2 for z/OS running in a Parallel Sysplex setup.

To get started, see [IBM DB2 pureScale on Azure](#).

## Considerations

When you migrate mainframe workloads to Azure infrastructure as a service (IaaS), you can choose from several types of on-demand, scalable computing resources, including Azure VMs. Azure offers a range of [Linux](#) and [Windows](#) VMs.

## Compute

Azure compute power compares favorably to a mainframe's capacity. If you're thinking of moving a mainframe workload to Azure, compare the mainframe metric of one

million instructions per second (MIPS) to virtual CPUs.

Learn how to [move mainframe compute to Azure](#).

## High availability and failover

Azure offers commitment-based service-level agreements (SLAs). Multiple-nines availability is the default, and SLAs can be optimized with local or geo-based replication of services. The full [Azure SLA](#) explains the guaranteed availability of Azure as a whole.

With Azure IaaS such as a VM, specific system functions provide failover support—for example, failover clustering instances and availability sets. When you use Azure platform as a service (PaaS) resources, the platform handles failover automatically. Examples include [Azure SQL Database](#) and [Azure Cosmos DB](#).

## Scalability

Mainframes typically scale up, while cloud environments scale out. Azure offers a range of [Linux](#) and [Windows](#) sizes to meet your needs. The cloud also scales up or down to match exact user specifications. Compute power, storage, and services [scale](#) on demand under a usage-based billing model.

## Storage

In the cloud, you have a range of flexible, scalable storage options, and you pay only for what 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.

Learn how to [move mainframe storage to Azure](#).

## Backup and recovery

Maintaining your own disaster recovery site can be an expensive proposition. Azure has easy-to-implement and cost-effective options for [backup](#), [recovery](#), and [redundancy](#) at local or regional levels, or via geo-redundancy.

# Azure Government for mainframe migrations

Many public sector entities would love to move their mainframe applications to a more modern, flexible platform. Microsoft Azure Government is a physically separated

instance of the global Microsoft Azure platform—packaged for federal, state, and local government systems. It provides world-class security, protection, and compliance services specifically for United States government agencies and their partners.

Azure Government earned a Provisional Authority to Operate (P-ATO) for FedRAMP High Impact for systems that need this type of environment.

To get started, download [Microsoft Azure Government cloud for mainframe applications](#).

## Next steps

Ask our [partners](#) to help you migrate or rehost your mainframe applications.

See also:

- [White papers about mainframe topics](#)
- [Mainframe migration](#)
- [Troubleshooting](#)
- [Demystifying mainframe to Azure migration](#)

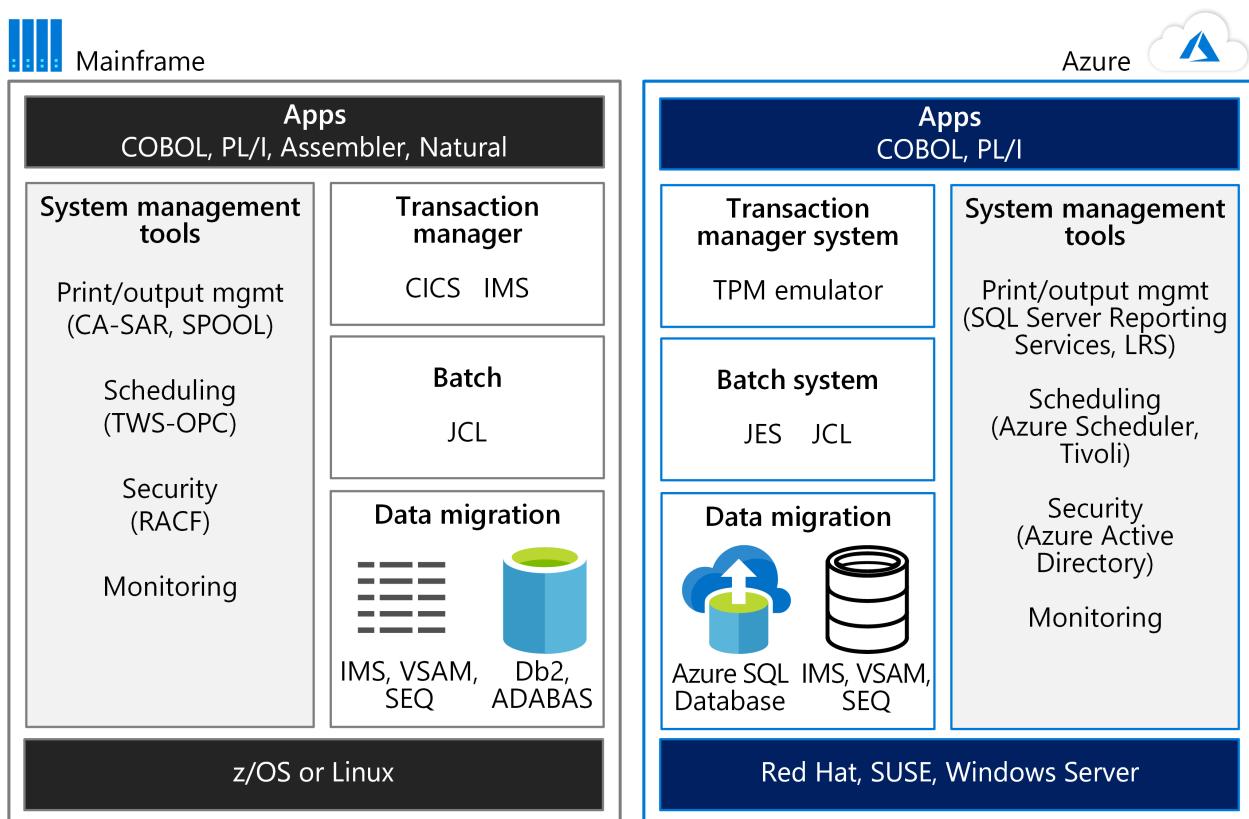
# Move mainframe compute to Azure

Article • 03/23/2021

Mainframes have a reputation for high reliability and availability and continue to be the trusted backbone of many enterprises. They're often thought to have nearly limitless scalability and computing power as well. However, some enterprises have outgrown the capability of the largest available mainframes. If this sounds like you, Azure offers agility, reach, and infrastructure savings.

To run mainframe workloads on Microsoft Azure, you need to know how your mainframe's compute capabilities compare to Azure. Based on an IBM z14 mainframe (the most current model as of this writing), this article tells you how to get comparable results on Azure.

To get started, consider the environments side by side. The following figure compares a mainframe environment for running applications to an Azure hosting environment.



The power of mainframes is often used for online transaction processing (OLTP) systems that handle millions of updates for thousands of users. These applications often use software for transaction processing, screen handling, and form entry. They may use a Customer Information Control System (CICS), Information Management System (IMS), or Transaction Interface Package (TIP).

As the figure shows, a TPM emulator on Azure can handle CICS and IMS workloads. A batch system emulator on Azure performs the role of Job Control Language (JCL). Mainframe data is migrated to Azure databases, such as Azure SQL Database. Azure services or other software hosted in Azure Virtual Machines can be used for system management.

## Mainframe compute at a glance

In the z14 mainframe, processors are arranged in up to four *drawers*. A *drawer* is simply a cluster of processors and chipsets. Each drawer can have six active central processor (CP) chips, and each CP has 10 system controller (SC) chips. In Intel x86 terminology, there are six sockets per drawer, 10 cores per socket, and four drawers. This architecture provides the rough equivalent of 24 sockets and 240 cores, maximum, for a z14.

The fast z14 CP has a 5.2 GHz clock speed. Typically, a z14 is delivered with all the CPs in the box. They're activated as needed. A customer is commonly charged for at least four hours of compute time per month despite actual usage.

A mainframe processor can be configured as one of the following types:

- General Purpose (GP) processor
- System z Integrated Information Processor (zIIP)
- Integrated Facility for Linux (IFL) processor
- System Assist Processor (SAP)
- Integrated Coupling Facility (ICF) processor

## Scaling mainframe compute up and out

IBM mainframes offer the ability to scale up to 240 cores (the current z14 size for a single system). Additionally, IBM mainframes can scale out through a feature called the Coupling Facility (CF). The CF allows multiple mainframe systems to access the same data simultaneously. Using the CF, the mainframe Parallel Sysplex technology groups mainframe processors in clusters. When this guide was written, the Parallel Sysplex feature supported 32 groupings of 64 processors each. Up to 2,048 processors can be grouped in this manner to scale out compute capacity.

A CF allows the compute clusters to share data with direct access. It's used for locking information, cache information, and the list of shared data resources. A Parallel Sysplex using one or more CFs can be thought of as a "shared everything" scale-out compute cluster. For more information about these features, see [Parallel Sysplex on IBM Z](#) on the IBM website.

Applications can use these features to provide both scale-out performance and high availability. For information about how CICS can use Parallel Sysplex with CF, download the [IBM CICS and the Coupling Facility: Beyond the Basics](#) redbook.

## Azure compute at a glance

Some people mistakenly think that Intel-based servers aren't as powerful as mainframes. However, the new core-dense, Intel-based systems have as much compute capacity as mainframes. This section describes the Azure infrastructure-as-a-service (IaaS) options for computing and storage. Azure provides platform-as-a-service (PaaS) options as well, but this article focuses on the IaaS choices that provide comparable mainframe capacity.

Azure Virtual Machines provide compute power in a range of sizes and types. In Azure, a virtual CPU (vCPU) roughly equates to a core on a mainframe.

Currently, the range of Azure Virtual Machine sizes provides from 1 to 128 vCPUs. Virtual machine (VM) types are optimized for particular workloads. For example, the following list shows the VM types (current as of this writing) and their recommended uses:

Size	Type and description
D-Series	General purpose with 64 vCPU and up to 3.5-GHz clock speed
E-Series	Memory optimized with up to 64 vCPUs
F-Series	Compute optimized with up to 64 vCPUs and 3..7 GHz clock speed
H-Series	Optimized for high-performance computing (HPC) applications
L-Series	Storage optimized for high-throughput applications backed by databases such as NoSQL
M-Series	Largest compute and memory optimized VMs with up to 128 vCPUs

For details about available VMs, see [Virtual Machine series](#).

A z14 mainframe can have up to 240 cores. However, z14 mainframes almost never use all the cores for a single application or workload. Instead, a mainframe segregates workloads into logical partitions (LPARs), and the LPARs have ratings—MIPS (Millions of Instructions Per Second) or MSU (Million Service Unit). When determining the comparable VM size needed to run a mainframe workload on Azure, factor in the MIPS (or MSU) rating.

The following are general estimates:

- 150 MIPS per vCPU
- 1,000 MIPS per processor

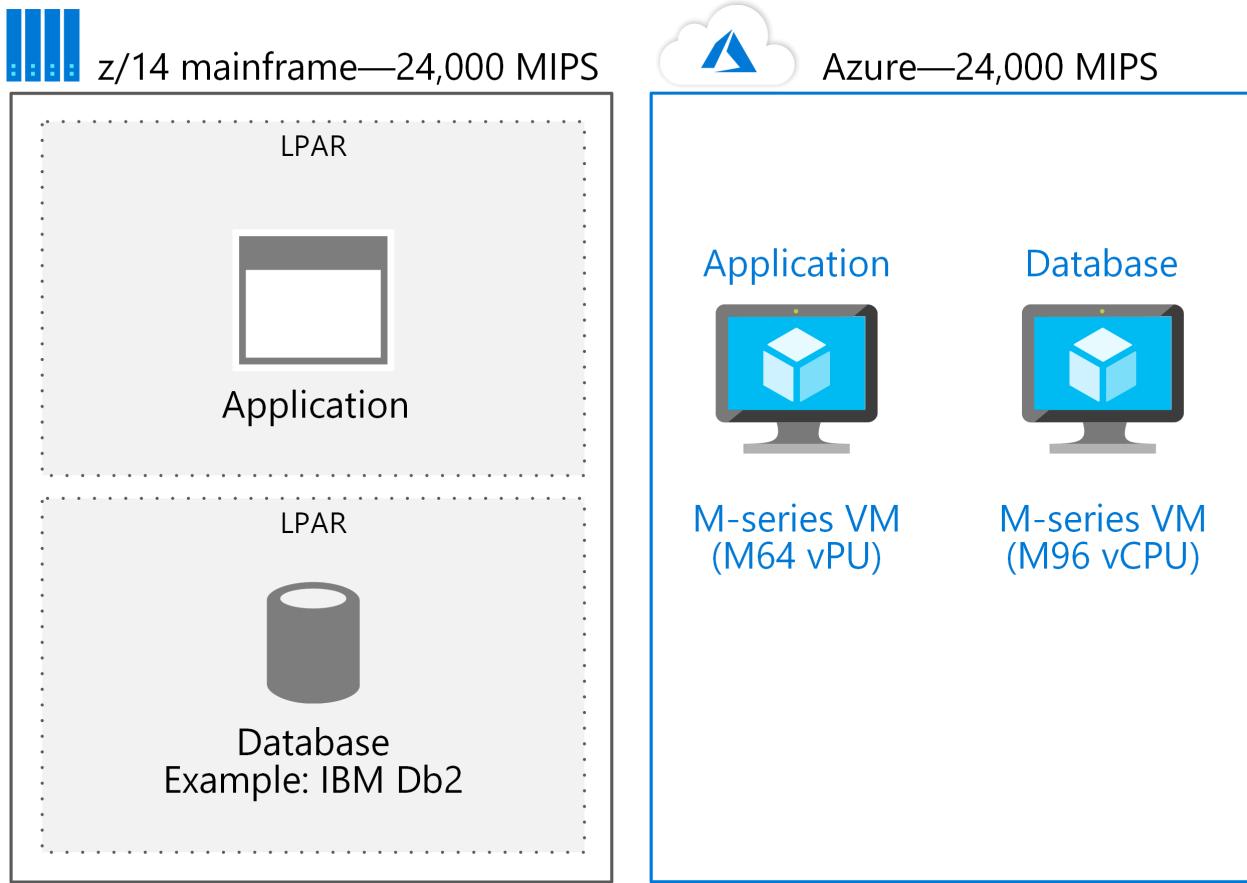
To determine the correct VM size for a given workload in an LPAR, first optimize the VM for the workload. Then determine the number of vCPUs needed. A conservative estimate is 150 MIPS per vCPU. Based on this estimate, for example, an F-series VM with 16 vCPUs could easily support an IBM Db2 workload coming from an LPAR with 2,400 MIPS.

## Azure compute scale-up

The M-series VMs can scale up to 128 vCPUs (at the time this article was written). Using the conservative estimate of 150 MIPS per vCPU, the M-series VM equates to about 19,000 MIPS. The general rule for estimating MIPS for a mainframe is 1,000 MIPS per processor. A z14 mainframe can have up to 24 processors and provide about 24,000 MIPS for a single mainframe system.

The largest single z14 mainframe has approximately 5,000 MIPS more than the largest VM available in Azure. Yet it's important to compare how workloads are deployed. If a mainframe system has both an application and a relational database, they're typically deployed on the same physical mainframe—each in its own LPAR. The same solution on Azure is often deployed using one VM for the application and a separate, suitably sized VM for the database.

For example, if a M64 vCPU system supports the application, and a M96 vCPU is used for the database, approximately 150 vCPUs are needed—or about 24,000 MIPS as the following figure shows.



The approach is to migrate LPARs to individual VMs. Then Azure easily scales up to the size needed for most applications that are deployed on a single mainframe system.

## Azure compute scale-out

One of the advantages of an Azure-based solution is the ability to scale out. Scaling makes nearly limitless compute capacity available to an application. Azure supports multiple methods to scale out compute power:

- **Load balancing across a cluster.** In this scenario, an application can use a [load balancer](#) or resource manager to spread out the workload among multiple VMs in a cluster. If more compute capacity is needed, additional VMs are added to the cluster.
- **Virtual machine scale sets.** In this burst scenario, an application can scale to additional [compute resources](#) based on VM usage. When demand falls, the number of VMs in a scale set can also go down, ensuring efficient use of compute power.
- **PaaS scaling.** Azure PaaS offerings scale compute resources. For example, [Azure Service Fabric](#) allocates compute resources to meet increases in the volume of requests.

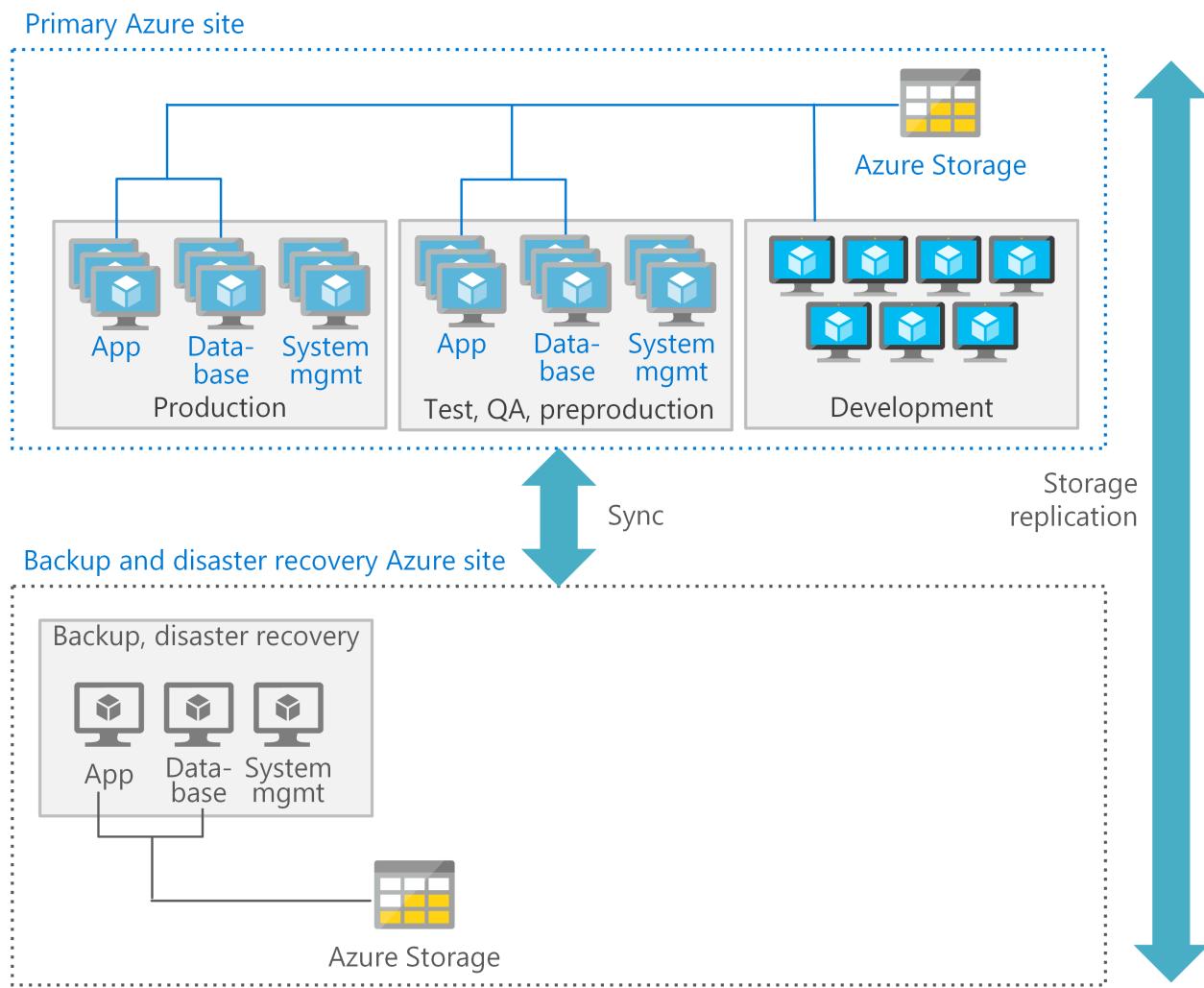
- **Kubernetes clusters.** Applications on Azure can use [Kubernetes clusters](#) for compute services for specified resources. Azure Kubernetes Service (AKS) is a managed service that orchestrates Kubernetes nodes, pools, and clusters on Azure.

To choose the right method for scaling out compute resources, it's important to understand how Azure and mainframes differ. The key is how—or if—data is shared by compute resources. In Azure, data (by default) is not typically shared by multiple VMs. If data sharing is required by multiple VMs in a scale-out compute cluster, the shared data must reside in a resource that supports this functionality. On Azure, data sharing involves storage as the following section discusses.

## Azure compute optimization

You can optimize each tier of processing in an Azure architecture. Use the most suitable type of VMs and features for each environment. The following figure shows one potential pattern for deploying VMs in Azure to support a CICS application that uses Db2. In the primary site, the production, preproduction, and testing VMs are deployed with high availability. The secondary site is for backup and disaster recovery.

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.



## Next steps

- Mainframe migration
- Mainframe rehosting on Azure Virtual Machines
- Move mainframe storage to Azure

## IBM resources

- Parallel Sysplex on IBM Z ↗
- IBM CICS and the Coupling Facility: Beyond the Basics ↗
- Creating required users for a Db2 pureScale Feature installation ↗
- Db2icrt - Create instance command ↗
- Db2 pureScale Clustered Database Solution ↗
- IBM Data Studio ↗

## Azure Government

- Microsoft Azure Government cloud for mainframe applications ↗
- Microsoft and FedRAMP ↗

## More migration resources

- [Azure Virtual Data Center Lift and Shift Guide ↗](#)
- [GlusterFS iSCSI ↗](#)

# Move mainframe storage to Azure

Article • 10/15/2021

To run mainframe workloads on Microsoft Azure, you need to know how your mainframe's capabilities compare to Azure. The massively scalable storage resources can help organizations begin to modernize without abandoning the applications they rely on.

Azure provides mainframe-like features and storage capacity that is comparable to IBM z14-based systems (the most current model as of this writing). This article tells you how to get comparable results on Azure.

## Mainframe storage at a glance

The IBM mainframe characterizes storage in two ways. The first is a direct access storage device (DASD). The second is sequential storage. For managing storage, the mainframe provides the Data Facility Storage Management Subsystem (DFSMS). It manages data access to the various storage devices.

[DASD](#) refers to a separate device for secondary (not in-memory) storage that allows a unique address to be used for direct access of data. Originally, the term DASD applied to spinning disks, magnetic drums, or data cells. However, now the term can also apply to solid-state storage devices (SSDs), storage area networks (SANs), network attached storage (NAS), and optical drives. For the purposes of this document, DASD refers to spinning disks, SANs, and SSDs.

In contrast to DASD storage, sequential storage on a mainframe refers to devices like tape drives where data is accessed from a starting point, then read or written in a line.

Storage devices are typically attached using a fiber connection (FICON) or are accessed directly on the mainframe's IO bus using [HiperSockets](#), an IBM technology for high-speed communications between partitions on a server with a hypervisor.

Most mainframe systems separate storage into two types:

- *Online storage* (also known as hot storage) is needed for daily operations. DASD storage is usually used for this purpose. However, sequential storage, such as daily tape backups (logical or physical), can also be used for this purpose.
- *Archive storage* (also known as cold storage) is not guaranteed to be mounted at a given time. Instead, it is mounted and accessed as needed. Archive storage is often implemented using sequential tape backups (logical or physical) for storage.

# Mainframe versus IO latency and IOPS

Mainframes are often used for applications that require high performance IO and low IO latency. They can do this using the FICON connections to IO devices and HiperSockets.

When HiperSockets are used to connect applications and devices directly to a mainframe's IO channel, latency in the microseconds can be achieved.

## Azure storage at a glance

Azure infrastructure-as-a-service ([IaaS](#)) options for storage provide comparable mainframe capacity.

Microsoft offers petabytes worth of storage for applications hosted in Azure, and you have several storage options. These range from SSD storage for high performance to low-cost blob storage for mass storage and archives. Additionally, Azure provides a data redundancy option for storage—something that takes more effort to set up in a mainframe environment.

Azure storage is available as [Azure Disks](#), [Azure Files](#), and [Azure Blobs](#) as the following table summarizes. Learn more about [when to use each](#).

Type	Description	Use when you want to:
Azure Files	Provides an SMB interface, client libraries, and a <a href="#">REST</a> interface that allows access from anywhere to stored files.	<ul style="list-style-type: none"><li>Lift and shift an application to the cloud when the application uses the native file system APIs to share data between it and other applications running in Azure.</li><li>Store development and debugging tools that need to be accessed from many VMs.</li></ul>
Azure Blobs	Provides client libraries and a <a href="#">REST</a> interface that allows unstructured data to be stored and accessed at a massive scale in block blobs. Also supports <a href="#">Azure Data Lake Storage Gen2</a> for enterprise big data analytics solutions.	<ul style="list-style-type: none"><li>Support streaming and random-access scenarios in an application.</li><li>Have access to application data from anywhere.</li><li>Build an enterprise data lake on Azure and perform big data analytics.</li></ul>

Type	Description	Use when you want to:
Azure Disks	Provides client libraries and a <a href="#">REST</a> interface that allows data to be persistently stored and accessed from an attached virtual hard disk.	<ul style="list-style-type: none"> <li>Lift and shift applications that use native file system APIs to read and write data to persistent disks.</li> <li>Store data that is not required to be accessed from outside the VM to which the disk is attached.</li> </ul>

## Azure hot (online) and cold (archive) storage

The type of storage for a given system depends on the requirements of the system, including storage size, throughput, and IOPS. For DASD-type storage on a mainframe, applications on Azure typically use Azure Disks drive storage instead. For mainframe archive storage, blob storage is used on Azure.

SSDs provide the highest storage performance on Azure. The following options are available (as of the writing of this document):

Type	Size	IOPS
Ultra SSD	4 GB to 64 TB	1,200 to 160,000 IOPS
Premium SSD	32 GB to 32 TB	12 to 15,000 IOPS
Standard SSD	32 GB to 32 TB	12 to 2,000 IOPS

Blob storage provides the largest volume of storage on Azure. In addition to storage size, Azure offers both managed and unmanaged storage. With managed storage, Azure takes care of managing the underlying storage accounts. With unmanaged storage, the user takes responsibility for setting up Azure storage accounts of the appropriate size to meet the storage requirements.

## Next steps

- [Mainframe migration](#)
- [Mainframe rehosting on Azure Virtual Machines](#)
- [Move mainframe compute to Azure](#)
- [Deciding when to use Azure Blobs, Azure Files, or Azure Disks](#)
- [Standard SSD Managed Disks for Azure VM workloads](#)

## IBM resources

- [Parallel Sysplex on IBM Z](#)
- [IBM CICS and the Coupling Facility: Beyond the Basics](#)
- [Creating required users for a Db2 pureScale Feature installation](#)
- [Db2icrt - Create instance command](#)
- [Db2 pureScale Clustered Database Solution](#)
- [IBM Data Studio](#)

## Azure Government

- [Microsoft Azure Government cloud for mainframe applications](#)
- [Microsoft and FedRAMP](#)

## More migration resources

- [Azure Virtual Data Center Lift and Shift Guide](#)
- [GlusterFS iSCSI](#)

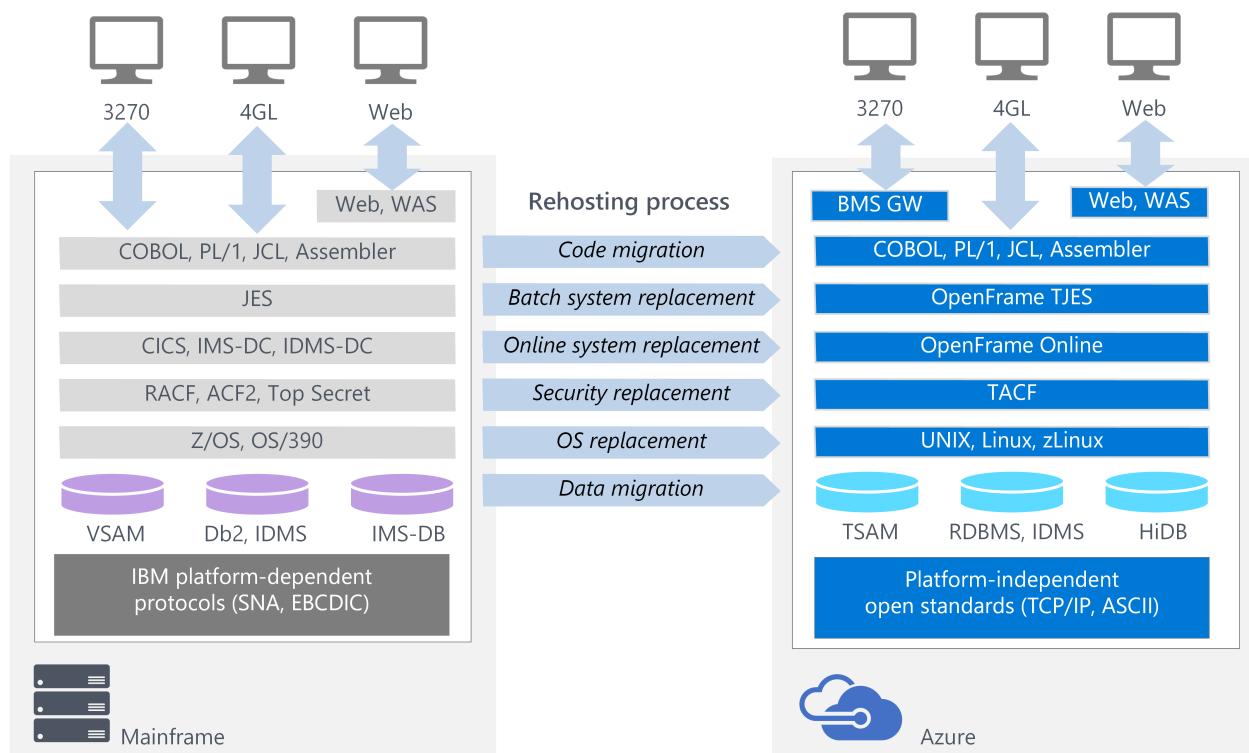
# Get started with TmaxSoft OpenFrame on Azure

Article • 03/23/2021

Take your existing mainframe assets and move them to Microsoft Azure using TmaxSoft OpenFrame. This popular rehosting solution creates an emulation environment on Azure, enabling you to quickly migrate applications. No reformatting is required.

## OpenFrame rehosting environment

Set up an OpenFrame environment on Azure for development, demos, testing, or production workloads. As the following figure shows, OpenFrame includes multiple components that create the mainframe emulation environment on Azure. For example, OpenFrame online services replace the mainframe middleware such as IBM Customer Information Control System (CICS). OpenFrame Batch, with its TJES component, replaces the IBM mainframe's Job Entry Subsystem (JES).



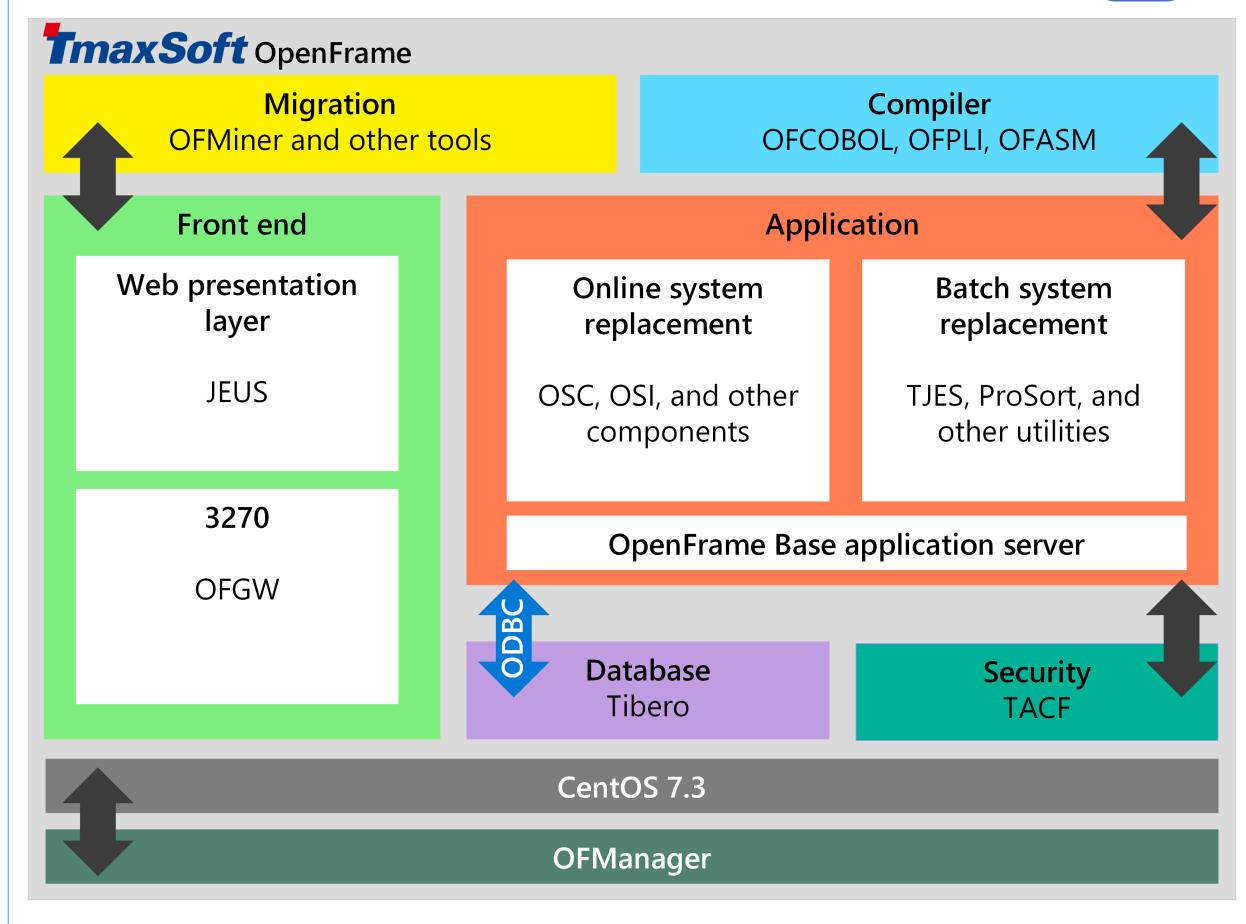
### ⓘ Note

To run the OpenFrame environment on Azure, you must have a valid product license or trial license from TmaxSoft.

# OpenFrame components

The following components are part of the OpenFrame environment on Azure:

- **Migration tools** including OFMiner, a solution that analyzes the mainframes assets and then migrates them to Azure.
- **Compilers**, including OFCOBOL, a compiler that interprets the mainframe's COBOL programs; OFPLI, which interprets the mainframe's PL/I programs; and OFASM, a compiler that interprets the mainframe's assembler programs.
- **Front end** components, including Java Enterprise User Solution (JEUS ), a web application server that is certified for Java Enterprise Edition 6.OFGW, and the OpenFrame gateway component that provides a 3270 listener.
- **Application** environment. OpenFrame Base is the middleware that manages the entire system. OpenFrame Server Type C (OSC) replaces the mainframe's middleware and IBM CICS.
- **Relational database**, such as Tibero (shown), Oracle Database, Microsoft SQL Server, IBM Db2, or MySQL. The OpenFrame applications use Open Database Connectivity (ODBC) protocol to communicate with the database.
- **Security** via TACF, a service module that controls user access to systems and resources.
- **OFManager** is a solution that provides OpenFrame's operation and management functions in the web environment.



## Next steps

- Install TmaxSoft OpenFrame on Azure

# AIX UNIX on-premises to Azure Linux migration

Azure NetApp Files

Azure Site Recovery

Azure SQL Database

Azure Virtual Machines

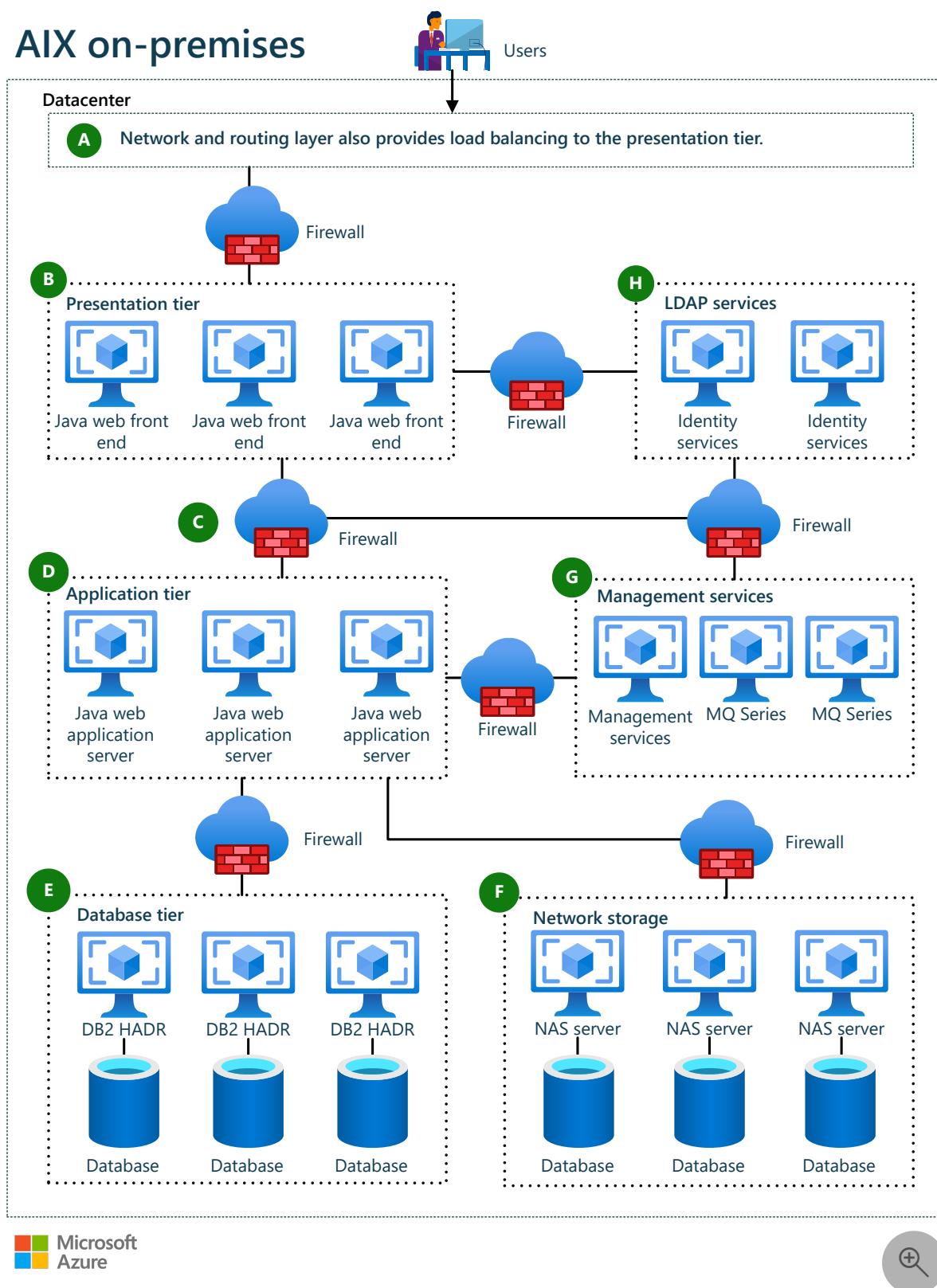
Azure Virtual Network

This solution describes a migration from an IBM AIX Unix platform to Red Hat Enterprise Linux (RHEL) in Azure. The real-world example was a Health and Human Services application for a large customer. Low transaction time and latency were important requirements for both the legacy and the Azure systems. A key functionality is storing customer information in a database that links into a network file store containing related graphical images. Azure addresses this need with Azure NetApp Files.

## Architecture

The following diagram shows the pre-migration, on-premises AIX legacy system architecture:

# AIX on-premises

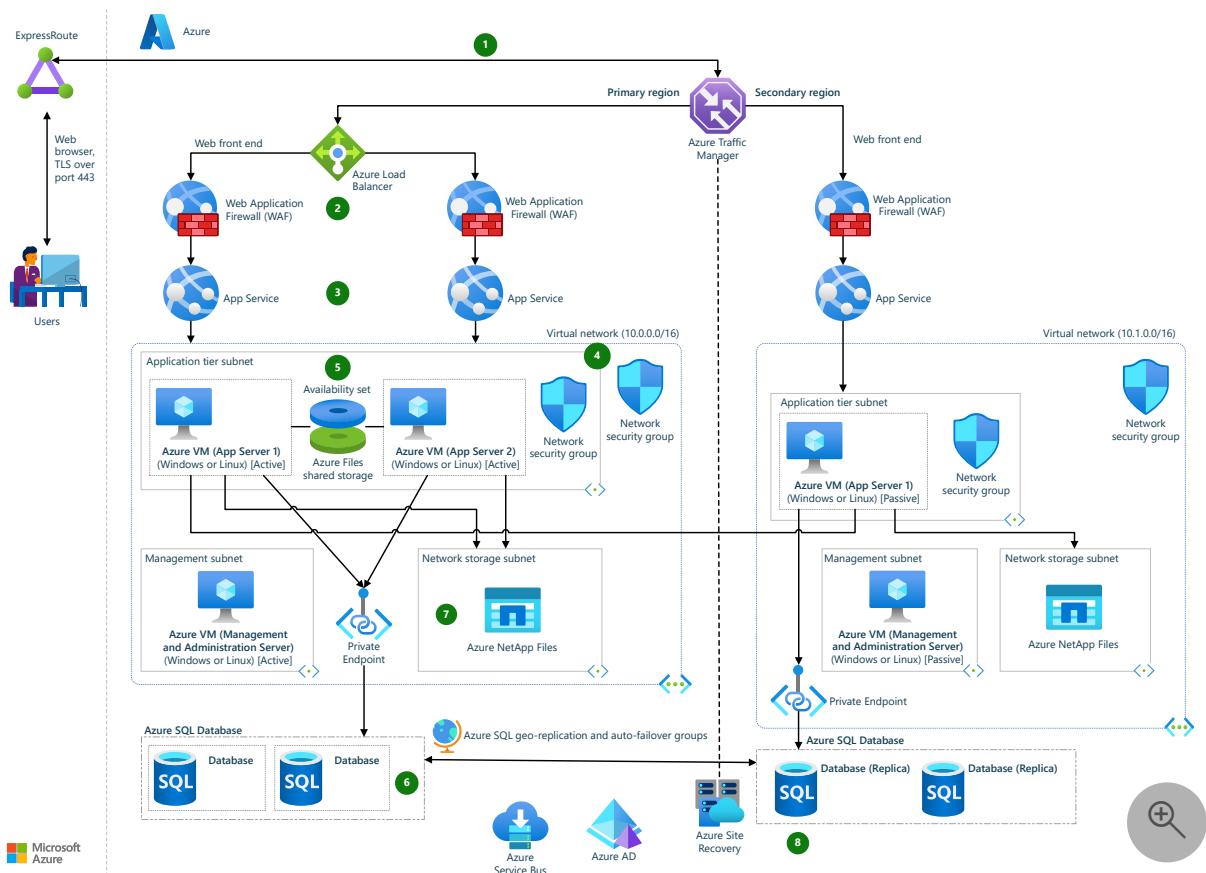


Download a [Visio file](#) of this architecture.

- Network appliances provide an extensive network routing and load-balancing layer (A).
- The presentation tier (B) uses three Java web front-end machines in their own subnet, which segments network traffic by firewalls.

- Firewalls (C) provide network boundaries between all participating tiers and subsystems. While firewalls are effective, they're also an administrative burden.
- The system provides user requests to the application tier (D), which has three web application servers.
- The application tier calls into the DB2 database and the network attached storage (NAS):
  - The database (E) is DB2 on AIX. Three DB2 servers are configured in a HA/DR cluster.
  - The application stores binary objects like pictures and PDFs for customers and users in a NAS subsystem (F).
- Management and administration servers and the MQ servers (G) are in their own subnet, segmented by firewalls.
- Lightweight Directory Access Protocol (LDAP) identity management services (H) are in their own subnet, segmented by firewalls.

The following diagram shows the Azure RHEL post-migration system architecture:



Download a [Visio file](#) of this architecture.

# Dataflow

1. Traffic into the Azure system routes through Azure ExpressRoute and Azure Traffic Manager:
  - ExpressRoute provides a secure, reliable private connection to Azure virtual networks. ExpressRoute connects to Azure with low latency, high reliability and speed, and bandwidths up to 100 Gbps.
  - Traffic Manager distributes the public-facing application traffic across Azure regions.
2. A network management layer provides endpoint security, routing, and load-balancing services. This layer uses Azure Load Balancer and Azure Web Application Firewall.
3. Azure App Service serves as the presentation tier. App Service is a platform-as-a-service (PaaS) layer for .NET or Java applications. You can configure App Service for availability and scalability within and across Azure regions.
4. The solution encapsulates each application tier in its own virtual network, segmented with network security groups.
5. [Availability sets](#) and shared Azure Storage provide HA and scalability for virtual machines (VMs) at the application tier level. Application cluster servers share transaction state, and scale up VMs as necessary.
6. The application uses a [private endpoint](#) connection to store and access data in Azure SQL Database. SQL Database runs in a business continuity configuration, which provides geo-replication and autofailover groups for automatic and cross-geographic BCDR.
7. Azure NetApp Files provides a shared NAS, with fast access to binary data and replication to the secondary region.
8. The secondary region provides BCDR with the following components:
  - Azure Site Recovery backs up VM images for DR failover in an active-passive configuration. Site Recovery creates consistent VM image replicas in the secondary region and keeps the VM images in sync.
  - SQL Database business continuity configuration keeps the database transactions consistent. SQL Database provisions replica databases and keeps them in sync with synchronous or asynchronous data replication.

The system also contains the following components:

- One or more VMs in the Management virtual network provide management and administration functionality.
- Azure Service Bus implements the MQ Series infrastructure and provides message queue services for the applications. For more information on migrating from MQ Series to Azure Service Bus, see [Migrate from ActiveMQ to Azure Service Bus](#).
- Microsoft Entra ID provides identity and access management for all Azure entities and identities migrated from the legacy LDAP services.

## Components

- [Azure ExpressRoute](#) extends an on-premises network into Microsoft cloud services over a private connection, facilitated by a connectivity provider. ExpressRoute provides a secure, reliable private connection to the Azure system, with low latency and high speed and bandwidth.
- [Azure Traffic Manager](#) is a DNS-based traffic load balancer that distributes traffic across Azure regions, with high availability and quick responsiveness.
- [Azure Load Balancer](#) supports high availability by distributing incoming network traffic among backend VMs according to configured load-balancing rules and health probes. Load Balancer operates at layer 4 of the Open Systems Interconnection (OSI) model.
- [Azure Web Application Firewall](#) is a cloud-native WAF service that protects web apps against malicious attacks and common web vulnerabilities.
- [Azure App Service](#) is a fully managed web hosting service for quickly and easily deploying enterprise web apps for any platform on a scalable and reliable cloud infrastructure.
- [Azure Virtual Machines](#) is one of several Azure services that provide on-demand, scalable computing resources. With Azure VMs, you get the flexibility of virtualization without having to buy and maintain physical hardware.
  - [Azure SSD managed disks](#) are block-level storage volumes for Azure VMs.
  - [Azure virtual network interface cards \(NICs\)](#) let Azure VMs communicate with internet, Azure, and on-premises resources. You can add several virtual NICs to an Azure VM, so child VMs can have their own dedicated network interface devices and IP addresses.
- [Azure Virtual Network](#) is the fundamental building block for Azure private networks. Virtual Network lets many types of Azure resources, such as VMs,

securely communicate with each other, the internet, and on-premises networks.

Virtual Network offers Azure infrastructure benefits like scalability, availability, and isolation.

- [Azure Files](#) storage offers fully managed file shares in the cloud that are accessible via the industry-standard Server Message Block (SMB) protocol. Cloud and on-premises Windows, Linux, and macOS deployments can mount Azure file shares concurrently.
- [Azure SQL Database](#) is a fully managed database PaaS that always runs on the latest OS and stable SQL Server database engine version, with highest availability. SQL Database handles database management functions, such as upgrades, patching, backups, and monitoring, without user involvement.
- [Azure NetApp Files](#) offers enterprise-grade Azure file shares powered by NetApp. Azure NetApp Files makes it easy for enterprises to migrate and run complex, file-based applications with no code changes.
- [Azure Site Recovery](#) is an Azure-native DR service. Site Recovery deploys replication, failover, and recovery processes to help keep applications running during planned and unplanned outages.
- [Azure Service Bus](#) is a reliable cloud messaging service with simple hybrid integration.
- [Microsoft Entra ID](#) is Microsoft's cloud-based enterprise identity and access management service. Microsoft Entra single sign-on and multifactor authentication help users sign in and access resources, while protecting from cybersecurity attacks.

## Alternatives

[Azure App Service environments](#) are appropriate for application workloads that require high scale, isolation, and secure network access. This feature offers fully isolated and dedicated environments for securely running App Service apps at high scale. App Service environments can host the following types of apps:

- Linux web apps, as in the current example
- Windows web apps
- Docker containers
- Mobile apps
- Functions

# Scenario details

One distinct difference between the legacy system and the cloud implementation is in handling network segmentation. The legacy system segmented networks with firewalls. A cloud platform like Azure segments networks with virtual networks and network security groups that filter traffic based on several criteria.

Another difference between the systems is their high availability (HA) and disaster recovery (DR) models. In the legacy system, HA/DR primarily used backups, and to some extent used redundant servers in the same datacenter. This configuration provided modest DR, but almost no HA capabilities. Improving HA/DR was a key driver for moving to the Azure platform. Azure uses clustering, shared storage, and Azure Site Recovery to provide a high level of HA/DR.

## Potential use cases

Key drivers for moving from on-premises IBM AIX to RHEL in Azure might include the following factors:

- **Updated hardware and reduced costs.** On-premises, legacy hardware components continually go out of date and out of support. Cloud components are always up to date. Month-to-month costs can be less in the cloud.
- **Agile DevOps environment.** Deploying compliance changes in an on-premises AIX environment can take weeks. You might have to set up similar performance engineering environments many times to test changes. In an Azure cloud environment, you can set up user acceptance testing (UAT) and development environments in hours. You can implement changes through a modern, well-defined DevOps continuous integration and continuous delivery (CI/CD) pipeline.
- **Improved Business Continuity and Disaster Recovery (BCDR).** In on-premises environments, recovery time objectives (RTOs) can be long. In the example on-premises AIX environment, the RTO via traditional backups and restores was two days. Migrating to Azure reduced the RTO to two hours.

## Considerations

The following considerations, based on the [Microsoft Azure Well-Architected Framework](#), apply to this solution:

## Availability

- Azure NetApp Files can keep the file store in the secondary region updated with [Cross-region replication of Azure NetApp Files Volumes](#). This Azure feature provides data protection through cross-region volume replication. You can fail over critical applications if there is a region-wide outage. Cross-region volume replication is currently in preview.
- Application cluster servers scale up VMs as necessary, which increases availability within Azure regions.

## Operations

For proactive monitoring and management, consider using [Azure Monitor](#) for monitoring migrated AIX workloads.

## Performance efficiency

- The potential bottlenecks in this architecture are the storage and compute subsystems. Make sure to choose your storage and VM SKUs accordingly.
- The available VM disk types are ultra disks, premium solid-state drives (SSDs), standard SSDs, and standard hard disk drives (HDDs). For this solution, it's best to use either premium SSDs or ultra disks.
- To estimate sizing for VMs coming from an AIX system, keep in mind that the AIX CPUs are about 1.4 times faster than most x86 vCPUs. This guideline can vary by workload.
- Place multiple VMs that need to communicate with each other in a [proximity placement group](#). Locating the VMs close to each other provides the lowest communication latency.

## Scalability

- [Azure ExpressRoute](#) supports high scale for implementations that use significant bandwidth, either for initial replication or ongoing changed data replication.
- Infrastructure management, including scalability, is automated in Azure databases.
- You can scale out the application tier by adding more application server VM instances.

## Security

- This solution uses Azure network security groups to manage traffic between Azure resources. For more information, see [Network security groups](#).
- Follow [Azure best practices for network security](#) as closely as possible.
- For VM or infrastructure-as-a-service (IaaS) security, follow the [Security best practices for IaaS workloads in Azure](#).

## Cost optimization

- Migrating AIX workloads to Linux in Azure can bring substantial cost savings. You eliminate hardware maintenance, reduce facility costs, and can usually reduce operational costs by a factor of eight to 10. Azure can accommodate added capacity for seasonal or periodic workloads as needed, which reduces overall cost.
- Migrating AIX workloads to Azure can also reduce costs by using cloud-native services. Examples include:
  - Using Azure App Service for the presentation tier instead of setting up multiple VMs.
  - Segmenting workloads with Azure virtual networks instead of using hardware-based firewalls.

## Contributors

*This article is maintained by Microsoft. It was originally written by the following contributors.*

Principal author:

- [Jonathon Frost](#) | Principal Program Manager

## Next steps

- [Migrating AIX Workloads to Azure: Approaches and Best Practices](#).
- [AIX to Red Hat Enterprise Linux Strategic Migration Planning Guide](#).
- For more information, contact [legacy2azure@microsoft.com](mailto:legacy2azure@microsoft.com).

## Related resources

- [High availability and disaster recovery scenarios for IaaS apps](#)
- [Multi-tier web application built for HA/DR](#)

- Multi-region N-tier application
- Run a Linux VM on Azure

# High-volume batch transaction processing

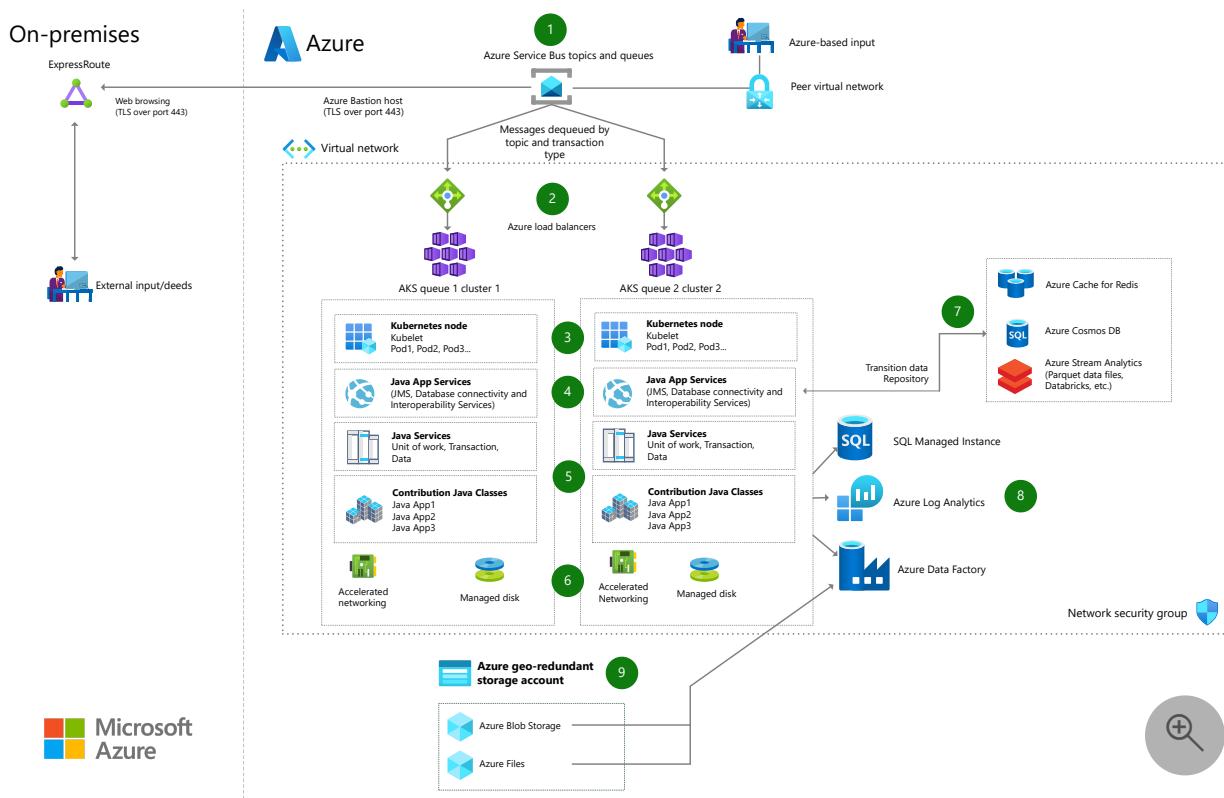
Azure Kubernetes Service (AKS)   Azure Service Bus   Azure Virtual Machines

The architecture uses AKS to implement compute clusters of the applications that process high-volume batches of transactions. The applications receive the transactions in messages from Service Bus topics or queues. The topics and queues can be at Azure datacenters in different geographic regions, and multiple AKS clusters can read input from them.

## ⓘ Note

This architecture suits a type of batch transaction processing that, on IBM mainframes, is often implemented by using the IBM MQ family of message-oriented middleware.

## Architecture



Download a [Visio file](#) of this architecture.

# Workflow

The numbered circles in the diagram correspond to the numbered steps in the following list.

1. The architecture uses Service Bus topics and queues to organize the batch processing input and to pass it downstream for processing.
2. Azure Load Balancer, a Layer 4 (TCP, UDP) load balancer, distributes incoming traffic among healthy instances of services defined in a load-balanced set. Load balancing and management of connections optimize processing.
3. The AKS cluster worker nodes listen to Service Bus queue endpoints for input.
4. The Java nodes use Java Message Service to connect to Service Bus, and Java interfaces like Java Database Connectivity to connect to other data sources. They use other Java APIs as needed.
5. The recoverable transactions run along with the business code for each batch step.
6. The batch infrastructure uses Azure accelerated networking for speed.
7. Azure Cache for Redis, Azure Cosmos DB, and Azure Stream Analytics provide working storage if needed.
8. The permanent data layer uses Azure Data Factory for data integration and Azure SQL Managed Instance, business critical performance tier, for high availability. The permanent storage is loosely coupled for easy switching to other database technologies, and for optimization of storage organization (using shards or partitions, for example).
9. The data solutions (transitional and permanent) use the Azure Storage geo-redundant storage (GRS) option to protect against catastrophic failures.

# Components

The architecture uses these components:

- [Azure Virtual Network](#) provides a secure private network in the cloud. It can connect virtual machines (VMs) to one another, to the internet, and to on-premises networks.
- [Azure ExpressRoute](#) provides private connections between Azure datacenters and on-premises infrastructure.
- [Azure Bastion](#) provides private and fully managed RDP and SSH access to VMs.
- [Azure Virtual Machines](#) provides the flexibility of virtualization without having to provide and maintain the hardware that hosts it. The operating system choices include Windows and Linux.
- A VM created with accelerated networking uses single root I/O virtualization (SR-IOV), greatly improving its networking performance. For more information, see

[Create a Windows VM with accelerated networking using Azure PowerShell](#) and [Overview of Single Root I/O Virtualization \(SR-IOV\)](#).

- An Azure network interface connects a VM to the internet, and to Azure and on-premises resources. As shown in this architecture, you can give each child VM its own network interface and IP address. For more information on network interfaces, see [Create, change, or delete a network interface](#).
- [Azure Managed Disks](#) are high-performance, highly durable block storage for VMs. There are four disk storage options for the cloud: Ultra Disk Storage, Premium SSD, Standard SSD, and Standard HDD.
- [Azure Kubernetes Service \(AKS\)](#) is a fully managed Kubernetes service for deploying and managing containerized applications.
- [Service Bus](#) provides reliable cloud messaging as a service (MaaS) and simple hybrid integration.
- [Azure load balancing services](#) provides scaling for high availability and high performance. This architecture uses [Load Balancer](#). It provides low-latency Layer 4 (TCP, UDP) load balancing capabilities to balance traffic between VMs, and across multi-tiered hybrid apps.
- [Azure Cache for Redis](#) is a lightning-fast and fully managed in-memory caching service for sharing data and state among compute resources.
- [Azure Cosmos DB](#) is a fast NoSQL database with open APIs for any scale.
- [Azure Stream Analytics](#) provides real-time analytics on fast-moving streams of data from applications and devices.
- [Azure Databricks](#) is a fast, easy, and collaborative big data analytics service based on Apache Spark™.
- [Azure SQL](#) is a family of SQL cloud databases that provides a unified experience for your entire SQL portfolio, and a wide range of deployment options from edge to cloud.
- [Azure SQL Managed Instance](#), part of the Azure SQL service portfolio, is a managed, secure, and always up-to-date SQL instance in the cloud.
- [Data Factory](#) is a fully managed and serverless data integration solution for preparing, and transforming all your data at scale.
- Data Factory supports the Parquet file data format. For more information, see [Parquet format in Azure Data Factory](#).
- Log Analytics is a tool in the Azure portal used to edit and run log queries on [Azure Monitor](#) logs. For more information, see [Overview of Log Analytics in Azure Monitor](#).
- The geo-redundant storage (GRS) option of [Azure Storage](#) copies your data synchronously three times within a single physical location in the primary region, then copies it asynchronously to a single physical location in the secondary region. For more information, see [Azure Storage redundancy](#).

- [Azure Blob Storage](#) is massively scalable and secure REST-based object storage for cloud-native workloads, archives, data lakes, high-performance computing, and machine learning.
- [Azure Files](#) provides simple, secure, and serverless enterprise-grade file shares in the cloud. You use the industry-standard Server Message Block (SMB) and Network File System (NFS) protocols to access the shares.

## Scenario details

On Azure, you can implement batch transaction processing—such as posting payments to accounts—by using an architecture based on Microsoft Azure Kubernetes Service (AKS) and Azure Service Bus. This type of architecture provides the transaction processing speed, scaling, and reliability required for high-volume batch processing.

Typically, a message remains queued until its transaction completes, allowing for recovery if there's a failure. Also, you can replicate topics and queues to other regions, to share workloads and to continue processing even if a region fails.

## Potential use cases

The solution is ideal for the finance, education, and science industries. This architecture is for high-volume processing of batches of transactions, especially independent transactions that can be processed in parallel. It's therefore a likely candidate for use in migrating mainframe batch processing. Possible applications are:

- Processing of financial transactions, such as payroll, orders, and payments.
- Processing of experimental data gathered by scientific instruments.
- Other mainframe batch processing.

## Considerations

The following considerations, based on the [Azure Well-Architected Framework](#), apply to this solution:

## Availability

- [Azure Site Recovery](#) disaster recovery service protects against major outages. It's dependable, cost-effective, and easy to deploy.
- Availability sets for VMs ensure that enough VMs are available to meet mission-critical batch process needs.

- Service Bus, AKS, and Azure SQL Managed Instance provide high availability and recoverability across geographic regions.

## Operational

- [Azure Resource Manager templates \(ARM templates\)](#) provide a configuration language to describe your resources in templates that you can use for scripted deployment. The templates also provide monitoring and alerting capabilities.

## Performance efficiency

- The architecture is designed to accommodate parallel processing of independent transactions.
- Service Bus, AKS, and other Azure PaaS features provide high performance for transaction processing, computing, and data storage.

## Scalability

- Service Bus, AKS, and other Azure PaaS features dynamically scale as needed.

## Security

- All the components within the Service Bus batch architecture work with Azure security components, such as Microsoft Entra ID, Virtual Network, and encryption.

## Cost optimization

To estimate costs for your implementation of this solution, use the [Pricing calculator](#).

The autoscale features of AKS clusters—and other Azure Platform as a Service (PaaS) features that provide scaling on demand—keep costs at a minimum.

Here are pricing considerations for specific components:

- Most enterprises already have a Microsoft Active Directory implementation. If not, [Microsoft Entra ID P1 or P2](#) is low cost.
- [Windows VM pricing](#) and [Linux VM pricing](#) depend on your compute capacity.
- For Premium SSD or Ultra managed storage disks pricing, see [Managed Disks pricing](#).
- There are no upfront costs for [Azure SQL Database](#); you pay for resources as used.

- For [ExpressRoute](#), you pay a monthly port fee and outbound data transfer charges.
- [Azure Storage](#) costs depend on data redundancy options and volume.
- [Azure Files](#) pricing depends on many factors: data volume, data redundancy, transaction volume, and the number of file sync servers that you use.
- For SSD managed disk pricing, see [Managed Disks](#) pricing.
- For [Site Recovery](#), you pay for each protected instance.
- These services are free with your Azure subscription, but you pay for usage and traffic:
  - [Load Balancer](#).
  - Your activity run volume determines the cost of [Data Factory](#).
  - For [Azure Virtual Network](#), IP addresses carry a nominal charge.
  - Outbound data transfer volume determines [Azure Bastion](#) costs.

## Next steps

- To learn more about AKS, read: [Azure Kubernetes Service solution journey](#).
- To learn more about Service Bus, read: [Service Bus queues, topics, and subscriptions](#).

## Related resources

- Techniques used in this architecture:
  - [Azure Service Bus Geo-disaster recovery](#).
  - [Use geo-redundancy to design highly available applications](#).
  - [What are ARM templates?](#)
- Azure reference architectures:
  - [Migrate IBM mainframe applications to Azure with TmaxSoft OpenFrame](#).
  - [Refactor IBM z/OS mainframe Coupling Facility \(CF\) to Azure](#).
  - [Micro Focus Enterprise Server on Azure VMs](#).
  - [Unisys mainframe migration](#).

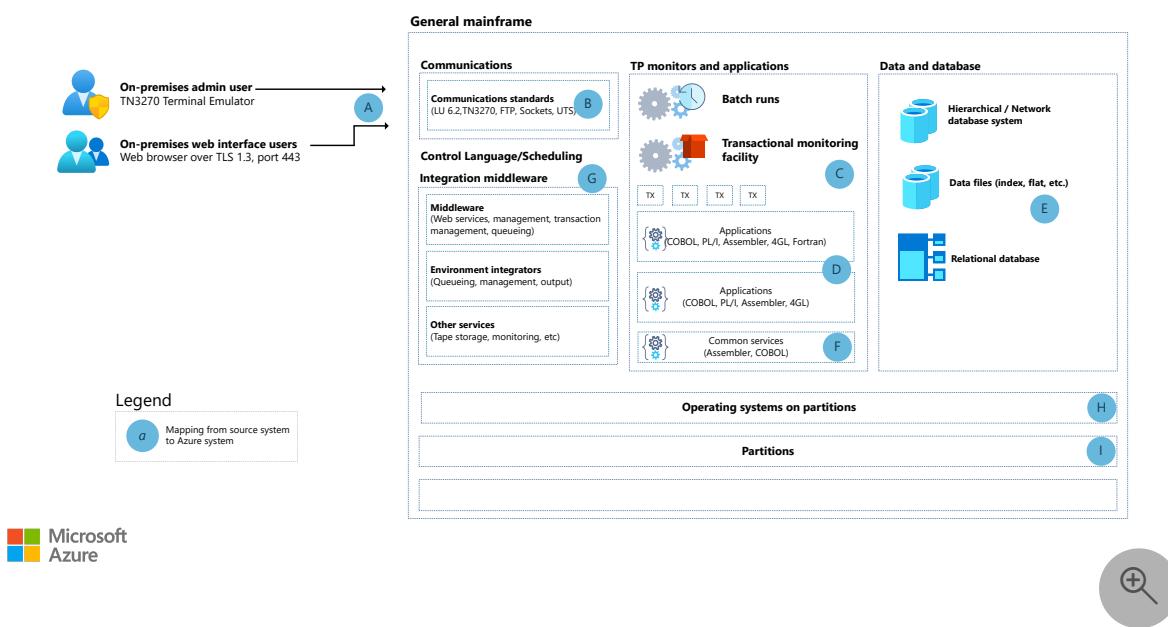
# Extend mainframe applications to Azure by using Verastream Host Integrator

Microsoft Power Platform   Azure Virtual Machines   Azure Virtual Network   Azure Kubernetes Service (AKS)

Azure Monitor

This architecture shows how legacy mainframe and midrange terminal-based applications (such as TN-3270) and data can be extended to Azure without any changes to the existing mainframe and midrange application landscape. There are multiple ways in which this scenario can be achieved. The solution discussed in this article uses Azure services like Kubernetes service (AKS), platforms like Power Platform, and Micro Focus Verastream Host Integrator (VHI).

## Legacy IBM z/OS architecture



Download a [Visio file](#) of this architecture.

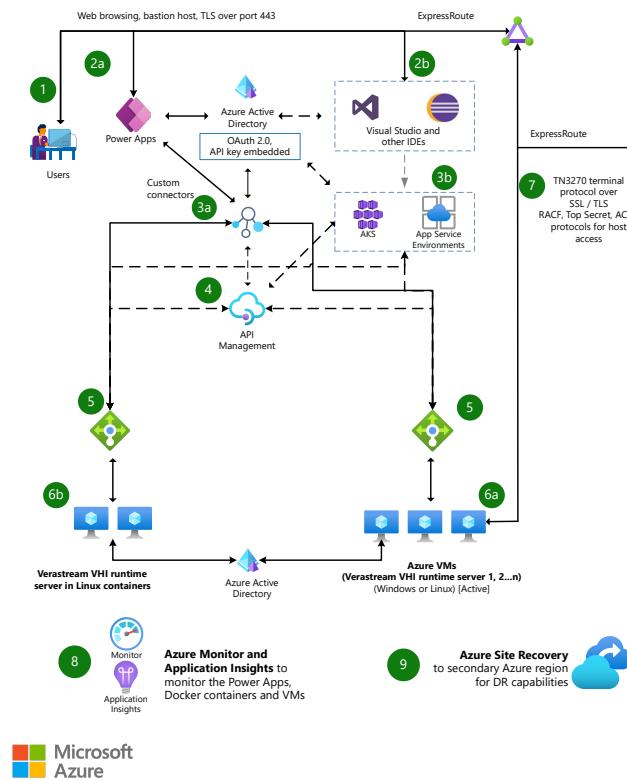
## Workflow

1. Data is input over TCP/IP, including TN3270 and HTTP(S).

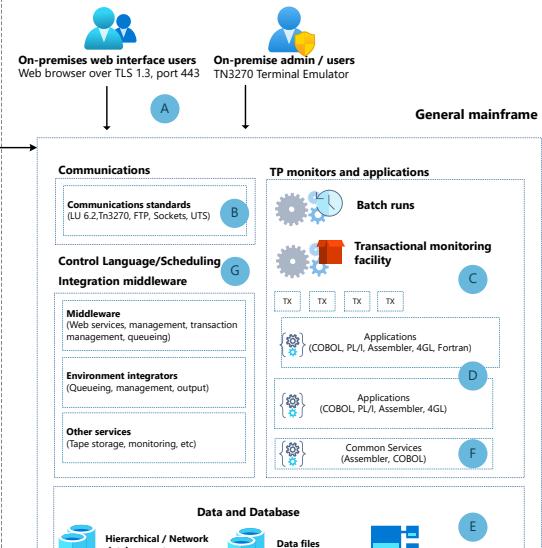
2. Data is input into the mainframe via standard mainframe protocols.
3. Receiving applications can be either batch or online systems.
4. Business applications written in COBOL, PL/I, or Assembler (or compatible languages) run in environments enabled for batch and online.
5. Data and database services commonly used are hierarchical and network database systems, data files, and relational database types enabled within the environment.
6. Common services enabled include program execution, I/O operations, error detection, and protection within the environment.
7. Middleware and utility services manage services like tape storage, queueing, output, and web services within the environment.
8. Operating systems provide the specific interface between the engine and the software it's running.
9. The partitions used are needed to run separate workloads or to segregate work types within the environment.

## Mainframe architecture extended to Azure

Extend the mainframe architecture to Azure



On-premises (no change)



Legend

- *n* Technical annotations
- *a* Mapping from source system to Azure



Download a [Visio file](#) of this architecture.

## Workflow

The following workflow corresponds to the preceding diagram:

1. Data is typically input from users, either from the internet or an intranet.
2. User access to the application is now enabled via a web-based presentation layer with the help of an application created with Power Apps, and validation is integrated with Microsoft Entra ID for a seamless sign-on experience. If validated, the user can access a specific Power Apps when they sign in to the Power Platform. User access is enabled using the Mainframe ID and password, which is validated against the mainframe with Verastream. (The 2b and 3b flow is an [alternative](#) workflow addressed later in this article.)
3. Application functionality is enabled by defining custom connectors. The custom connector definitions contain the corresponding Verastream APIs configured in Verastream Host Integrator software.
4. (Optional) Azure API Management. (For more information, see [Alternatives](#).)
5. Traffic is distributed evenly across multiple runtime servers with the help of a load balancer. The workload is dynamically balanced for optimal performance under high transaction volumes.
6. You can deploy and configure the VHI Server software by using one of these options:
  - Azure VMs (Windows or Linux)
  - VHI Session server in Linux containers (to be managed later by AKS)

Multiple runtime environments help with workload management and also help provide failover protection. For example, when a service outage occurs on any runtime server, the remaining servers automatically provide uninterrupted failover protection. The Verastream management console provides an interface to manage the extended server environment. It lets administrators remotely configure, deploy, and monitor resources. Users and groups from the directory server can be added to the Administrator, Developer, and User authorization profiles. You can use Azure VM bastion hosts to provide admin access to the VMs, which improves security by minimizing open ports.

7. Verastream services that run on the Azure Virtual Machines or Linux Docker containers (to be managed later by AKS) will then connect to the on-premises Mainframe Transaction Processing Application over TN3270 protocol with SSL/TLS. RACF, Top Secret, and ACF2-based protocols will continue to be used for host access, which is facilitated by Azure ExpressRoute.
8. Azure Application Monitor and Application Insights can be used to monitor Power Platform, the application APIs, Verastream services, session pools, and security. Verastream comes with a fully configurable ability to view and report all pertinent information to third-party SNMP or JMX management tools, which can be used by Azure Monitor and Azure Application Insights.
9. Azure Site Recovery is used for disaster recovery of the VMs.

## Components

- [Power Platform](#) increases agility across your organization by helping you build low-code apps that modernize processes and solve tough challenges.
- [Azure API Management](#) provides a hybrid, multicloud management platform for APIs across all environments. APIs enable digital experiences, simplify application integration, underpin new digital products, and make data and services reusable and universally accessible.
- [Azure Monitor](#) helps maximize the availability and performance of your applications and services. It's a comprehensive solution for collecting, analyzing, and acting on customer data from your cloud and on-premises environments. This information helps you understand how your applications perform and proactively identify issues that affect them and the resources they depend on.
- [Azure Virtual Machines](#) is one of several Azure services that provide on-demand, scalable computing resources. With Azure VMs, you get the flexibility of virtualization without having to buy and maintain physical hardware.
- [Microsoft Azure Virtual Network](#) is the fundamental building block for Azure private networks. Virtual Network lets many types of Azure resources, such as VMs, communicate with each other, the internet, and on-premises networks. Virtual Network is similar to a traditional network that you'd operate in your own datacenter but offers Azure infrastructure benefits like scalability, availability, and isolation.
- [Azure ExpressRoute](#) extends an on-premises network into Microsoft cloud services over a private connection, facilitated by a connectivity provider. With

ExpressRoute, you can establish connections to Microsoft cloud services, such as Microsoft Azure and Office 365.

- [Azure Kubernetes Service \(AKS\)](#) simplifies deploying managed Kubernetes clusters in Azure by offloading the operational overhead to Azure. AKS helps you deploy and manage the containerized components in this architecture, including your custom-made web UI applications. Because Kubernetes masters are managed by Azure, you only manage and maintain the agent nodes.

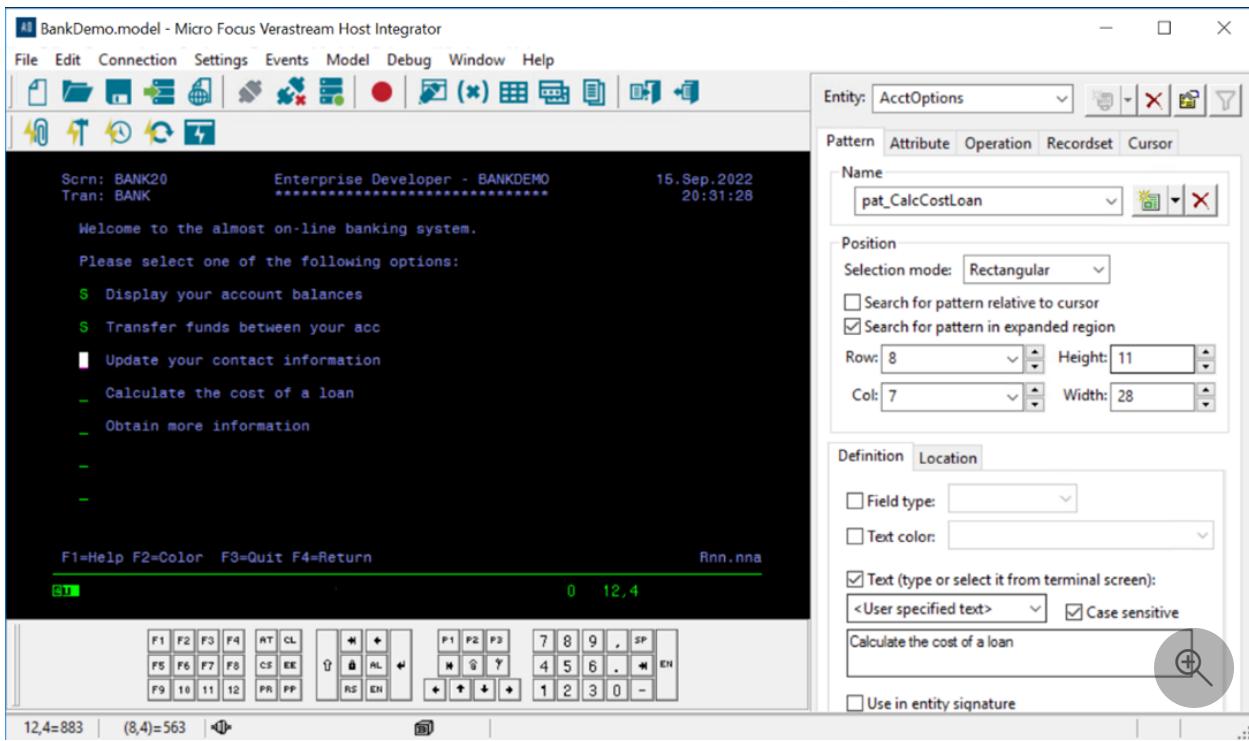
## Alternatives

- (2b in the [previous image](#).) As an alternative to Power Apps, which is a low-code or no-code option, you can develop a custom-made web UI application with programming languages like C#, Angular JS, or Java using IDEs like Visual Studio or Eclipse.
- (3b in the [previous image](#).) These UI applications can be deployed on AKS and App Service Environments as well. These applications can then either directly connect to the APIs hosted on the Verastream Host Integrator runtime environments or connect via the Azure API Management.
- (4 in the [previous image](#).) Azure API Management (optional) lets you publish the Verastream services as APIs and manage them with policies. Doing so helps control the number of incoming calls, direct traffic, and set usage quotas at different levels. This method is an alternative to directly connecting to the services running on Verastream Host Integrator runtime environments.

## Scenario details

As part of Power Platform, Power Apps is an intuitive, collaborative, and extensible platform of low-code tools that makes it easy to create efficient and flexible solutions. With Power Apps, production-ready apps with less code can be created with custom connectors or out-of-the-box connectors.

Micro Focus' VHI is a powerful integration platform that simplifies mainframe and host-based application functionality into a component form, web-service, such as RESTful and SOAP-based web services. It then deploys them natively on an Azure VM (Windows or Linux), or via a Linux Docker container runtime environment (Verastream Host Integrator).



This architecture is focused on extending a COBOL-CICS screen application workload to the Azure platform by using two approaches:

- Verastream Host Integrator and Azure Power Apps (low-code or no-code approach)
- Verastream Host Integrator and Visual Studio or Eclipse IDEs for developing a web-based UI and deploying the application natively on Azure using App Service Environments or Kubernetes service

This integration doesn't require any changes in the mainframe or midrange platform.

End users can now access the same business functionality that was originally available using mainframe and midrange terminals from outside the mainframe and midrange environment, such as from a mobile or desktop screen using web browsers.

Power Platform Power Apps offers a low-code or no-code option to create a web-based UI that will in turn connect to the above developed services.

This solution is essentially a no-changes-needed approach with respect to the application on mainframe and midrange environments because Verastream services integrate to the existing mainframe and midrange-based application over TN3270 protocols, similar to how a business user would.

## Potential use cases

Many scenarios can benefit from the extend-to-Azure architecture, including these use cases:

- Enable direct access to legacy applications through smartphones and tablets via web browsers for users in the field to improve productivity.
- Maintain a competitive edge by expanding your user base and offering to the entire internet instead of just call center users.
- Automate workflows between applications, implementing seamless business processes without making changes to existing legacy applications.
- Enable automated application testing. VHI can use encapsulated application business logic as reusable services to support continuous integration and continuous deployment (CI/CD) practices that respond to ever-growing business demands to deliver applications on time, with minimal bugs.

## Considerations

These considerations implement the pillars of the Azure Well-Architected Framework, which is a set of guiding tenets that can be used to improve the quality of a workload. For more information, see [Microsoft Azure Well-Architected Framework](#).

## Reliability

Reliability ensures your application can meet the commitments you make to your customers. For more information, see [Overview of the reliability pillar](#).

- You can deploy this architecture in multiple regions.

## Security

Security provides assurances against deliberate attacks and the abuse of your valuable data and systems. For more information, see [Overview of the security pillar](#).

- Use single sign-on to access the Power Platform by using Microsoft Entra ID and authentication via LDAP, which is supported by VHI. Any host-based security implementations (such as RACF, TopSecret, or ACF-2) remain fully active.
- VHI accommodates end-to-end security using TLS and SSH. Host-to-server and server-to-client communications can be secured. Public key cryptography helps protect all data passed between client web applications and the Verastream runtime server. FIPS-validated crypto libraries enhance compliance with data-protection guidelines defined by the U.S. National Institute of Standards and

Technology. While a requirement for many government IT systems, these security standards benefit private-sector organizations as well.

- This solution uses an Azure network security group (NSG) to manage traffic between Azure resources. For more information, see [Network security groups](#).
- [Azure Bastion](#) maximizes admin access security by minimizing open ports. Bastion provides secure and seamless RDP/SSH connectivity to virtual network VMs directly from the Azure portal over TLS.

## Cost optimization

Cost optimization is about looking at ways to reduce unnecessary expenses and improve operational efficiencies. For more information, see [Overview of the cost optimization pillar](#).

- Azure provides cost optimization by running on Windows VMs or Linux containers (to be managed later by AKS). Doing so lets you shut down the VMs or containers when they're not in use and script a schedule for known usage patterns. Azure focuses on avoiding unnecessary costs by identifying the right number or resource types, analyzing spend over time, and scaling to meet business needs without overspending.
- For compute, use [Azure Reservations](#) and [Azure savings plan for compute](#) with a one-year or three-year contract and receive significant savings off pay-as-you-go prices. In many cases, you can further reduce your costs with reserved instance size flexibility.
- Azure provides various licensing options for the Power Apps platform as well, which can be controlled and managed with respect to the total number of users, sign-ins that are allowed, and page views.

Use [Azure Pricing Calculator](#) to estimate the cost of implementing the solution.

## Operational excellence

Operational excellence covers the operations processes that deploy an application and keep it running in production. For more information, see [Overview of the operational excellence pillar](#).

- With the extend-target architecture, you have full flexibility with your deployment options in development and production. This transformation supports both the immediate adoption of the cloud and the adoption of both DevOps and Agile working principles.

- Holistic Monitoring in Azure Monitor can be plugged in to get full observability across the integrated solution. As part of the Azure Monitor suite, Azure Application Insights is recommended due to its direct integration capabilities to monitor Power Apps, the VMs and Linux containers using Docker on Azure and for the services, VHI session pools, and security. The Verastream Management console provides an interface to configure the reporting of pertinent information to Azure Monitor.

## Performance efficiency

Performance efficiency is the ability of your workload to scale to meet the demands placed on it by users in an efficient manner. For more information, see [Performance efficiency pillar overview](#).

- Performance efficiency is built into this solution because of the load balancers. When multiple runtime servers are deployed, the workload is dynamically balanced for optimal performance under high-transaction volumes. If a service outage occurs on any runtime server, the remaining servers automatically provide uninterrupted failover protection.
- At the VHI level, the platform manages sessions using session pooling and emphasis on a low ratio of sessions to users. Verastream scales seamlessly across multiple runtime servers to deliver rapid response and 24/7 reliability.

## Contributors

*This article is maintained by Microsoft. It was originally written by the following contributors.*

Principal authors:

- [Jim Dugan](#) | Principal TPM
- [Venkat Ramakrishnan](#) | Senior TPM

Other contributor:

- [Bhaskar Bandam](#) | Senior TPM

*To see non-public LinkedIn profiles, sign in to LinkedIn.*

## Next steps

- [Verastream Host Integrator | Micro Focus](#)

- [VHI data sheet ↗](#)
- [Power Platform ↗](#)
- [Azure Kubernetes Service documentation](#)
- [Virtual machines in Azure](#)
- [What is Azure Virtual Network?](#)
- [Azure Monitor ↗](#)

For more information, contact [legacy2azure@microsoft.com](mailto:legacy2azure@microsoft.com).

## Related resources

- [General mainframe refactor to Azure](#)
- [Azure mainframe and midrange architecture design](#)
- [Make the switch from mainframes to Azure](#)

# Extend mainframes to digital channels by using standards-based REST APIs

Microsoft Entra ID

Azure ExpressRoute

Azure Monitor

Azure Red Hat OpenShift

Power Apps

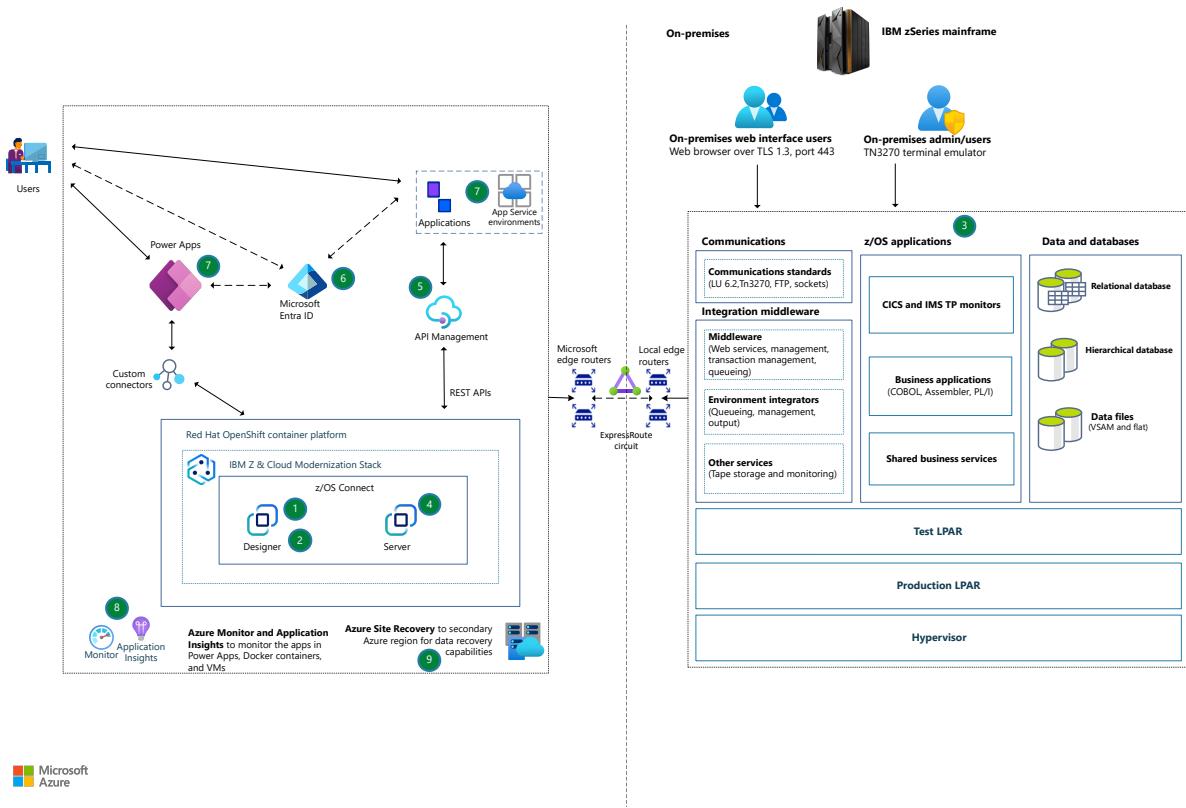
Digital transformation is imperative for any business that's trying to compete in the marketplace. This transformation demands timely access to data and data insights, fueling new business processes and client experiences, which might cause an overlooked or misunderstood effect to existing applications and data. The demand for streamlined access has led to an integration approach that relies on Representational State Transfer (REST) APIs based on industry standards. This architecture shows how IBM Z and Cloud Modernization Stack with standards-based REST APIs achieves a low-code solution for mainframe subsystems.

## Overview

This architecture extends mainframe applications to Azure without disruptions or modifications to existing mainframe applications. IBM z/OS Connect, a component of IBM Z and Cloud Modernization Stack, is used to provide a more reliable and more secure connectivity between applications on Azure and applications and data on z/OS. Its purpose is to integrate and provide access to the data and services available on the mainframe.

IBM Z and Cloud Modernization Stack and z/OS Connect are easily deployed on Azure via Azure Marketplace or Azure Resource Manager templates. When you use this solution, you can build REST APIs for z/OS applications and data while adhering to OpenAPI standards. This approach allows you to scale business-critical application programming interfaces (APIs) and take advantage of the strengths of IBM Z. Seamless integration with API management solutions like [Azure API Management](#) ensures effective API governance. You can integrate APIs with web applications or Microsoft Power Platform for efficient data exchange and integration.

## Architecture



Download a [Visio file](#) of this architecture.

## Workflow

Take the following steps to create and deploy APIs for mainframe applications by using a contract-first approach:

1. Import an OpenAPI v3 (OAS3) declarative JSON API schema file into the z/OS Connect Designer. For more information, see [What is the z/OS Connect Designer?](#)
2. Use z/OS Connect Designer to [Map your API and z/OS Assets](#).
3. Test the functionality of the APIs by interacting with core z/OS applications and push the mappings into source control management (SCM).
4. Build a web archive (WAR) file and run the production in [the z/OS Connect Server image](#).
5. Import the OAS3 specification into [API Management](#) and establish a connection with the z/OS Connect Server.
6. Enable and enforce API authentication and authorization mechanisms by using Microsoft Entra ID for enhanced security. For more information, see [Authentication and authorization to APIs in Azure API Management](#).

Access mainframe applications through Azure by:

- Signing in to Microsoft Entra ID (step 6 in the image), which provides access to the client application. The client applications also communicate with Microsoft Entra ID for authentication and authorization of access to resources.

- Accessing client applications such as Power Apps or a custom web app (step 7 in the image), which then access the mainframe applications through REST API access to IBM Z and Cloud Modernization Stack.

The steps taken by IT staff to monitor the system with Azure tools and implement disaster recovery measures by using Azure Site Recovery include:

- Deploy new or enhanced applications (step 7 in the image) to consume the REST API interfaces exposed through API Management.
- Use Azure Application Monitor and Application Insights (step 8 in the image) to monitor Power Platform, the application APIs, and security.
- Use Azure Site Recovery for disaster recovery (step 9 in the image).

## Components

- [Red Hat OpenShift](#) reduces the friction of developing, modernizing, deploying, running, and managing applications. Red Hat OpenShift delivers a consistent experience across public cloud, on-premises, hybrid cloud, and edge architectures.
- [IBM Z and Cloud Modernization Stack](#) provides simple and more secure access to mainframe applications and data through APIs. You can use modern [DevOps for IBM Z](#) with industry-standard tooling and modern languages that expand your talent pool.
- [IBM z/OS Connect](#) is a middleware solution that provides more secure connectivity between cloud-native applications and IBM z/OS systems. It enables organizations to integrate and use data and services residing on the mainframe, while also embracing modern technologies and Open standards.
- [Azure API Management](#) provides a hybrid, multicloud management platform for APIs across all environments. APIs enable digital experiences, simplify application integration, underpin new digital products, and make data and services reusable and accessible.
- [Azure App Service](#) is a fully managed platform for building, deploying, and scaling web apps. It supports various programming languages and frameworks, offering seamless integration with Azure services. App Service provides autoscaling and high availability, simplifying app deployment and management. [Visual Studio](#) is an integrated development environment (IDE) that you can use to write, edit, debug, and build code, and then deploy your web app.
- [Microsoft Power Platform on Azure](#) increases agility across your organization by allowing you to rapidly implement [Low-code application development on Azure](#)

to modernize processes and solve challenges.

- [Azure Monitor](#) helps maximize the availability and performance of applications and services. It delivers a comprehensive solution for collecting, analyzing, and acting on information from cloud and on-premises environments. This information helps you identify issues and understand how your applications are performing.
- [Azure ExpressRoute](#) extends on-premises networks into the Microsoft cloud over a private connection facilitated by a connectivity provider. With ExpressRoute, you can establish connections to Microsoft cloud services, such as Microsoft Azure and Office 365.
- [Azure Site Recovery](#) is a disaster recovery solution that helps protect and recover applications and workloads running on virtual or physical machines. It provides business continuity and minimizes downtime during planned or unplanned outages.

## Alternatives

In place of ExpressRoute gateway, you can use the Azure VPN Gateway. The virtual network gateway enables more secure site-to-site connectivity, connecting an on-premises network to Azure virtual network through encrypted tunnels. For more information, see [What is Azure VPN Gateway?](#)

## Scenario details

[z/OS Connect Designer](#) features an intuitive web user interface that provides a low-code approach, built specifically to create APIs for IBM Z. This graphical interface shortens development time and the learning curve for new developers who use z/OS Connect.

[Azure API Management](#) is a fully managed service that helps organizations to publish, secure, and manage APIs for their applications. It provides a comprehensive set of tools and features to create, monitor, and control the lifecycle of APIs.

[Microsoft Power Platform](#) Power Apps is a low-code or no-code option to create a web-based user interface that connects to the previously mentioned developed services. This architecture illustrates both a low-code Power Apps client and a custom web app client.

## Potential use cases

Benefits from using REST APIs to access mainframe applications include:

- *Frontend applications:* Front-end applications written in Java, Java EE, .NET Framework, and C and C++ can use REST APIs for mainframe applications. These applications can share business logic and units of work with back-end Customer Information Control System (CICS) applications developed in COBOL, PL/I, and other languages. This integration allows for communication between the front-end and back-end systems for efficient data exchange and processing.
- *Hybrid solutions with citizen developers:* REST APIs in mainframe applications help citizen developers within enterprises build hybrid solutions. Citizen developers can use mainframe APIs and other APIs available within their organization to create innovative applications and integrations. This democratization of API access allows for faster development cycles and promotes collaboration between different teams.

REST APIs in mainframe applications offer opportunities for modernization and expansion while preserving the essential business logic and data integrity of the mainframe systems. REST APIs in mainframe applications support an array of front-end technologies and empower citizen developers.

## Considerations

These considerations implement the pillars of the Azure Well-Architected Framework, which is a set of guiding tenets that can be used to improve the quality of a workload. For more information, see [Microsoft Azure Well-Architected Framework](#).

## Reliability

Reliability ensures your application can meet the commitments you make to your customers. For more information, see [Overview of the reliability pillar](#).

- [Red Hat OpenShift Container Platform](#) provides automated deployment capabilities that help ensure your applications are deployed consistently and reliably.
- Reliability is a fundamental pillar of IBM z/OS Connect. It's engineered to manage high-transaction volumes and handle many concurrent connections. The solution's scalability extends both horizontally and vertically, enabling it to accommodate the evolving demands of expanding workloads.

## Security

Security provides assurances against deliberate attacks and the abuse of your valuable data and systems. For more information, see [Overview of the security pillar](#).

- Microsoft Entra ID provides a variety of security features and capabilities to help you protect identities, applications, and data. It also provides authentication and authorization of users and applications. The integration of Microsoft Entra ID with OAuth enables more secure authentication and authorization for applications.
- IBM zSystems provides robust security capabilities for DevSecOps to mitigate business risks, safeguard application data, and help you ensure long-term security for your systems.

## Cost optimization

Cost optimization reduces unnecessary expenses and improves operational efficiencies. For more information, see [Overview of the cost optimization pillar](#).

- IBM Z and Cloud Modernization Stack and Azure DevOps reduce the need for custom z/OS tooling by allowing organizations to implement the same CI/CD toolchain and practices as the rest of their enterprise.
- Azure provides various licensing options for the Power Apps platform, which are managed depending on the total number of users, allowed sign-ins, and page views.

Use the [Pricing calculator](#) to estimate the cost of implementing your solution.

## Operational excellence

Operational excellence covers the operations processes that deploy an application and keep it running in production. For more information, see [Overview of the operational excellence pillar](#).

- [IBM z/OS Connect](#) facilitates access to backend application functions, converting them into microservices with accessible APIs. IBM z/OS Connect enables other applications to interact at scale with these services while also providing API management and monitoring capabilities.
- [Red Hat OpenShift Container Platform](#) streamlines deployment processes, bolsters scalability, fortifies security measures, offers robust monitoring capabilities, facilitates continuous integration and delivery, and integrates with existing operational tools and processes.

# Performance efficiency

Performance efficiency covers the operations processes that deploy an application and keep it in production. For more information, see [Overview of the performance efficiency pillar](#).

- z/OS Connect handles multiple API requests concurrently by using the parallel processing capabilities of IBM Z. This parallel execution enhances performance by using system resources and reducing response times for API calls.
- Performance efficiency is a core strength of IBM z/OS Connect. It handles high transaction volumes and manages concurrent connections. The solution's scalability expands both horizontally and vertically, allowing it to adapt to the evolving demands of workloads.

## Contributors

*This article is maintained by Microsoft. It was originally written by the following contributors.*

Principal authors:

- [Bhaskar Bandam](#) ↗ | Senior Technical Program Manager
- [Ivan Dovgan](#) ↗ | Chief Architect

Other Contributors:

- [Jim Dugan](#) ↗ | Principal Technical Program Manager
- Madhu Ananthapadmanabh | Z Hybrid Cloud Integration solution architect

*To see non-public LinkedIn profiles, sign in to LinkedIn.*

## Next steps

For more information, contact [legacy2azure@microsoft.com](mailto:legacy2azure@microsoft.com).

- [Azure DevOps Services](#) ↗
- [IBM Z and Cloud Modernization Stack](#) ↗
- [Technical White Paper on Azure DevOps for z Systems](#) ↗
- [Microsoft Power Platform](#) ↗
- [IBM z/OS Connect overview](#) ↗

## Related resources

- Implement Azure DevOps for mainframe applications that use IBM Z and Cloud Modernization Stack

# General mainframe refactor to Azure

Azure Files

Azure Load Balancer

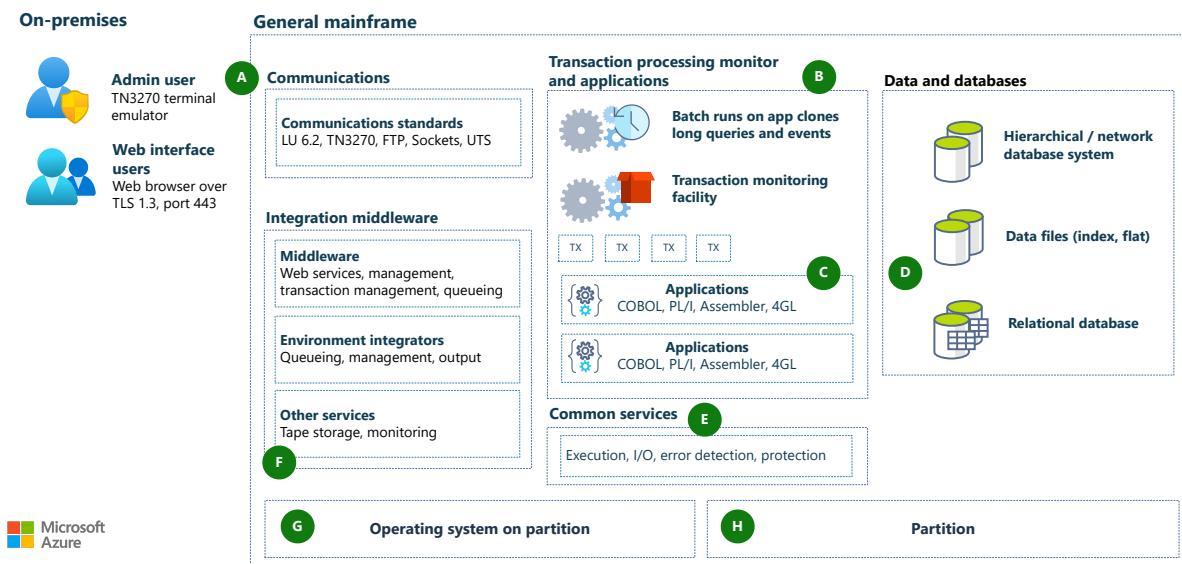
Azure SQL Database

Azure Storage

Azure Virtual Machines

The following architecture illustrates a general refactoring approach that can use Azure Kubernetes Service (AKS) or Azure virtual machines (VMs). The choice depends on existing applications' portability and your preference. Refactoring can speed up the move into Azure by automatically converting code to Java or .NET, and converting pre-relational to relational databases.

## Mainframe architecture



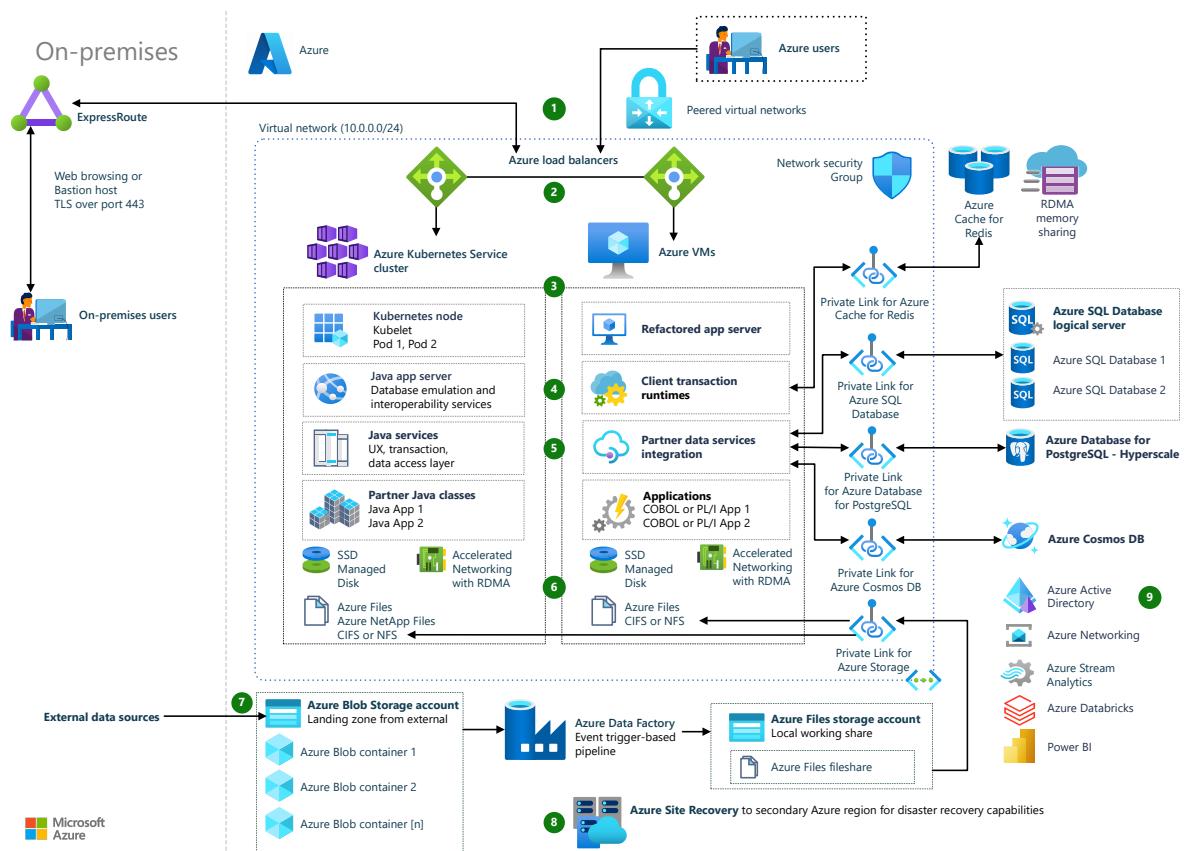
Download a [Visio file](#) of this architecture.

## Workflow

- On-premises users access the mainframe over TCP/IP by using standard mainframe protocols like TN3270 and HTTPS (A).
- Receiving applications can be either batch or online systems (B).
- COBOL, PL/I, Assembler, or compatible languages run in enabled environments (C).
- Typical data and database services include hierarchical or network database systems, index or flat data files, and relational databases (D).
- Common services include program execution, I/O operations, error detection, and protection (E).

- Middleware and utility services manage tape storage, queueing, output, and web services (F).
- Operating systems are the interface between the compute engine and the software (G).
- Partitions run separate workloads or segregate work types within the environment (H).

## Refactored Azure architecture



Download a [Visio file](#) of this architecture.

## Workflow

1. Input comes from remote clients via ExpressRoute, or from other Azure users.

TCP/IP is the primary way to connect to the system.

- On-premises users can access web-based applications over Transport Layer Security (TLS) port 443. Web applications' presentation layers can remain unchanged, to minimize end user retraining. Or, you can update the presentation layers with modern UX frameworks.

- On-premises administrative access uses Azure Bastion hosts to maximize security by minimizing open ports.
  - Azure users connect to the system via virtual network peering.
2. In Azure, Azure Load Balancer manages access to the application compute clusters. Load Balancer supports scale-out compute resources to handle input. You can use a level-7 application level or level-4 network level load balancer, depending on how the application input reaches the compute cluster entry point.
3. Application compute clusters can run on Azure VMs, or run in containers in AKS clusters. Usually, mainframe system emulation for PL/I or COBOL applications uses VMs, and applications refactored to Java or .NET use containers. Some mainframe system emulation software also supports deployment in containers. Compute resources use premium or ultra solid-state drive (SSD) managed disks with Accelerated Networking and Remote Direct Memory Access (RDMA).
4. Application servers in the compute clusters host the applications based on language capability, such as Java classes or COBOL programs. The servers receive application input, and share application state and data by using Azure Cache for Redis or RDMA.
5. Data services in the application clusters support multiple connections to persistent data sources. Azure Private Link provides private connectivity from within the virtual network to Azure services. Data sources can include:
- PaaS data services like Azure SQL Database, Azure Cosmos DB, and Azure Database for PostgreSQL - Hyperscale.
  - Databases on VMs, such as Oracle or Db2.
  - Big data repositories like [Azure Databricks](#) and Azure Data Lake.
  - Streaming data services like Apache Kafka and [Azure Stream Analytics](#).
6. Data storage can be either local-redundant or geo-redundant, depending on usage. Data storage can use a combination of:
- High-performance storage with ultra or premium SSD disks.
  - File storage with Azure NetApp Files or Azure Files.
  - Standard storage, including blob, archive, and backup storage.
7. Azure PaaS data services provide scalable and highly available data storage that you can share among compute cluster resources. This storage can also be geo-redundant.
- Azure Blob Storage is a common landing zone for external data sources.

- Azure Data Factory supports data ingestion and synchronization of multiple Azure and external data sources.
8. Azure Site Recovery provides DR for VM and container cluster components.
9. Services like [Microsoft Entra ID](#), [Azure Networking](#), Azure Stream Analytics, Azure Databricks, and [Power BI](#) can easily integrate with the modernized system.

## Components

This example features the following Azure components. Several of these components and workflows are interchangeable or optional depending on your scenario.

- [Azure ExpressRoute](#) extends your on-premises networks into Azure over a private, dedicated fiber connection from a connectivity provider. ExpressRoute establishes connections to Microsoft cloud services like Azure and Microsoft 365.
- [Azure Bastion](#) provides seamless Remote Desktop Protocol (RDP) or secure shell (SSH) connectivity to virtual network VMs from the Azure portal over TLS. Azure Bastion maximizes administrative access security by minimizing open ports.
- [Azure Load Balancer](#) distributes incoming traffic to the compute resource clusters. You can define rules and other criteria to distribute the traffic.
- [Azure Kubernetes Service \(AKS\)](#) is a fully managed Kubernetes service to deploy and manage containerized applications. AKS offers serverless Kubernetes, an integrated continuous integration and continuous delivery (CI/CD) experience, and enterprise-grade security and governance.
- [Azure Virtual Machines](#) offers many sizes and types of on-demand, scalable computing resources. With Azure VMs, you get the flexibility of virtualization without having to buy and maintain physical hardware.
- [Azure Virtual Network](#) is the fundamental building block of Azure private networks. Azure VMs within virtual networks can communicate securely with each other, the internet, and on-premises networks. A virtual network is like a traditional on-premises network, but with Azure infrastructure benefits like scalability, high availability, and isolation.
- [Azure Private Link](#) provides private connectivity from a virtual network to Azure services. Private Link simplifies network architecture and secures the connection between Azure endpoints by eliminating public internet exposure.

- [Azure Cache for Redis](#) adds a quick caching layer to application architecture to handle large volumes at high speed. Azure Cache for Redis scales performance simply and cost-effectively, with the benefits of a fully managed service.
- [Azure Storage](#) offers scalable, secure cloud storage for all your data, applications, and workloads.
  - [Azure Disk Storage](#) is high-performance, durable block storage for business-critical applications. Azure managed disks are block-level storage volumes that are managed by Azure on Azure VMs. The available types of disks are ultra disks, premium SSDs, standard SSDs, and standard hard disk drives (HDDs). This architecture uses either premium SSDs or ultra disk SSDs.
  - [Azure Files](#) offers fully managed file shares in the cloud that are accessible via the industry standard Server Message Block (SMB) protocol. Cloud and on-premises Windows, Linux, and macOS deployments can mount Azure Files file shares concurrently.
  - [Azure NetApp Files](#) provides enterprise-grade Azure file shares powered by NetApp. NetApp Files makes it easy for enterprises to migrate and run complex, file-based applications with no code changes.
  - [Azure Blob Storage](#) is scalable and secure object storage for archives, data lakes, high-performance computing, machine learning, and cloud-native workloads.
- [Azure databases](#) offer a choice of fully managed relational and NoSQL databases to fit modern application needs. Automated infrastructure management provides scalability, availability, and security.
  - [Azure SQL Database](#) is a fully managed PaaS database engine. SQL Database always runs on the latest stable version of SQL Server and a patched OS with 99.99 percent availability. Built-in PaaS database management capabilities include upgrading, patching, backups, and monitoring. You can focus on domain-specific, business-critical database administration and optimization.
  - [Azure Database for PostgreSQL](#) is a fully managed database based on the open-source Postgres relational database engine. The [Hyperscale \(Citus\) deployment option](#) scales queries across multiple machines using sharding, for applications that require greater scale and performance.
  - [Azure Cosmos DB](#) is a fully managed, fast NoSQL database with open APIs for any scale.

- [Azure Site Recovery](#) mirrors Azure VMs to a secondary Azure region for quick failover and DR if an Azure datacenter fails.

## Scenario details

Refactoring workloads to Azure can transform mainframe applications that run on Windows Server or Linux. You can run these applications more cost effectively with cloud-based Azure infrastructure as a service (IaaS) and platform as a service (PaaS).

The general refactoring approach for mainframe applications also drives infrastructure transformation from legacy-proprietary into standardized, benchmarked, open technologies. This transformation promotes agile DevOps principles that are today's high-productivity, open-systems standard. Refactoring transitions away from islands of unique legacy infrastructures, processes, and applications to a unified land of better business and IT alignment.

This general refactoring approach can use Azure Kubernetes Service (AKS) or Azure virtual machines (VMs). The choice depends on existing applications' portability and your preference. Refactoring can speed up the move into Azure by automatically converting code to Java or .NET, and converting pre-relational to relational databases.

Refactoring supports different methodologies for moving client workloads to Azure. One method is to convert and move the entire mainframe system to Azure at once, saving interim mainframe maintenance and facility support costs. This approach carries some risk. All application conversion, data migration, and testing processes must align for a smooth transition from the mainframe to Azure.

A second methodology is to move applications from the mainframe to Azure gradually, with complete transition as the ultimate goal. This tactic provides savings per application, and lessons learned to convert each application can help with later conversions. Modernizing each application on its own schedule can be more relaxed than converting everything at once.

## Potential use cases

Refactoring on Azure can help organizations to:

- Modernize infrastructure, and escape mainframes' high costs, limitations, and rigidity.
- Move mainframe workloads to the cloud without the side effects of a complete redevelopment.

- Migrate business-critical applications, while maintaining continuity with other on-premises applications.
- Benefit from Azure's horizontal and vertical scalability.
- Gain disaster recovery (DR) capabilities.

## Considerations

The following considerations, based on the [Azure Well-Architected Framework](#), apply to this solution:

### Availability

Azure Site Recovery mirrors the Azure VMs to a secondary Azure region for quick failover and DR if the primary Azure datacenter fails.

### Operations

Refactoring not only supports faster cloud adoption, but also promotes adoption of DevOps and Agile working principles. You have full flexibility in development and production deployment options.

### Resiliency

Performance efficiency is built into this solution by the load balancers. If one presentation or transaction server fails, other servers behind the load balancers can run the workloads.

### Security

This solution uses an Azure network security group (NSG) to manage traffic between Azure resources. For more information, see [Network security groups](#).

Private Link provides private, direct connections isolated to the Azure networking backbone between the Azure VMs and Azure services.

Azure Bastion maximizes administrative access security by minimizing open ports. Bastion provides secure and seamless RDP and SSH connectivity to virtual network VMs from the Azure portal over TLS.

### Cost optimization

Azure avoids unnecessary costs by identifying the correct number of resource types, analyzing spending over time, and scaling to meet business needs without overspending.

- Azure provides cost optimization by running on VMs. You can turn off the VMs when not in use, and script a schedule for known usage patterns. See the [Azure Well-Architected Framework](#) for more information about cost optimization for [VM instances](#).
- The VMs in this architecture use either premium SSDs or ultra disk SSDs. For more information about disk options and pricing, see [Managed Disks pricing](#).
- SQL Database optimizes costs with serverless compute and Hyperscale storage resources that automatically scale. For more information about SQL Database options and pricing, see [Azure SQL Database pricing](#).

Use the [Pricing calculator](#) to estimate costs for your implementation of this solution.

## Contributors

*This article is maintained by Microsoft. It was originally written by the following contributors.*

Principal author:

- [Jonathon Frost](#) | Principal Software Engineer

*To see non-public LinkedIn profiles, sign in to LinkedIn.*

## Next steps

- For more information, please contact [legacy2azure@microsoft.com](mailto:legacy2azure@microsoft.com).
- [What is Azure ExpressRoute](#)
- [What is Azure Virtual Network](#)
- [Introduction to Azure managed disks](#)
- [What is Azure Private Link](#)
- [What is Azure SQL Database](#)
- [What is Azure Files](#)

## Related resources

- [Azure mainframe and midrange architecture concepts and patterns](#)

- Rehost mainframe applications to Azure with Raincode compilers
- Refactor IBM z/OS mainframe Coupling Facility (CF) to Azure
- Unisys mainframe migration
- IBM z/OS mainframe migration with Avanade AMT
- High-volume batch transaction processing
- Modernize mainframe & midrange data

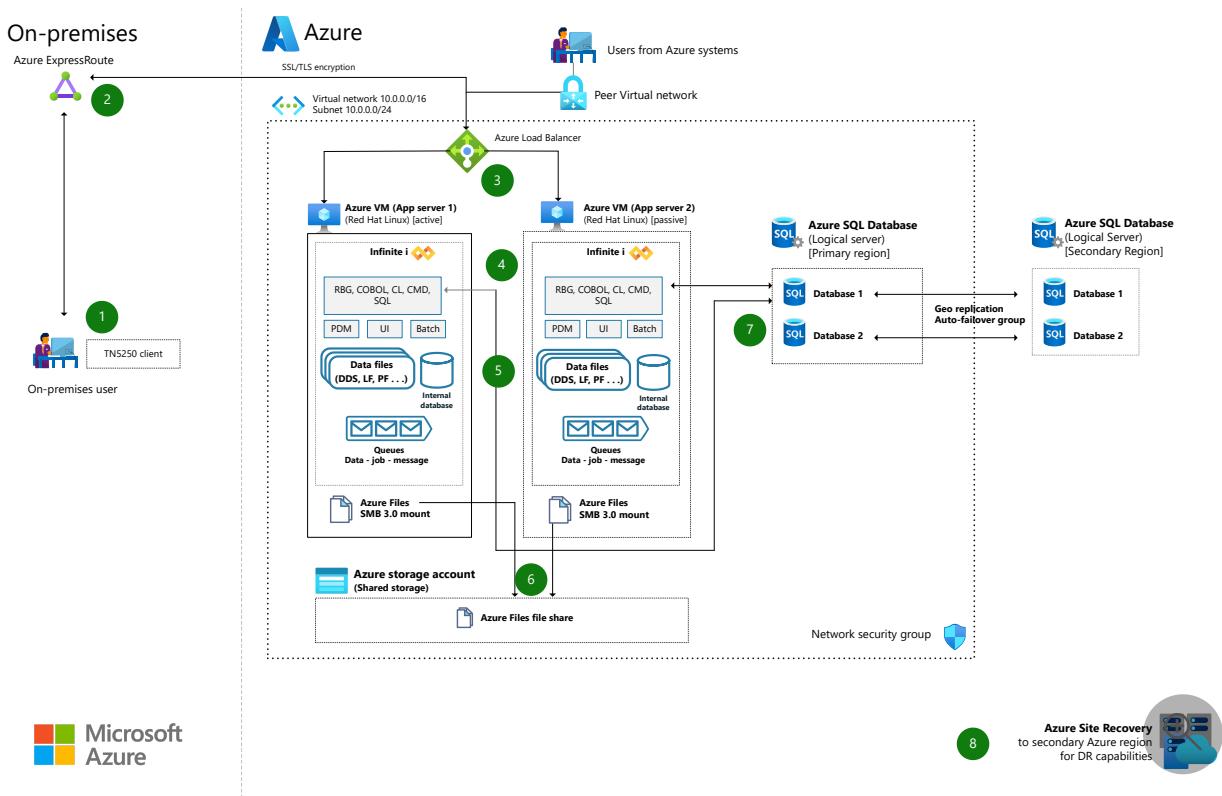
# IBM System i to Azure using Infinite i

Azure Virtual Machines

Azure SQL Database

The Infinite i suite is from Microsoft partner Infinite Corporation. The architecture described here uses it to migrate System i workloads to Azure. It converts RPG and COBOL source code to object code that runs natively on x86 virtual machines (VMs). Application screens and interactions work as before, thus minimizing user retraining. After migration, you maintain programs as usual by making changes to the source code.

## Architecture



Download a [Visio file](#) of this architecture.

## Workflow

1. TN5250 web terminal emulation provides user access to Azure over an SSL/TLS encrypted connection.
2. Azure ExpressRoute provides a dedicated high-speed connection between on-premises and Azure resources.

3. Infinite i application servers run the migrated workloads. Each server runs in its own Microsoft Azure Virtual Machines VM. The architecture uses two or more VMs for high availability, and Azure Load Balancer controls inbound and outbound network traffic. Infinite i supports an active-passive configuration (one active VM, one standby VM).
4. The compilers translate System i source code to 64-bit object code that runs on Azure x86 VMs.
5. An Infinite i internal database emulates the behavior of a DB2/400 database, including features such as physical files, logical files, multi-member files, joins, triggers, referential integrity, commitment control, and journaling. When an application runs on Azure, it accesses data as it did in the AS/400 environment, with no code changes required. Infinite i provides internal database connectors (ODBC and JDBC) for connecting to physical and logical files in the internal database.
6. Azure Files provides file shares to implement Infinite i files. Mounting a file share on the Azure VM gives programs direct access to the files. The file share also holds load modules and log files.
7. Instead of the internal database that step 5 describes, you can migrate the DB2/400 database to a standard SQL database. The database options are: SQL Server, Azure SQL, Oracle, and MySQL. These options support the same features as the internal database. When Infinite i migrates the database, it creates a database schema that maps physical files to tables and logical files to views.
8. Azure Site Recovery provides disaster recovery.

## Components

The architecture uses these components:

- [Azure Virtual Machines](#) VMs are on-demand, scalable computing resources that give you the flexibility of virtualization but eliminate the maintenance demands of physical hardware. The operating system choices include Windows and Linux. The VMs are an on-demand and scalable resource.
- [Azure Virtual Machine Scale Sets](#) is automated and load-balanced VM scaling that simplifies management of your applications and increases availability.
- [Azure Virtual Network](#) is a secure private network in the cloud. It connects VMs to one another, to the internet, and to on-premises networks.
- [Azure Private Link](#) carries private connections to Azure services.
- [Azure load balancing services](#) scale VMs for high availability and high performance. This architecture uses [Load Balancer](#), which provides low-latency balancing of traffic among VMs and across multi-tiered hybrid apps.

- [Azure Disk Storage](#) is highly durable and high-performance block storage for Azure VMs. There are four disk storage options for the cloud: Ultra Disk SSD Managed Disks, Premium SSD Managed Disks, Standard SSD Managed Disks, and Standard HDD Managed Disks.
- [Azure Files](#) offers simple, secure, and serverless enterprise-grade file shares in the cloud. The shares support access by the industry-standard Server Message Block (SMB) and Network File System (NFS) protocols. They can be mounted concurrently by cloud and on-premises deployments of Windows, Linux, and macOS.
- [Azure ExpressRoute](#) carries private connections between on-premises infrastructure and Azure datacenters.
- [Azure SQL](#) is a family of SQL cloud databases that provides a unified experience for your entire SQL portfolio, and a wide range of deployment options from edge to cloud.
- [Azure SQL Database](#), part of the Azure SQL family, is a fully managed platform as a service (PaaS) database engine. It handles most database management functions, such as upgrading, patching, backups, and monitoring, without your involvement. Azure SQL Database is always running on the latest stable version of the SQL Server database engine and patched OS, with 99.99 percent availability.

## Scenario details

You can easily migrate your System i and AS/400 workloads to Azure. The migrated workloads will match or improve performance and availability, at lower cost and with opportunities to modernize.

To migrate your applications, you compile them with the Infinite i suite. After deployment on Infinite i on Azure, the applications run as they did on the System i platform. The Infinite i runtime environment provides everything you need to run jobs and execute control language commands in a Linux environment.

There are compilers and translators for these technologies: RPG, RPG/ILE, RPG/Free, COBOL, Control Language Programs (CLP), and Data Description Specifications (DDS).

The Infinite i suite is from Microsoft partner Infinite Corporation. The architecture described here uses it to migrate System i workloads to Azure. It converts RPG and COBOL source code to object code that runs natively on x86 virtual machines (VMs). Application screens and interactions work as before, thus minimizing user retraining. After migration, you maintain programs as usual by making changes to the source code.

The benefits of the Infinite i environment include:

- Easy migration of System i workloads to Azure.
- Conversion of tape archives for backup and regulatory compliance.
- Application screens work as before. You have the option of updating the screens to web-based user interfaces.
- The Infinite internal database that holds your data emulates DB2/400. You have the option of migrating to a standard SQL database instead, with minor code changes or none at all.
- Your savings on licensing and maintenance significantly reduces your total cost of ownership.
- On Azure you have faster and lower-cost options for disaster recovery than you have on System i.

## Potential use cases

Use this architecture to easily migrate IBM System i and AS/400 workloads to Azure, and to modernize them and reduce costs.

## Considerations

The following considerations apply to this solution.

## Availability

The architecture accommodates redundancy and disaster recovery for high availability:

- [Azure Site Recovery](#) disaster recovery service protects against major outages by minimizing downtime and data loss, resulting in low impact recoveries from major failures. The service is dependable, cost-effective, and easy to deploy.
- For more information on various availability options, see [Availability options for Azure Virtual Machines](#).

Take these steps to improve availability:

- Use [Azure Availability Zones](#) to protect against infrastructure disruptions by eliminating all single points of failure. The SLA for VMs is for 99.99% uptime.
- Use an availability set, which is a grouping of VMs, for redundancy and availability. See [Availability sets overview](#) for more information.
- For increased availability, use Virtual Machine Scale Sets to set up a group of load-balanced VMs that make up an Azure Virtual Machine Scale Set.
- [Azure load balancing services](#) provide scaling for high availability and high performance.

# Operations

- The Infinite i deployment methodology calls for converting and testing workloads before migrating them to the Azure platform.
- When you move workloads to Azure, you can use Azure services such as Availability Zones, scale sets, and [Azure Site Recovery](#).
- Azure DevOps can help manage the migration.
- Consider using [Azure Resource Manager templates \(ARM templates\)](#) for scripted deployment, and for monitoring and alerting capabilities.

# Performance

- Azure services, including VMs, scale to meet desired performance.
- The Infinite i migration design process considers the performance characteristics of the workloads running on System i, and selects the right configuration of Azure services for the desired performance on Azure.
- Infinite i can take advantage of Azure scale sets to add capacity as needed.
- The architecture is designed to accommodate parallel processing of independent transactions.
- For this architecture, Premium SSDs or Ultra Disk SSDs are usually a good choice.

# Security

- Infinite i migrates the System i user-based access roles to Azure.
- The Infinite i runtime environment provides the same level of security on Azure as the System i environment provided.
- Azure security best practices can further protect the overall application environment.

# Cost optimization

The Infinite i solution keeps costs at a minimum to lower your total cost of ownership:

- The migration to Azure eliminates IBM licensing and maintenance costs.
- Linux has lower implementation costs than IBM platforms.
- The autoscale feature of PaaS services does scaling-on-demand to minimize costs.

To estimate the cost of implementing this solution, use the [Pricing calculator](#).

Here are pricing considerations for specific components:

- [Windows VM pricing](#) and [Linux VM pricing](#) depend on your compute capacity.

- For [ExpressRoute](#), you pay a monthly port fee and outbound data transfer charges.
- [Azure Storage](#) costs depend on data redundancy options and volume.
- [Azure Files](#) pricing depends on many factors: data volume, data redundancy, transaction volume, and the number of file sync servers that you use.
- For Premium SSD or Ultra SSD managed storage disks pricing, see [Managed Disks pricing](#).
- There are no upfront costs for [Azure SQL Database](#); you pay for resources as used.
- For [Site Recovery](#), you pay for each protected instance.
- These services are free with your Azure subscription, but you pay for usage and traffic:
  - [Load Balancer](#).
  - For [Azure Virtual Network](#), IP addresses carry a nominal charge.

## Contributors

*This article is maintained by Microsoft. It was originally written by the following contributors.*

Principal author:

- [Jonathon Frost](#) | Principal Software Engineer

*To see non-public LinkedIn profiles, sign in to LinkedIn.*

## Next steps

- For more information, contact [legacy2azure@microsoft.com](mailto:legacy2azure@microsoft.com).
- Infinite i from partner Infinite Corporation:
  - [Overview](#)
  - [Migrate Legacy Cold Storage AS/400](#)
  - [Infinite Cloud: beautiful screens from IBM i / AS400 green screens](#)
- Optimizing costs:
  - [Microsoft Azure Well-Architected Framework](#) has information about cost optimization for [VM instances](#).
  - [Checklist - Optimize cost](#)
  - [Virtual machines](#)

## Related resources

- Understand data store models
- Migrating IBM system workloads:
  - [High-volume batch transaction processing](#)
  - [IBM z/OS mainframe migration with Avanade AMT](#)
  - [Micro Focus Enterprise Server on Azure VMs](#)
  - [Refactor IBM z/OS mainframe Coupling Facility \(CF\) to Azure](#)
  - [Mainframe access to Azure databases](#)
  - [Replicate and sync mainframe data in Azure](#)
  - [Migrate IBM mainframe applications to Azure with TmaxSoft OpenFrame](#)
- IBM System i (AS/400) information:
  - [IBM Power Systems Servers: Powering the hybrid enterprise ↗](#)
  - [IBM i: A platform for innovators, by innovators ↗](#)
  - [IBM Power System case studies ↗](#)
  - [IBM Power Systems: enterprise servers ↗](#)

# IBM z/OS mainframe migration with Avanade AMT

Azure Load Balancer

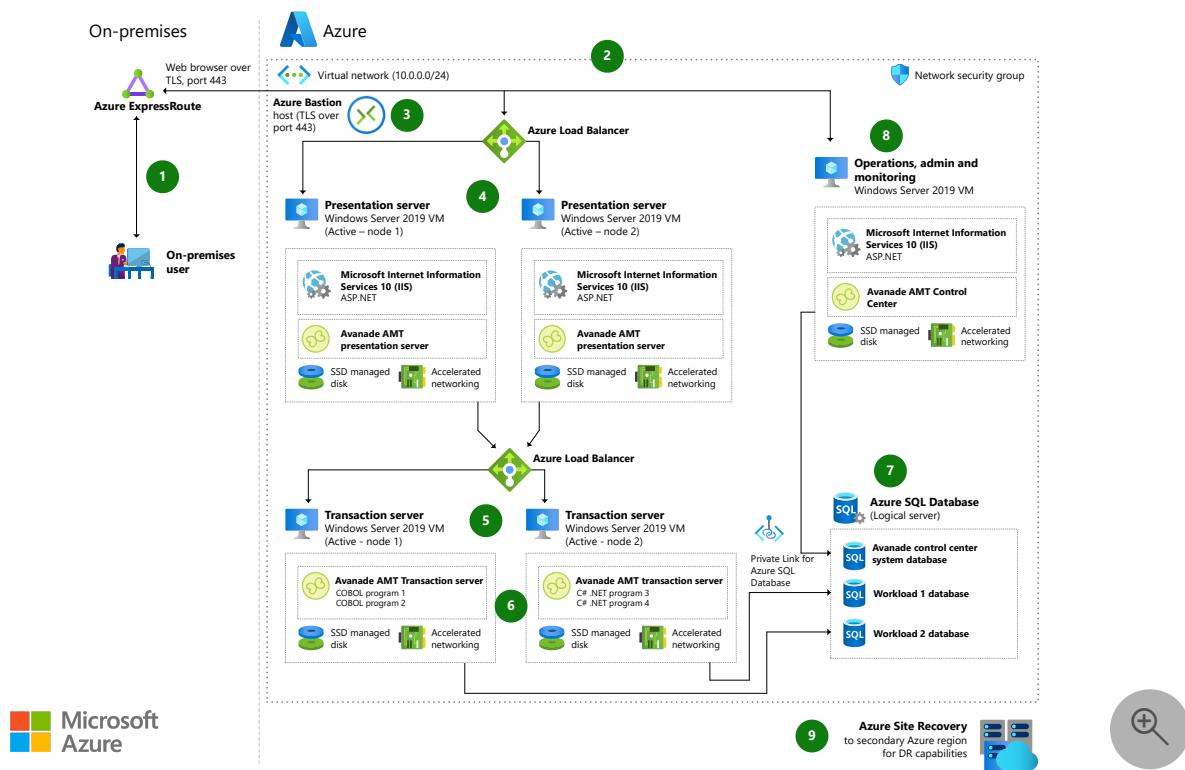
Azure SQL Database

Azure Virtual Machines

Azure Virtual Network

This article describes how [Avanade's Automated Migration Technology](#) (AMT) migrates an IBM z/OS mainframe system to the Azure cloud. The AMT framework converts proprietary IBM z/OS mainframe applications into native .NET applications that run on Windows Server OS virtual machines (VMs). On-premises mainframe resources migrate to cost-effective, scalable, secure Azure infrastructure-as-a-service (IaaS) and platform-as-a-service (PaaS) environments.

## Architecture



Download a [Visio file](#) of this architecture.

## Workflow

The preceding diagram shows how the typical components of an IBM z/OS mainframe system can map and migrate to Azure capabilities.

1. A web browser to access Azure resources replaces standard mainframe protocols like HTTPS and [TN3270 terminal emulation](#) for demand and online users. Users access web-based applications over a private Azure ExpressRoute connection through Transport Layer Security (TLS) port 443.
2. For security and performance, this solution deploys all Azure resources in an Azure Virtual Network, with a network security group to help manage traffic.
3. For admin access to the Azure VMs, Azure Bastion hosts maximize security by minimizing open ports.
4. AMT converts mainframe presentation loads to VM server farms. Two sets of two VMs run the web and application layers. The VMs use Premium SSD or Ultra managed disks with Accelerated Networking for high performance.

Azure Load Balancer fronts the VMs running the web and application layers in an *active-active* arrangement to spread query traffic.

Presentation layer code runs in IIS and uses ASP.NET to maintain the z/OS mainframe user-interface screens. You can leave web applications' presentation layers unchanged, to minimize user retraining, or you can update the presentation layers with modern user experience frameworks.

5. Mainframe batch and transaction loads convert to sufficient server farms to handle this type of work. Another Azure Load Balancer fronts the transaction servers to distribute the traffic.
6. Application code converts to AMT COBOL or directly to .NET C#. AMT maintains the original code structure to use as a baseline or for future edits. If code needs changing or editing, AMT can maintain and reprocess the original code, or you can edit the converted C# code directly to advance the code base to new standards.
7. AMT automates migrating all DB2, IMS, and IDMS hierarchical, network, or relational databases to Azure SQL. AMT Transform converts DMS and RDMS schemas to SQL, and converts [Job Control Language \(JCL\)](#) and [Rexx](#) scripts to VBScript or Windows PowerShell. Azure Private Link for Azure SQL Database provides a private, direct connection from the Azure VMs to Azure SQL Database.

AMT Transform also converts all binary or indexed [Virtual Storage Access Method \(VSAM\)](#), flat files, and virtual tape files to Azure Files storage.

8. Workload automation, scheduling, reporting, and system monitoring functions that are Azure-capable can keep their current platforms. This example uses AMT Control Center for operations.

The system can support printers and other legacy system output devices if they have IP addresses connected to the Azure network.

9. Azure Site Recovery mirrors the Azure VMs to a secondary Azure region for quick failover and disaster recovery in case of Azure datacenter failure.

## Components

- [Azure ExpressRoute](#) extends your on-premises networks into the Microsoft cloud over a private connection facilitated by a connectivity provider. You can use ExpressRoute to establish connections to cloud services like Azure and Microsoft 365.
- [Azure Bastion](#) is a fully managed platform as a service (PaaS) that you provision inside your virtual network. Azure Bastion provides secure and seamless Remote Desktop Protocol (RDP) and secure shell (SSH) connectivity to the VMs in your virtual network directly from the Azure portal over TLS.
- [Azure Virtual Machines](#) provides on-demand, scalable computing resources that give you the flexibility of virtualization without having to buy and maintain physical hardware.
- [Azure Virtual Network](#) is the fundamental building block for Azure private networks. With Virtual Network, Azure resources like VMs can securely communicate with each other, the internet, and on-premises networks. An Azure Virtual Network is similar to a traditional network on premises, but with Azure infrastructure benefits like scalability, availability, and isolation.
- [Virtual network interfaces](#) let Azure VMs communicate with internet, Azure, and on-premises resources. You can add several network interface cards to one Azure VM, so child VMs can have their own dedicated network interface devices and IP addresses.
- [Azure Managed Disks](#) provides block-level storage volumes that Azure manages on Azure VMs. The available types of disks are Ultra disks, Premium solid-state drives (SSDs), Standard SSDs, and Standard hard disk drives (HDDs).
- [Azure Files](#) offers fully managed file shares in an Azure Storage account that are accessible from the cloud or on-premises. Windows, Linux, and macOS deployments can mount Azure file shares concurrently, and access files via the industry standard Server Message Block (SMB) protocol.

- [Azure SQL Database](#) is a fully managed PaaS database engine that is always running on the latest stable version of SQL Server and patched OS, with 99.99% availability. SQL Database handles most database management functions like upgrading, patching, backups, and monitoring without user involvement. These PaaS capabilities let you focus on business critical, domain-specific database administration and optimization.
- [Azure Site Recovery](#) uses replication, failover, and recovery processes to help keep your applications running during planned and unplanned outages.

## Alternatives

The AMT Framework supports several methodologies to move client workloads to Azure:

- One migration method is to convert and move the entire mainframe system to Azure at once, saving interim mainframe maintenance and facility support costs. This approach carries some risk because all processes, like application conversion, data migration, and testing, must align for a smooth transition.
- A second methodology is to move applications from the mainframe to Azure gradually, with complete transition as the ultimate goal. This tactic provides savings per application, and lessons learned to convert each application can help with subsequent conversions.

Modernizing each application on its own schedule can be more relaxed than converting everything at once. If regaining time on the mainframe is a goal, the stepped method can provide more processing cycles on the mainframe as applications convert to Azure. Eventual starvation of the mainframe can highlight the need to retire the mainframe expense.

## Scenario details

Transforming proprietary legacy applications, infrastructures, and processes to standardized, benchmarked cloud technologies promotes agile DevOps principles and practices that are today's productivity norm. The transformation of legacy applications and infrastructures leads to more unified business and IT alignment.

[Avanade's Automated Migration Technology](#) (AMT) migrates an IBM z/OS mainframe system to the Azure cloud. The AMT framework converts proprietary IBM z/OS mainframe applications into native .NET applications that run on Windows Server OS virtual machines (VMs). On-premises mainframe resources migrate to cost-effective,

scalable, secure Azure infrastructure-as-a-service (IaaS) and platform-as-a-service (PaaS) environments.

AMT provides an accelerated move into Azure without rewriting application code or redesigning data architecture. The migration framework converts legacy code to C#, while maintaining the source code in its original form. Application user interfaces and interactions can remain unchanged, minimizing the need for user retraining.

## Potential use cases

Many scenarios can benefit from Avanade AMT migration. Possibilities include the following cases:

- Modernizing infrastructure to avoid the high costs, limitations, and rigidity of mainframes.
- Moving mainframe workloads to the cloud without the side effects of a complete redevelopment.
- Migrating mission-critical applications to the cloud while maintaining continuity with on-premises mainframe applications.
- Implementing flexible horizontal and vertical scalability.
- Deploying high availability (HA) and disaster recovery (DR) capabilities.

## Considerations

The following considerations apply to this solution:

### Availability

[Azure Site Recovery](#)  mirrors the Azure VMs to a secondary Azure region for quick failover and DR if there is Azure datacenter failure. [Azure auto-failover group replication](#) provides data protection by managing the database replication and failover to the secondary region.

### Resiliency

Azure Load Balancer builds resiliency into this solution. If one presentation or transaction server fails, the other servers behind the load balancer take on the workload.

### Scalability

- Avanade AMT has proven single-application scalability equivalent to at least 28,000 million IBM mainframe instructions per second (MIPS).
- Each set of servers can scale out to provide more throughput. For information, see [Virtual machine scale sets](#).

## Security

- This solution uses an Azure network security group to manage traffic between Azure resources. For more information, see [Network security groups](#).
- [Private Link](#) provides a private, direct connection isolated to the Azure networking backbone from the Azure VMs to Azure SQL Database.
- [Azure Bastion](#) maximizes admin access security by minimizing open ports.

## Cost optimization

Azure helps you avoid unnecessary costs by identifying resource needs, analyzing spending over time, and scaling to meet business needs without overspending. Avanade AMT in Azure runs on Windows VMs, which help you optimize costs by turning off VMs when not in use and scripting schedules for known usage patterns.

- Azure services like Virtual Network, Load Balancer, and Azure Bastion are free with your Azure subscription. You pay for usage and traffic.
- With Azure Site Recovery, you pay for each protected instance. If VMs in server sets are clones, only one instance needs to participate in Site Recovery.
- Azure SQL Database uses [hyperscale or business critical](#) tiers in this solution, for high input/output operations per second (IOPS) and high uptime SLA. For pricing information, see [Azure SQL Database pricing](#).
- This solution works best with Premium SSD or Ultra Managed Disks. For pricing information, see [Managed Disks pricing](#).

To estimate and calculate costs for your implementation of this solution, use the [Azure pricing calculator](#).

## Contributors

*This article is maintained by Microsoft. It was originally written by the following contributors.*

Principal author:

- [Philip Brooks](#) | Senior Technical Program Manager

*To see non-public LinkedIn profiles, sign in to LinkedIn.*

## Next steps

- For more information, please contact [legacy2azure@microsoft.com](mailto:legacy2azure@microsoft.com).
- See the [Microsoft Azure Well-Architected Framework](#) for more information about cost optimization for [VM instances](#).
- Visit the [Azure Marketplace](#) for information about [Asysco AMT GO](#).
- See the blog post [MIPS Equivalent Sizing for IBM CICS COBOL Applications Migrated to Microsoft Azure](#).

## Related resources

- [Refactor IBM z/OS mainframe Coupling Facility \(CF\) to Azure](#).

# IBM z/OS online transaction processing on Azure

Azure Front Door

Azure Traffic Manager

Azure Kubernetes Service (AKS)

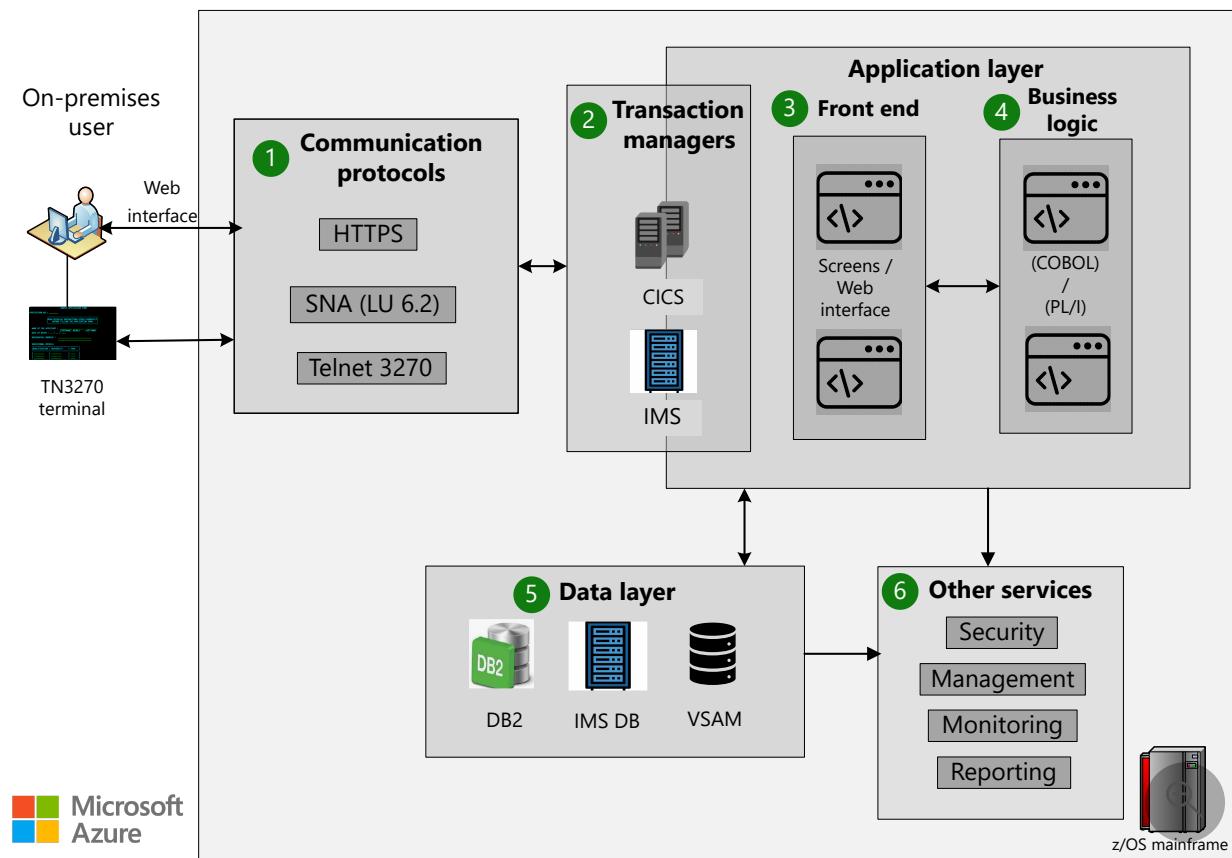
Azure Spring Apps

Azure Cache for Redis

Online transaction processing (OLTP) systems interact directly with users and are the face of the business. With a dynamically adaptable infrastructure, businesses can realize and launch their products quickly to delight their users.

## Architecture

The following diagram shows the architecture of the workload to be migrated, an OLTP system running on a z/OS mainframe:



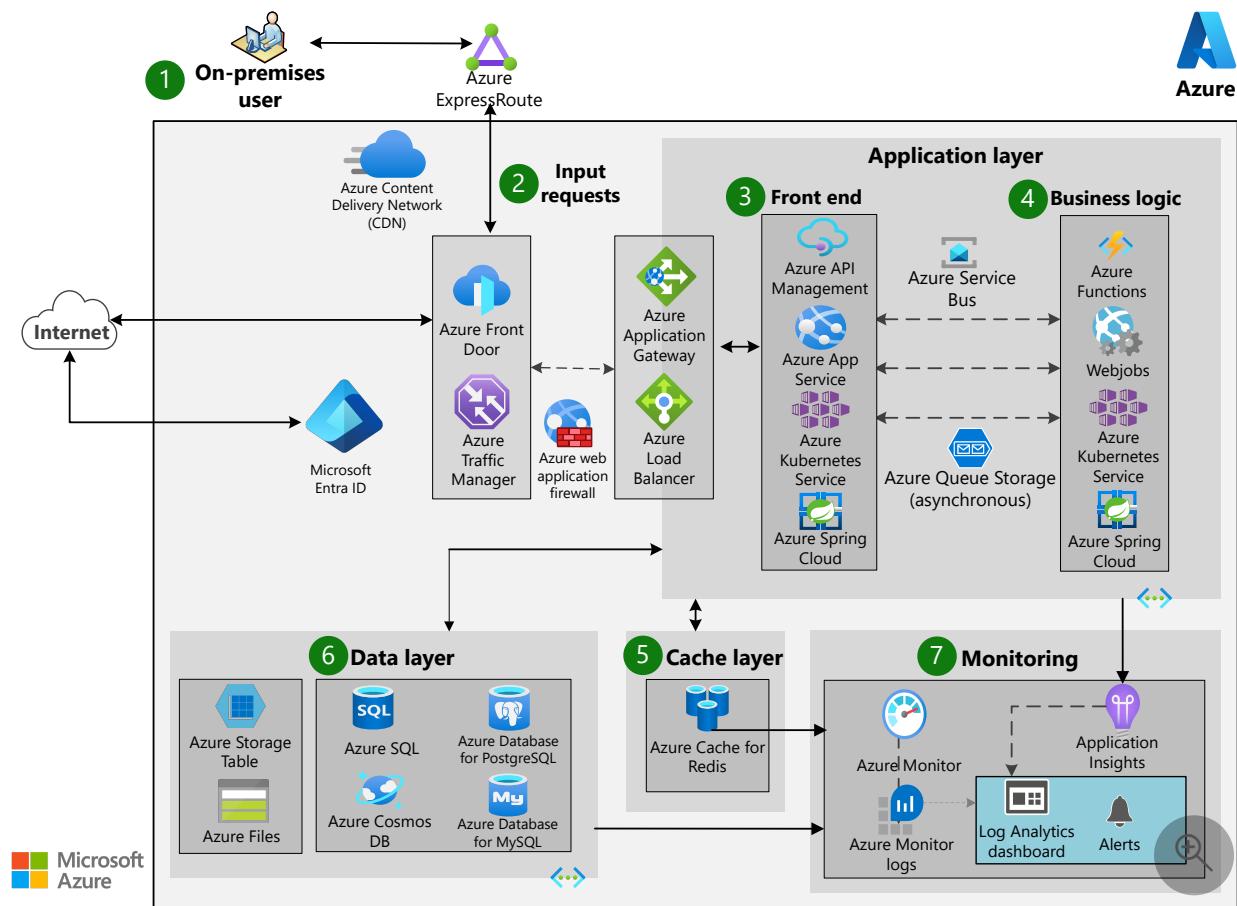
Download a [Visio file](#) of this architecture.

## Workflow

The following workflow corresponds to the preceding diagram:

1. Users connect to the mainframe over TCP/IP using standard mainframe protocols like TN3270 and HTTPS.
2. The transaction managers interact with the users and invoke the application to satisfy user requests.
3. In the front end of the application layer, users interact with the CICS/IMS screens or with web pages.
4. The transaction managers use the business logic written in COBOL or PL/1 to implement the transactions.
5. Application code uses storage capabilities of the data layer, typically DB2, IMS DB, or VSAM.
6. Along with transaction processing, other services provide authentication, security, management, monitoring, and reporting. These services interact with all other services in the system.

Here, we see how to migrate this architecture to Azure.



Download a [Visio file](#) of this architecture.

1. Mainframe users are familiar with 3270 terminals and on-premises connectivity. In the migrated system, they interact with Azure applications via public internet or via a private connection implemented with Azure ExpressRoute. Microsoft Entra ID provides authentication.

2. Input requests go to a global load balancer service, like Azure Front Door or Azure Traffic Manager. The load balancer can serve a geographically spread user base. It routes the requests according to rules defined for the supported workloads. These load balancers can coordinate with Azure Application Gateway or Azure Load Balancer for load balancing of the application layer. The Azure Content Delivery Network service caches static content in edge servers for quick response, secured using the Web Application Firewall (WAF) service.
3. The front end of the application layer uses Azure services like Azure App Service to implement application screens and to interact with users. The screens are migrated versions of the mainframe screens.
4. COBOL and PL/1 code in the back end of the application layer implements the business logic. The code can use services like Azure Functions, WebJobs, and Azure Spring Apps microservices. Applications can run in an Azure Kubernetes Service (AKS) container.
5. An in-memory data store accelerates high-throughput OLTP applications. One such store is In-Memory OLTP, a feature of Azure SQL Database and Azure SQL Managed Instance. Another is Azure Cache for Redis.
6. The data layer can include, for example:
  - a. Files, tables, and blobs implemented using Azure Storage services.
  - b. Relational databases from the Azure SQL family.
  - c. Azure implementations of the PostgreSQL and MySQL open-source databases.
  - d. Azure Cosmos DB, a NoSQL database.These stores hold data migrated from the mainframe for use by the application layer.
7. Azure native services like Application Insights and Azure Monitor proactively monitor the health of the system. You can integrate the Monitor logs using an Azure dashboard.

## Components

This architecture consists of several Azure cloud services and is divided into four categories of resources: networking and identity, application, storage, and monitoring. The services for each and their roles are described in the following sections.

### Networking and identity

- [Azure ExpressRoute](#) carries private connections between on-premises infrastructure and Azure datacenters.
- [Microsoft Entra ID](#) is an identity and access management service that can synchronize with an on-premises directory.
- [Azure Front Door](#) provides global HTTP load balancing with instant failover. Its caching option can quicken delivery of static content.
- [Azure Traffic Manager](#) directs incoming DNS requests based on your choice of traffic routing methods.
- [Azure Web Application Firewall](#) helps protect web apps from malicious attacks and common web vulnerabilities, such as SQL injection and cross-site scripting.
- [Azure Content Delivery Network \(CDN\)](#) caches static content in edge servers for quick response, and uses network optimizations to improve response for dynamic content. CDN is especially useful when the user base is global.
- [Azure Application Gateway](#) is an application delivery controller service. It operates at layer 7, the application layer, and has various load-balancing capabilities.
- [Azure Load Balancer](#) is a layer 4 (TCP, UDP) load balancer. In this architecture, it provides load balancing options for Spring Apps and AKS.

## Application

- [Azure API Management](#) supports the publishing, routing, securing, logging, and analytics of APIs. You can control how the data is presented and extended, and which apps can access it. You can restrict access to your apps, or allow third parties.
- [Azure App Service](#) is a fully managed service for building, deploying, and scaling web apps. You can build apps using .NET, .NET Core, Node.js, Java, Python, or PHP. The apps can run in containers or on Windows or Linux. In a mainframe migration, the front-end screens or web interface can be coded as HTTP-based REST APIs. They can be segregated as per the mainframe application, and can be stateless to orchestrate a microservices-based system.
- WebJobs is a feature of Azure App Service that runs a program or script in the same instance as a web app, API app, or mobile app. A web job can be a good choice for implementing sharable and reusable program logic. For technical information, see [Run background tasks with WebJobs in Azure App Service](#).
- [Azure Kubernetes Service \(AKS\)](#) is a fully managed Kubernetes service for deploying and managing containerized applications. AKS simplifies deployment of a managed AKS cluster in Azure by offloading the operational overhead to Azure.
- [Azure Spring Apps](#) is a fully managed Spring service, jointly built and operated by Microsoft and VMware. With it, you can easily deploy, manage, and run Spring

microservices, and write Spring applications using Java or .NET.

- [Azure Service Bus](#) is a reliable cloud messaging service for simple hybrid integration. Service Bus and Storage queues can connect the front end with the business logic in the migrated system.
- [Azure Functions](#) provides an environment for running small pieces of code, called functions, without having to establish an application infrastructure. You can use it to process bulk data, integrate systems, work with IoT, and build simple APIs and microservices. With microservices, you can create servers that connect to Azure services and are always up to date.
- [Azure Cache for Redis](#) is a fully managed in-memory caching service for sharing data and state among compute resources. It includes both the open-source Redis (OSS Redis) and a commercial product from Redis Labs (Redis Enterprise) as a managed service. You can improve performance of high-throughput OLTP applications by designing them to scale and to make use of an in-memory data store such as Azure Cache for Redis.

## Storage

- [Azure Storage](#) is a set of massively scalable and secure cloud services for data, apps, and workloads. It includes [Azure Files](#), [Azure Table Storage](#), and [Azure Queue Storage](#). Azure Files is often an effective tool for migrating mainframe workloads.
- [Azure SQL](#) is a family of SQL cloud databases that provides flexible options for application migration, modernization, and development. The family includes:
  - [SQL Server on Azure Virtual Machines](#)
  - [Azure SQL Managed Instance](#)
  - [Azure SQL Database](#)
  - [Azure SQL Edge](#)
- [Azure Cosmos DB](#) is a fully managed NoSQL database service with open-source APIs for MongoDB and Cassandra. A possible application is to migrate mainframe non-tabular data to Azure.
- [Azure Database for PostgreSQL](#) is a fully managed, intelligent, and scalable PostgreSQL that has native connectivity with Azure services.
- [Azure Database for MySQL](#) is a fully managed, scalable MySQL database.
- In-Memory OLTP is a feature of [Azure SQL Database](#) and [Azure SQL Managed Instance](#) that provides fast in-memory data storage. For technical information, see [Optimize performance by using in-memory technologies in Azure SQL Database and Azure SQL Managed Instance](#).

## Monitoring

- [Azure Monitor](#) collects, analyzes, and acts on personal data from your Azure and on-premises environments.
- Log Analytics is a tool in the Azure portal used to query Monitor logs using a powerful query language. You can work with the results of your queries interactively or use them with other Azure Monitor features such as log query alerts or workbooks. For more information, see [Overview of Log Analytics in Azure Monitor](#).
- Application Insights is a feature of Monitor that provides code-level monitoring of application usage, availability, and performance. It monitors the application, detects application anomalies such as mediocre performance and failures, and sends personal data to the Azure portal. You can also use Application Insights for logging, distributed tracing, and custom application metrics.
- Azure Monitor Alerts are a feature of Monitor. For more information, see [Create, view, and manage metric alerts using Azure Monitor](#).

## Scenario details

With ever-evolving business needs and data, applications must produce and scale without creating infrastructure issues. This example workload shows how you can migrate a z/OS mainframe OLTP application to a secure, scalable, and highly available system in the cloud, by using Azure platform as a service (PaaS) services. Such a migration helps businesses in finance, health, insurance, and retail to minimize application delivery timelines, and it helps reduce the costs of running the applications.

## Potential use cases

This architecture is ideal for OLTP workloads that have these characteristics:

- They serve an international user base.
- Their usage varies greatly over time, so they benefit from flexible scaling and usage-based pricing.

## Considerations

These considerations implement the pillars of the Azure Well-Architected Framework, which is a set of guiding tenets that can be used to improve the quality of a workload. For more information, see [Microsoft Azure Well-Architected Framework](#).

## Reliability

Reliability ensures your application can meet the commitments you make to your customers. For more information, see [Overview of the reliability pillar](#).

- This OLTP architecture can be deployed in multiple regions and can have a geo-replicated data layer.
- The Azure database services support zone redundancy and can fail over to a secondary node if an outage occurs, or to allow for maintenance activities.

## Security

Security provides assurances against deliberate attacks and the abuse of your valuable data and systems. For more information, see [Overview of the security pillar](#).

- ExpressRoute creates a private connection to Azure from an on-premises environment. You can also use site-to-site VPN.
- Microsoft Entra ID can authenticate resources and control access using Azure role-based access control.
- Database services in Azure support various security options like data encryption at rest.
- For general guidance on designing secure solutions, see [Overview of the security pillar](#).

## Cost optimization

Cost optimization is about looking at ways to reduce unnecessary expenses and improve operational efficiencies. For more information, see [Overview of the cost optimization pillar](#).

Use the [Azure Pricing Calculator](#) to estimate costs for your implementation.

## Operational excellence

Operational excellence covers the operations processes that deploy an application and keep it running in production. For more information, see [Overview of the operational excellence pillar](#).

- This scenario uses Azure Monitor and Application Insights to monitor the health of the Azure resources. You can set alerts for proactive management.
- For guidance on resiliency in Azure, see [Designing reliable Azure applications](#).

## Performance efficiency

Performance efficiency is the ability of your workload to scale to meet the demands placed on it by users in an efficient manner. For more information, see [Performance efficiency pillar overview](#).

- This architecture uses Azure PaaS services like App Service, which has autoscaling capabilities.
- For guidance on autoscaling in Azure, see [Autoscaling](#).

## Contributors

*This article is maintained by Microsoft. It was originally written by the following contributors.*

Principal author:

- [Ashish Khandelwal](#) | Principal Engineering Architecture Manager
- [Nithish Aruldoss](#) | Engineering Architect

*To see non-public LinkedIn profiles, sign in to LinkedIn.*

## Next steps

- For more information, contact [datasqlninja@microsoft.com](mailto:datasqlninja@microsoft.com).
- [Azure Database Migration Guides](#)

## Related resources

See the following related architectures and related technical information:

## Related architectures

- [High-volume batch transaction processing](#)
- [IBM z/OS mainframe migration with Avanade AMT](#)
- [Micro Focus Enterprise Server on Azure VMs](#)
- [Refactor IBM z/OS mainframe Coupling Facility \(CF\) to Azure](#)
- [Mainframe access to Azure databases](#)
- [Replicate and sync mainframe data in Azure](#)
- [Migrate IBM mainframe applications to Azure with TmaxSoft OpenFrame](#)

## Related technical information

- Run background tasks with WebJobs in Azure App Service
- Optimize performance by using in-memory technologies in Azure SQL Database and Azure SQL Managed Instance
- Azure Monitor overview
- Create, view, and manage metric alerts using Azure Monitor
- Create and share dashboards of Log Analytics data
- Overview of the security pillar

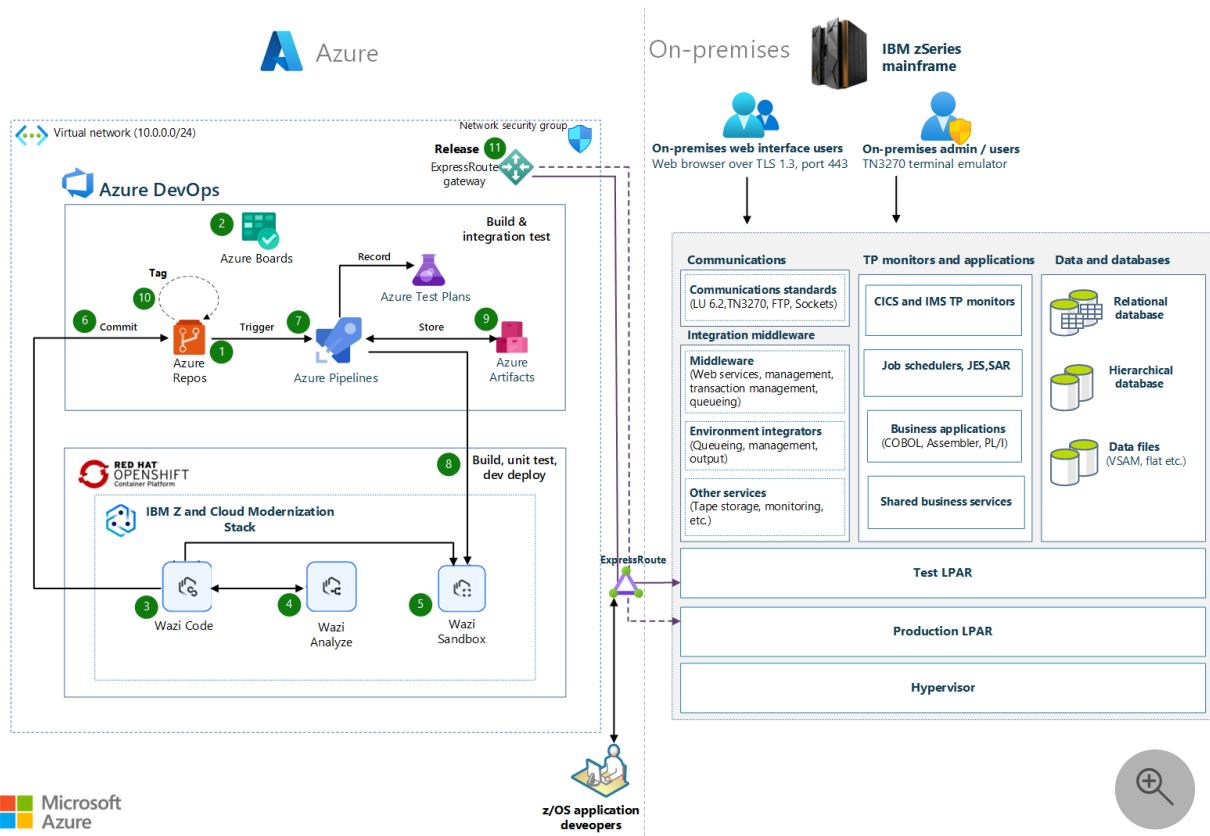
# Implement Azure DevOps for mainframe applications that use IBM Z and Cloud Modernization Stack

Azure DevOps   Azure Repos   Azure Pipelines   Azure Boards   Azure Test Plans

This article can help you get started with building Azure DevOps pipelines that run Git-based IBM Dependency Based Builds (DBBs) on z/OS. Although Azure DevOps is primarily used by distributed applications, many organizations are looking for consolidated solutions that integrate with modern IBM mainframe development workflows.

*Red Hat is a trademark of its respective company. No endorsement is implied by the use of this mark.*

## Architecture



Download a [Visio file](#) of this architecture.

# Workflow

1. The z/OS application source code is migrated into a Git-based source code management (SCM) system.
2. Azure Boards streamlines work item creation, assignment, status monitoring, and progress tracking.
3. Wazi Code simplifies the development of mainframe applications.
4. Wazi Analyze is used for code analysis and dependency resolution.
5. Unit testing of code changes can be tested manually against a developer-owned Wazi Sandbox environment.
6. Code changes are committed back to the Git repository from the web-based IDE.
7. Code changes pushed to the repository trigger a build via Azure Pipelines.
8. Automated builds, deployments, and tests are run against a containerized z/OS environment within the Wazi Sandbox.
9. In Azure Artifacts, successful builds are systematically stored to provide a centralized repository for versioned artifacts.
10. Production release scenarios follow a typical Git-based release automation process, starting with branching/tagging Git commits to be used in production.
11. When automation is triggered on release branches or tags, it initiates production deployment based on change management guidelines and the approval process in the on-premises z/OS environments in your datacenter.

# Components

- [Azure DevOps](#) provides developer services to help teams plan work, collaborate on code development, and build and deploy applications. Developers can work in the cloud by using Azure DevOps Services or on-premises by using Azure DevOps Server.
- [IBM Z and Cloud Modernization Stack](#) provides highly secure access to mainframe applications and data via APIs. It provides industry-standard tools that can help you implement DevOps, and supports modern languages.
- The [Wazi components](#) that are available in IBM Z and Cloud Modernization Stack enable the analysis, development, and testing of mainframe applications in an isolated way.
- [Red Hat OpenShift Container Platform](#) brings together services to reduce the friction of developing, modernizing, deploying, running, and managing applications. Built on Kubernetes, it provides a consistent experience across public cloud, on-premises, hybrid cloud, and edge architectures.
- [Azure Virtual Network](#) is the fundamental building block for your private network on Azure. It provides enhanced-security connections that enable Azure

resources, like Azure virtual machines, to communicate with each other, the internet, and on-premises networks. Virtual Network is similar to a traditional network that you might operate in your own datacenter, but it provides the additional benefits of the Azure infrastructure, like scale, availability, and isolation.

- [Azure ExpressRoute](#) enables you to extend your on-premises networks into the Microsoft Cloud over a private connection that's facilitated by a connectivity provider. You can use ExpressRoute to establish connections to Microsoft cloud services like Azure and Office 365.

## Alternatives

You can use [Azure VPN Gateway](#) to connect on-premises networks to Azure over the public internet by using an enhanced-security, encrypted connection. VPN Gateway can provide a cost-effective option for organizations that don't require the high bandwidth and low latency of ExpressRoute.

## Scenario details

Azure DevOps can help you develop and modernize your z/OS applications that use IBM Z and Cloud Modernization Stack. Doing so can result in increased agility and enable developers to work faster and more productively. You can modify your existing COBOL, PL/I, Java, or assembler programs by using any IDE.

## Azure DevOps

Azure DevOps provides developer services to help teams plan work, collaborate on code development, and build and deploy applications. Developers can work in the cloud by using Azure DevOps Services or on-premises by using Azure DevOps Server. (Azure DevOps Services was previously known as *Visual Studio Team Services*.)

Azure DevOps provides integrated features that you can access through your web browser or IDE client. You can use one or more of the following services, depending on your business needs:

- [Azure Repos](#) provides Git repositories or Team Foundation Version Control for source control of your code.
- [Azure Pipelines](#) provides build and release services to support continuous integration and continuous delivery (CI/CD) of your apps.
- [Azure Boards](#) provides a suite of Agile tools to support the planning and tracking of work, code defects, and issues by using Kanban and Scrum methods.

- [Azure Test Plans](#) provides tools to test your apps, including manual/exploratory testing and continuous testing.
- [Azure Artifacts](#) enables teams to share Maven, npm, and NuGet packages from public and private sources and integrate package sharing into CI/CD pipelines. By using Azure Artifacts to manage COBOL files, you can ensure that all developers in your organization have access to the correct versions of files and their dependencies.

## IBM Z and Cloud Modernization Stack with Azure DevOps

The Azure DevOps Git-based code repository and agent-based Azure pipelines can integrate with IBM DBB solutions on IBM zSystems for mainframe application development and build processes. DBB, available through the Wazi Code component of IBM Z and Cloud Modernization Stack, complements Azure DevOps by providing intelligent build, link, and edit capabilities. The combination enables developers working on large, complex applications to develop, build, and deliver in an agile manner.

You can use Azure Boards, which is independent of platform, to manage mainframe-based sprints. Azure DevOps runs CI pipelines that can construct and deploy mainframe applications that run in either batch or online mode, with or without Db2 and MQ.

By integrating [IBM UrbanCode Deploy](#) into Azure pipelines, you can get a comprehensive CI/CD pipeline for on-premises, cloud, or hybrid platforms that encompasses z/OS applications. The IBM Z and Cloud Modernization Stack, based on the Red Hat OpenShift platform, includes various capabilities to modernize mainframe applications. For development and test purposes, IBM Wazi includes Wazi Code (IDEs, DBB), Wazi Analyze, and Wazi Sandbox, which together provide mainframe developers with a cloud-native experience that doesn't require application installation.

Here are some additional benefits of using Azure DevOps together with IBM Z and Cloud Modernization Stack.

- Wazi Code provides an in-browser IDE that you can use to develop, build, and debug COBOL, PL/I, Assembler, Java, Rexx, and JCL applications. By using this approach, you can easily modify mainframe applications and take advantage of programming languages like Python, Node.js, and Go, which can all run on IBM z/OS. This approach enables you to easily adapt to new languages while still retaining the flexibility to work with preferred IDEs.
- [DevOps for zSystems](#), integrated with [Azure DevOps](#) solutions, spans Azure services and z/OS environments to orchestrate the development, integration, and deployment of applications across IBM zSystems and Azure.

- IBM zSystems development tools and processes are no longer siloed. Adopting Azure DevOps and adapting to IBM DBB blends mainframe repositories into a single source code repository across technology stacks, and an integrated change management and approval process with coordination of auditing and reporting across the enterprise.
- Solutions from Azure and IBM enable organizations to free themselves from the restrictions and increasing costs of legacy tools to focus on effective project management. IBM and Azure speed up transformations to the hybrid cloud, so you can deliver solutions quickly and efficiently.

## Potential use cases

- Implement an end-to-end CI/CD pipeline for on-premises, cloud, or hybrid platforms that include z/OS applications.
- Enable mainframe teams to work in parallel with distributed development teams, implement a single standard, and use common tools for DevOps and SDLC across the enterprise.
- Provide a consistent and familiar developer experience for z/OS applications together with a flexible dev and test infrastructure on Azure.
- Speed up application development and the development lifecycles of mainframe applications.

## Considerations

These considerations implement the pillars of the Azure Well-Architected Framework, which is a set of guiding tenets that you can use to improve the quality of a workload. For more information, see [Microsoft Azure Well-Architected Framework](#).

## Reliability

Reliability ensures your application can meet the commitments you make to your customers. For more information, see [Overview of the reliability pillar](#).

- Azure DevOps provides CI/CD pipelines that allow you to automate the build, testing, and deployment of your application. These pipelines can help you ensure that your application is always up to date and reliable.
- Azure DevOps provides collaboration tools that allow your team to work together effectively.
- RedHat OpenShift Container Platform provides automated deployment capabilities, which can help ensure that your applications are deployed consistently

and reliably.

- Using a standard and consistent approach to building, testing, and deploying applications across Azure and IBM zSystems can help developers reduce the risks associated with code changes.
- Azure DevOps and IBM Z and Cloud Modernization Stack enable application modernization with fit-for-purpose workload placement across environments, which reduces the risk involved in full migration scenarios.

## Security

Security provides assurances against deliberate attacks and the abuse of your valuable data and systems. For more information, see [Overview of the security pillar](#).

- Azure DevOps allows you to control access to your resources and data by using role-based access control (RBAC). RBAC helps to ensure that only authorized users can access your resources. For more information, see [Security best practices](#).
- Azure DevOps integrates with Microsoft Defender for Cloud, which provides additional security insights and recommendations that can help you identify potential security problems and take steps to address them.
- You can enhance and extend your current DevSecOps practices by taking advantage of the security capabilities of IBM zSystems. By doing so, you can effectively mitigate business risks, help safeguard application data, and help ensure the long-term security for your systems.

## Cost optimization

Cost optimization is about reducing unnecessary expenses and improving operational efficiencies. For more information, see [Overview of the cost optimization pillar](#).

- Azure DevOps provides automation capabilities that help you reduce the time and effort required by manual tasks. By automating tasks, you can optimize costs and improve efficiency.
- Azure DevOps integrates with other Azure services, like Azure Monitor and Azure Advisor. This integration can help you optimize costs by providing insights into resource usage and recommendations for cost optimization. Use the [Azure pricing calculator](#) to estimate the cost of implementing your solution.
- IBM Z and Cloud Modernization Stack and Azure DevOps reduce the need for customized z/OS tools by allowing organizations to implement the same CI/CD toolchain and practices as the rest of the enterprise.

## Operational excellence

Operational excellence covers the operations processes that deploy an application and keep it running in production. For more information, see [Overview of the operational excellence pillar](#).

- Azure DevOps is a powerful platform that provides a comprehensive set of tools for managing the entire software development lifecycle. Overall, by using Azure DevOps, you can achieve operational excellence by improving collaboration, automating key tasks, monitoring performance, and streamlining the software development and deployment process.
- Wazi Sandbox, a personal z/OS environment, can be provisioned on Azure with all the required z/OS resources (IMS and CICS, for example) that enable developers to work in isolation, run initial builds, and conduct early tests to verify code changes.

## Performance efficiency

Operational excellence covers the operations processes that deploy an application and keep it running in production. For more information, see [Overview of the operational excellence pillar](#).

- You can ensure performance and stability of multi-tenant integration test environments by using isolated Wazi Sandbox instances to perform build and test operations earlier in the software development lifecycle.
- Azure DevOps can help you improve performance efficiency in several ways. It provides a robust CI/CD pipeline that can help automate the build, test, and deployment process. This pipeline helps reduce the time required to deploy new features and bug fixes, improve the quality of the application, and reduce the risk of errors during deployment.

## Contributors

*This article is maintained by Microsoft. It was originally written by the following contributors.*

Principal authors:

- [Bhaskar Bandam](#) | Transformation Specialist
- [Ivan Dovgan](#) | Senior Software Architect

Other contributors:

- [Mick Alberts](#) | Technical Writer
- [Bhuvi Vatsey](#) | Modernization Specialist

*To see non-public LinkedIn profiles, sign in to LinkedIn.*

## Next steps

For more information, contact [Legacy Migrations Engineering](#).

We also recommend these resources:

- [What is Azure DevOps?](#)
- [IBM Z and Cloud Modernization Stack ↗](#)
- [White paper on Azure DevOps for zSystems ↗](#)

## Related resources

- [Azure mainframe and midrange architecture design](#)
- [Mainframe migration overview](#)
- [Make the switch from mainframes to Azure](#)

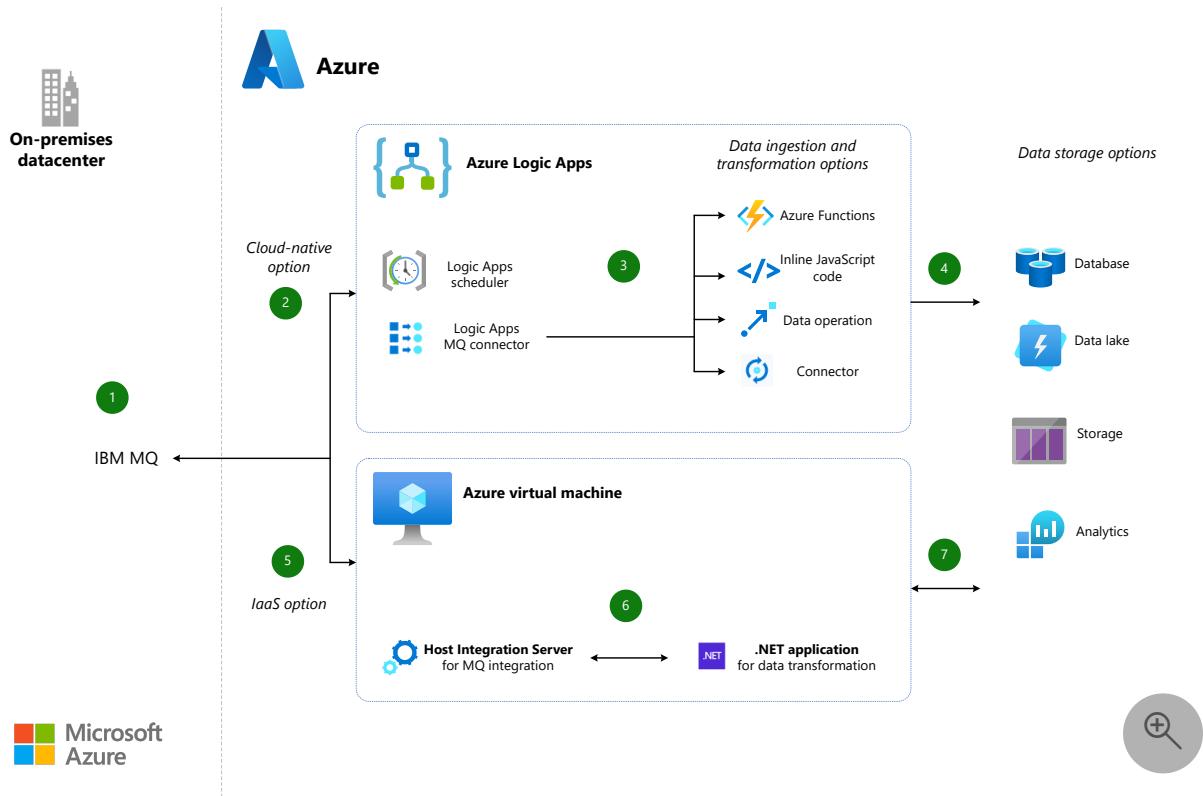
# Integrate IBM mainframe and midrange message queues with Azure

Azure Logic Apps   Azure SQL Database   Azure SQL Managed Instance

SQL Server on Azure Virtual Machines   Azure Database for PostgreSQL

When using Azure as a modern data platform, you have your choice of platform as a service (PaaS) or infrastructure as a service (IaaS). PaaS provides cloud-native options for data ingestion, transformation, and storage. IaaS gives you greater control over your hybrid infrastructure, starting with the size and type of virtual machines (VM) you choose. With either approach, you can take advantage of a variety of fully managed relational, NoSQL, and in-memory databases, storage solutions, and analytics offerings that span proprietary and open-source engines. This example architecture shows both approaches.

## Architecture



Download a [Visio file](#) of this architecture.

## Workflow

1. IBM MQ is the middleware that acts as a loosely coupled link between a mainframe or midrange system and Azure services. Messages are received and sent according to application requirements to communicate with the mainframe application layer.
2. In a cloud-native approach, Azure Logic Apps uses the MQ connector to exchange messages with IBM MQ. The Scheduler feature orchestrates the Azure workflow, sending and receiving messages at [recurring intervals](#) of one second.
3. The MQ connector can send the messages it reads directly to storage through a connector or send them to be transformed first. Logic Apps includes several options for data transformation, such as an inline [JavaScript](#) runtime that you can use to run simple JavaScript code snippets for data transformation or [data operations](#) that perform transformations on JSON, CSV, and HTML table data. You can also create serverless, single-task functions using [Azure Function](#).
4. Data is loaded into storage. Azure offers many managed data storage solutions, each providing different features and capabilities.
5. In an IaaS approach, a VM runs Microsoft Host Integration Server (HIS) with the BizTalk Adapter for WebSphere MQ. HIS exchanges messages with IBM MQ and exposes orchestration as web service to a custom .NET application.
6. A .NET application persists the data using any supported Azure data store. For example, the application can mask data or use private endpoints for security.
7. Data is loaded into storage. Azure offers many managed data storage solutions, each providing different features and capabilities.

## Components

[Azure Logic Apps](#) provides tools for data orchestration, data integration, and data transformation. It includes hundreds of [connectors](#) for accessing data on-premises or in the cloud. Make sure to test throughput and performance before choosing a data storage connector.

[Logic Apps Scheduler](#) provides triggers for starting and running workflows based on the interval and frequency of recurrence that you specify.

[Logic Apps MQ connector](#) connects your Logic Apps workflows to an IBM MQ server on-premises or on Azure. Workflows receive and send messages stored in your MQ server. A Microsoft MQ client is also included for communicating with a remote MQ

server across a TCP/IP network. You can use the client to connect to IBM WebSphere MQ 7.5, MQ 8.0, and MQ 9.0, 9.1, and 9.2.

[Host Integration Server](#) (HIS) can serve as a message integrator through the WebSphere MQ adapter in Microsoft BizTalk Server. A client and server adapter exchange messages between IBM MQ and BizTalk Server. HIS also serves as an MQ listener and can poll the MQ server for messages at intervals you specify.

[.NET](#) is a free, open-source development platform used in this example to create an app to pull the data through HIS to the data storage layer. It can also be used to access IBM WebSphere MQ Servers directly through the Microsoft Client for MQ.

## Alternatives

- For the data layer, you have your choice of managed services, including [Azure Database for PostgreSQL](#), [Azure Database for MySQL](#), [Azure Cosmos DB](#), [Azure Database for MariaDB](#), and [Azure SQL](#).
- For the storage layer, create an enterprise data lake using [Azure Data Lake Storage](#).
- For the data layer, create a big data analytics platform using [Azure Synapse Analytics](#).

## Scenario details

A popular approach in digital transformation scenarios is to see whether existing applications and middleware tiers can run as-is in a hybrid setup where Microsoft Azure serves as the scalable, distributed data platform. This example describes a data-first approach to middleware integration that enables IBM message queues (MQs) running on mainframe or midrange systems to work with Azure services so you can find the best data platform for your workload.

When using Azure as a modern data platform, you have your choice of platform as a service (PaaS) or infrastructure as a service (IaaS). PaaS provides cloud-native options for data ingestion, transformation, and storage. IaaS gives you greater control over your hybrid infrastructure, starting with the size and type of virtual machines (VM) you choose. With either approach, you can take advantage of a variety of fully managed relational, NoSQL, and in-memory databases, storage solutions, and analytics offerings that span proprietary and open-source engines.

This example architecture shows both approaches:

- **Cloud-native PaaS.** [Azure Logic Apps](#) exchanges messages with [IBM MQ](#) through the [MQ connector](#). Additional [connectors](#) provide quick access to events, data, and actions across other apps, services, systems, protocols, and platforms. Logic Apps also includes tools for transforming data from the queue if you need to modify the data format, structure, or values before storing it on Azure or sending it to the application layer.
- **VM-based IaaS.** Running [Microsoft Host Integration Server](#) (HIS) on a VM, you can use a messaging integration component that connects to IBM MQ. You control the data transformation process by creating a .NET application to read and write messages. The application can persist data in the Azure data store of your choice, and you can choose the MQ server's polling interval.

## Potential use cases

Either of these approaches can be used to:

- Enable loosely coupled applications that communicate through messaging systems to use the Azure data platform.
- Sync or replicate data incrementally between a mainframe or midrange system and Azure.
- Flow event messages between mainframe or midrange systems and Azure.

## Considerations

A hybrid datacenter configuration make sense for organizations that are developing their cloud strategies. Connecting to Azure can help bridge the gaps in your datacenter, enhance performance, improve business continuity, and expand your reach globally.

For example, applications on-premises can communicate with a modern data platform on Azure and begin taking advantage of big data analytics or machine learning. If you need a cost-effective storage solution, you can replicate mainframe data, store it on Azure, and keep the data in sync. Azure can also add the scale needed to support online transaction processing (OLTP), batch, and data ingestion systems.

## Availability

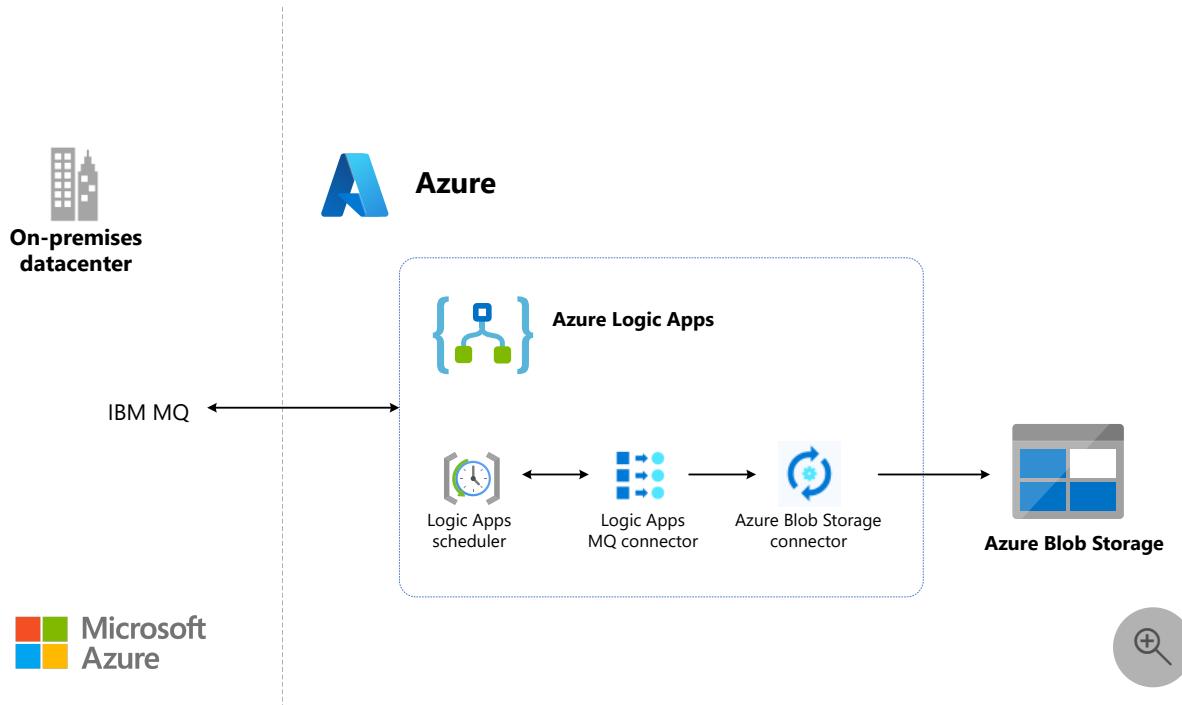
Azure service-level agreements (SLAs) describe your uptime guarantees. The SLAs for various components can vary. For example, Azure Logic Apps will be available at least 99.9 percent of the time. Configuration settings you choose can improve SLA.

# Performance efficiency

Make sure to test the throughput and performance of your data transformation layer before you finalize your architectural approach. Performance depends on several factors, including a workload's message size, latency, and the connectors that are used. Testing helps you find the most compatible target data platform.

## Storage

In this example architecture, Logic Apps connectors can be used to send messages directly to [Azure Storage](#) and [Azure Data Lake Storage](#). For example, Logic Apps includes the [Azure Blob Storage connector](#) as the following figure shows. The connector makes it easy to store massive amounts of unstructured data in [Azure Blob Storage](#). Your data becomes accessible from anywhere in the world via HTTP or HTTPS. Blob storage also supports [Azure Data Lake Storage Gen2](#), a big data analytics solution for the cloud. Data is loaded into storage using an Azure service such as the [AzCopy](#) tool, [Azure Data Factory](#), or another solution that can connect to storage.



Download a [Visio file](#) of this architecture.

Both the PaaS and IaaS architecture options support many popular managed database services. You can load data using a custom-built loader, a vendor solution, or a managed service such as [Azure Data Factory](#). Database options include:

- [Azure SQL Database](#). Part of the Azure SQL family, Azure SQL Database is the intelligent, scalable, relational database service built for the cloud. Always up to

date, it includes automated features that optimize performance, durability, and scalability, so you can focus on building new applications.

- [Azure SQL Managed Instance](#) . Part of the Azure SQL service portfolio, SQL Managed Instance combines the broadest SQL Server engine compatibility with all the benefits of a fully managed PaaS.
- [Azure SQL on Azure Virtual Machines](#) . Part of the Azure SQL family, this cost-effective option is designed for lifting and shifting SQL Server workloads to Azure. It combines the performance, security, and analytics of SQL Server with the flexibility and hybrid connectivity of Azure—with 100 percent code compatibility. Now includes SQL Server 2019 images.
- [Azure Database for PostgreSQL](#) . This fully managed relational database service is based on the community edition of the open-source PostgreSQL database engine. You can focus on application innovation instead of database management and easily scale your workloads.
- [Azure Database for MySQL](#). This fully managed relational database service is based on the community edition of the open-source MySQL database engine.
- [Azure Cosmos DB](#). A globally distributed, multi-model database, Azure Cosmos DB provides throughput and storage that scales elastically and independently across any number of geographic regions. It is a fully managed NoSQL database service that guarantees single-digit-millisecond latencies at the 99th percentile anywhere in the world.
- [Azure Synapse Analytics](#). This enterprise analytics service accelerates time to insight across data warehouses and big data systems.

## Cost optimization

This article outlines a wide range of Azure Services to demonstrate the various possibilities and you probably won't use them all for MQ integration.

- Use the [Azure pricing calculator](#) to estimate costs for the Azure resources.
- Use the [BizTalk pricing](#) to understand the pricing for the HIS solution.

## Contributors

*This article is maintained by Microsoft. It was originally written by the following contributors.*

Principal author:

- [Ashish Khandelwal](#) | Principal Engineering Architecture Manager

*To see non-public LinkedIn profiles, sign in to LinkedIn.*

## Next steps

- For more information, email [Azure Data Engineering - Mainframe & Midrange Modernization](#) (datasqlninja@microsoft.com).
- Read the [Azure Database Migration Guides](#).

## Related resources

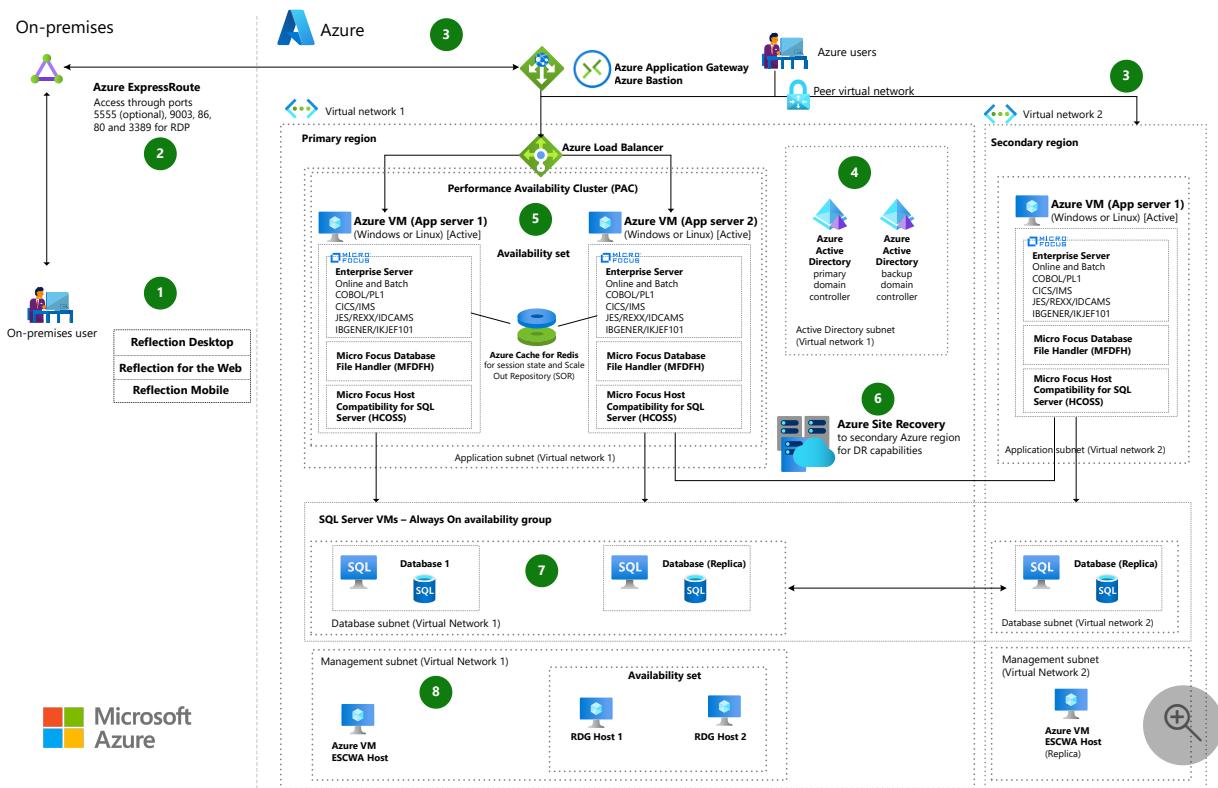
- [Azure data platform end-to-end](#)

# Micro Focus Enterprise Server on Azure VMs

Azure Site Recovery      Azure SQL Server on Virtual Machines      Azure Virtual Machines      Azure Virtual Network

This sample solution is a prescriptive, well-architected example of a Micro Focus Enterprise Server 6.0 VM-based deployment in Azure. The solution implements high availability (HA) and disaster recovery (DR) by using a secondary Azure failover region. The failover region uses Azure Site Recovery for the VMs in the application tier, and a SQL Server Always On configuration for the data tier. A Micro Focus Performance Availability Cluster (PAC) boosts VM performance, availability, and scalability.

# Architecture



*Download a [Visio file](#) of this architecture.*

## Workflow

1. Enterprise Server on-premises users interact with Enterprise Server applications through Micro Focus Reflection Desktop terminal emulator, Reflection for the Web, and Reflection Mobile. IBM 3270 terminal access can use any 3270 emulator.

Reflection Desktop is a secure, manageable, and easy to use Windows-based terminal emulator that connects users to IBM, UNIX, Linux, OpenVMS, HP 3000, and HP NonStop systems. Reflection for the Web provides Reflection features and functionality through a web interface, and Reflection Mobile provides Reflection features and functionality through a mobile interface.

2. On-premises users access the system over Azure ExpressRoute. Web-based users use ports 5555 (optional), 9003, 86, and 80. Remote desktop protocol (RDP) access uses port 3389. Port 3270 is open for 3270-based terminals and terminal emulators. For access to the Enterprise Server Common Web Administration (ESCPWA) administration tool, 3270 traffic can use any appropriately configured port.
3. A secure implementation of Enterprise Server requires a web services front-end and load balancer. This solution uses:
  - Azure Application Gateway, for complex instruction set computer (CICS) API access from the web
  - Azure Bastion, for secure access to VM management
  - Azure Load Balancer, to distribute incoming traffic among backend serversThe secondary DR Azure region also needs a web services front end to maintain secure access to the system.
4. The solution requires a Microsoft Entra implementation. Micro Focus Enterprise Server provides RACF and Top Secret identity integration using Microsoft Entra extensions.
5. A Performance and Availability Cluster (PAC) configures Enterprise Server instances in a scale-out architecture using VM [availability sets](#). In a PAC, several Enterprise Server instances work together as a single logical entity. A PAC has several advantages over a single scale-up Enterprise Server instance:
  - Distributed instances are more resistant to hardware or network issues.
  - Several instances working together perform better, maximize throughput, and provide for future horizontal scaling.
  - The instances share synchronized user and system data, using a data store called a Scale-Out Repository (SOR). The data store uses Azure Cache for Redis to improve performance and scalability.
6. For HA, Azure Site Recovery replicates a Production VM and keeps it synced in the failover region. Since the two VMs in the Production region are clones, only one needs to participate in Site Recovery.

7. Micro Focus Enterprise Server uses infrastructure-as-a-service (IaaS) SQL Server for deployments with heterogeneous distributed transactions. This solution uses a SQL Server IaaS database in an Always On cluster. With SQL Server Always On, the DR instance of the database is always online in passive, read-only mode. When failover occurs, the DR database instance becomes active.
8. As a security best practice, the solution deploys Enterprise Server management tools into a separate virtual network subnet.

## Components

This solution uses the following Azure components:

- [Azure ExpressRoute](#) extends on-premises networks into the Azure cloud over a private connection that a connectivity provider facilitates.
- [Azure Application Gateway](#) is a scalable and highly available web front end that acts as a reverse-proxy service and provides a Layer-7 internet load balancer and Web Application Firewall (WAF).
- [Azure Bastion](#) provides secure and seamless RDP and SSH access to your VMs by using SSL, without exposing public IP addresses.
- [Azure Load Balancer](#) distributes incoming network traffic across backend resources or servers according to configured load-balancing rules and health probes.
- [Azure Virtual Machines](#) offers on-demand, scalable computing resources in Azure. Azure Virtual Machines gives you the flexibility of virtualization without having to buy and maintain the physical hardware that runs it.

The Azure VMs that host Enterprise Server use [Azure Managed Disks](#) block-level storage volumes. Available Managed Disk types are Ultra disks, Premium solid-state drives (SSDs), standard SSDs, and standard hard disk drives (HDDs). This solution uses Premium SSDs or Ultra disks.

- [Azure Virtual Network](#) is the fundamental building block for private networks in Azure. An Azure Virtual Network is similar to a traditional network that you operate in your own datacenter, but adds Azure infrastructure benefits like scaling, availability, and isolation. Virtual Network lets Azure resources like VMs securely communicate with each other, the internet, and on-premises networks.

A [virtual network interface card \(NIC\)](#) lets an Azure VM communicate with internet, Azure, and on-premises resources over a virtual network. You can add NICs to an

Azure VM to give child VMs their own dedicated network interface devices and IP addresses.

- [Azure Cache for Redis](#) improves performance and scalability for applications that use backend data stores heavily. Azure Cache for Redis keeps frequently accessed data, like session state and SOR, in server memory for fast access and throughput.
- [SQL Server on Azure VMs](#) lets you use full versions of SQL Server in the cloud without having to manage on-premises hardware. Enterprise Server requires the [SQL Server IaaS Agent extension](#) for deployments that have heterogeneous distributed transactions.
- [Azure Site Recovery](#) keeps applications and workloads running during outages by replicating VMs from a primary site to a secondary location.

## Scenario details

Micro Focus Enterprise Server 6.0 is an application deployment environment for IBM z/OS mainframe applications. Enterprise Server can help you modernize and integrate your mainframe applications with technologies like .NET and Java. Enterprise Server also supports application flexibility across Linux and Windows with containerized or virtual machine (VM) deployments on Azure.

This sample solution is a prescriptive, well-architected example of a Micro Focus Enterprise Server 6.0 VM-based deployment in Azure. The solution implements high availability (HA) and disaster recovery (DR) by using a secondary Azure failover region. The failover region uses Azure Site Recovery for the VMs in the application tier, and a SQL Server Always On configuration for the data tier. A Micro Focus Performance Availability Cluster (PAC) boosts VM performance, availability, and scalability.

## Potential use cases

Deploying Enterprise Server on Azure VMs can help businesses:

- Provide a secure, stable host environment for cloud or on-premises access to mission-critical APIs.
- Lower operating and maintenance costs by supporting Linux and Windows platforms, containerized and VM-based deployments, and scale-out flexibility.
- Ensure Always On high availability and regional disaster recovery.

- Modernize applications to improve productivity and collaboration and respond to changing business needs.
- Streamline software deployment with a low-cost distributed environment, boosting developer productivity and paving the way to DevOps.

## Considerations

The following considerations, based on the [Microsoft Azure Well-Architected Framework](#), apply to this solution:

### Availability

- PACs and availability sets for Azure VMs ensure enough VMs are available to meet mission-critical batch process needs.
- SQL Server Always On Availability Groups and Azure Site Recovery provide reliability with HA and DR across geographic regions.

### Performance efficiency

- The PAC enables horizontal scaling according to application load.
- Azure Cache for Redis and Azure Storage accounts maintain critical component operations. These features provide high performance for data reads and writes, hot storage access, and long-term data storage.

### Scalability

A PAC configures several Enterprise Server instances in a scale-out architecture using [availability sets](#). The PAC supports future horizontal scaling.

### Security

All the components within the Micro Focus Enterprise Server architecture work with Azure security components like Microsoft Entra identity integration, virtual networks, and encryption as needed.

### Cost optimization

To estimate and calculate costs for your implementation of this solution, use the [Azure pricing calculator](#).

- Azure services like Application Gateway, Virtual Network, Load Balancer, and Azure Bastion are free with your Azure subscription. You pay for usage and traffic.
- Azure Site Recovery charges per protected instance.
- Most enterprises already have a Microsoft Active Directory implementation, but if you don't, Premium Microsoft Entra ID is low cost.
- For Premium SSD or Ultra managed storage disks pricing, see [Managed Disks pricing](#). Calculate VM needs based on your traffic hours, load, and storage requirements. Micro Focus Enterprise Server in Azure helps you optimize costs by turning off VMs when not in use, and scripting a schedule for known usage patterns.
- [Azure Hybrid Benefit](#) lets you use your on-premises SQL Server licenses on Azure. For more information, see the [Azure Hybrid Benefit FAQ](#).

## Contributors

*This article is maintained by Microsoft. It was originally written by the following contributors.*

Principal author:

- [Jonathon Frost](#) | Principal Software Engineer

*To see non-public LinkedIn profiles, sign in to LinkedIn.*

## Next steps

- For more information, contact [legacy2azure@microsoft.com](mailto:legacy2azure@microsoft.com).
- See the [Microsoft Azure Well-Architected Framework](#) for more information about cost optimization for VM instances.

## Related resources

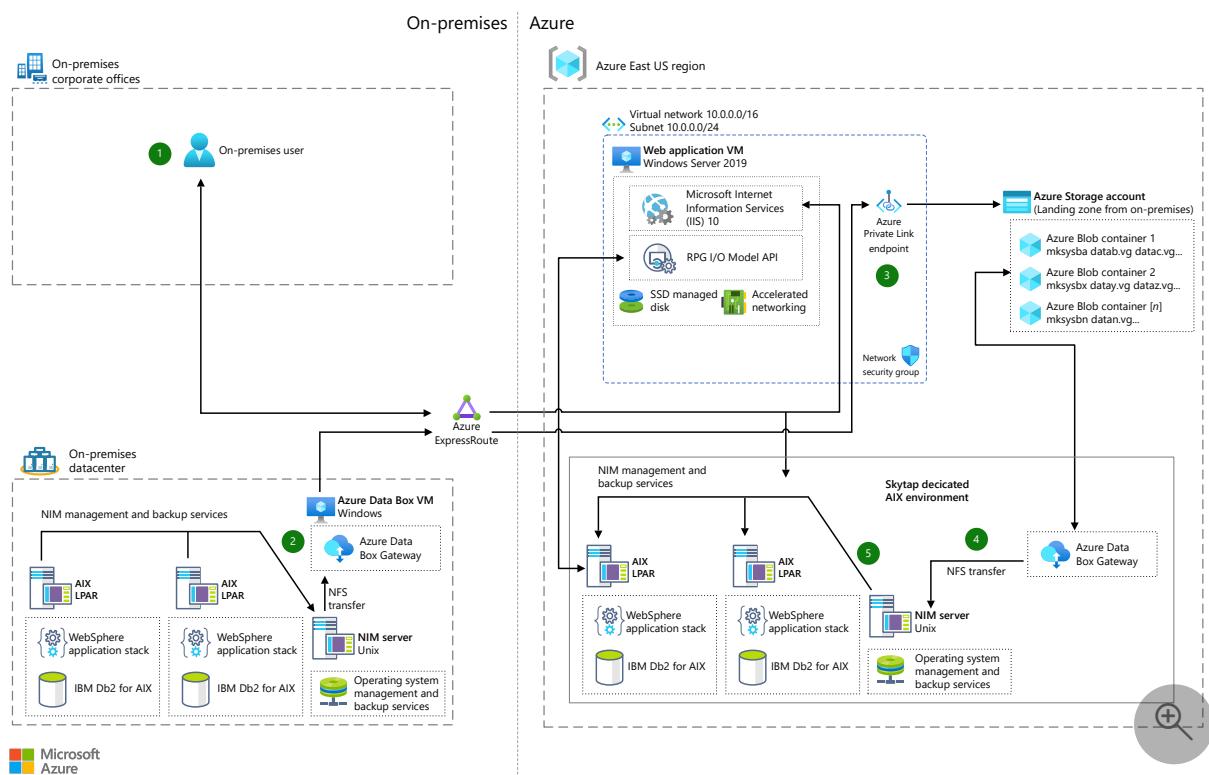
- [Refactor IBM z/OS mainframe Coupling Facility \(CF\) to Azure](#).
- [Replicate and sync mainframe data in Azure](#).
- [Multi-region n-tier application](#)
- [Multi-tier web application built for HA/DR](#)

# Migrate AIX workloads to Azure with Skytap

Azure Virtual Network   Azure Private Link   Azure ExpressRoute   Azure Virtual Machines   Azure Data Box

Skytap on Azure simplifies cloud migration for applications that run on IBM Power Systems. This example illustrates a migration of AIX logical partitions (LPARs) to Skytap on Azure and is based on best practices from recent customer experiences. A web app on Microsoft Azure gives users a modern interface for the resources running in LPARs on Skytap on Azure.

## Architecture



Download a [Visio file](#) of this architecture.

## Workflow

The numbers in the diagram correspond to the following data flow.

1. A user on-premises uses a web browser to connect to Azure through [Azure ExpressRoute](#), which creates a private connection. This web-based app provides a modern interface for the services that run on the AIX LPARs in Skytap on Azure.

2. Azure Data Box Gateway is deployed on-premises next to the datacenter's existing AIX infrastructure, which includes an AIX Network Installation Management (NIM) server. Data Box Gateway loads the data and completes the system restoration on Azure. AIX backups run using the operating system's native **mksyb** and **savevg** commands.
3. Files that are backed up to Data Box Gateway are migrated to the organization's Azure Blob Storage account through Azure Private Link, an endpoint for privately accessing Azure services.
4. In the Skytap on Azure environment, the NIM server running Unix is used to restore the base AIX operating system to the LPARs in Skytap on Azure.
5. The AIX LPAR is rebooted. Any data volume groups are restored through the Data Box Gateway via the Network File System (NFS) protocol. This process is repeated for each LPAR to be restored.

## Components

The architecture uses these components:

- [Skytap on Azure](#) is a service that runs IBM Power and x86 traditional workloads on hardware in Azure datacenters. Organizations that run applications on IBM Power-based AIX or Linux operating systems can migrate them to Azure with little upfront effort.
- [Azure Virtual Machine](#) instances provide on-demand, scalable computing power. A virtual machine (VM) gives you the flexibility of virtualization without having to buy and maintain the physical hardware that runs it.
- [Azure Virtual Network](#) is the fundamental building block for your private network in Azure. As a software defined network, a virtual network (VNet) provides an isolated environment for VMs and other Azure resources to communicate with each other, the internet, and on-premises networks. Learn more information on how Skytap on Azure connectivity works in the [Skytap Well-Architected Framework](#).
- [Azure Private Link](#) creates your own private link service in your virtual network so the web client can consume resources from Skytap on Azure.
- [Azure Blob Storage](#) is an object storage solution designed for storing massive amounts of unstructured data, such as text and binary data.

- [Azure ExpressRoute](#) extends your on-premises networks to Microsoft cloud services, including Azure and Office 365, over a private connection facilitated by a connectivity provider. Learn more information on how Azure ExpressRoute works with Skytap in the [Skytap Getting Started with Azure Networking guide](#).
- [Azure Data Box Gateway](#) is a virtual device that you install on-premises. You write data to it using the NFS and Server Message Block (SMB) protocols, and Data Box Gateway sends the data to Azure.

## Alternatives

- For access to the AIX instances running in Skytap on Azure, you can connect over a virtual private network (VPN) or the internet. For example, you can use SSH (Secure Shell) to access your AIX systems on Azure.
- To maximize security and minimize the number of open ports, you can use VMs as bastion hosts for administrative access to the LPARs. The bastion host runs within the VNet on Azure.
- To simplify user access, you can build modern front ends and apps on Azure for the AIX instances running in Skytap on Azure while continuing to run critical components or systems of record (SOR) on AIX.

## Scenario details

Since its introduction in 1986, the AIX operating system has been a top choice for large, mission-critical applications. AIX was designed for virtualization from the ground up using multiple LPARs that run in isolation on a given IBM Power System server. Until now, your choice was to rearchitect applications to move them to the cloud or bear the expense of maintaining them on-premises or in a co-located facility.

Skytap on Azure is dedicated hardware that provides a native IBM Power9 infrastructure with the AIX operating system. Full, cloud-based backup and recovery is provided with Azure Storage. You don't need to refactor or rearchitect applications to run them in Skytap on Azure, and the way you manage existing IBM Power applications on-premises changes very little.

After migration, you can start taking advantage of native Azure services to modernize applications, if desired, or continue to run systems on AIX. Either way, you immediately gain the resilience, flexibility, high availability, and scalability of Azure.

## Potential use cases

- Start an easy, self-service lift-and-shift of AIX workloads to Skytap on Azure.
- Improve business continuity with cost-effective Azure solutions for backup and disaster recovery.
- Add scale by rapidly deploying AIX LPARs on demand.
- Accelerate DevOps and increase your test coverage using on-demand resources.
- Create virtual labs using Skytap on Azure templates and environments, so you can easily demo AIX applications to customers and users.

## Considerations

### Availability

Skytap on Azure has high reliability built on IBM Power9 Systems backed by SSD RAID 6+1 storage and 10 Gb/sec backplane networking.

Skytap on Azure is supported by a service-level agreement (SLA) of 99.95 percent availability.

### Performance

Skytap on Azure provides high performance and efficiency that support demanding workloads up to 16 vCPUs and 512 GB of memory, while providing the benefits of cloud scale. With capacity on demand and pay-as-you-go pricing, you save the expense of adding hardware on premises to support higher demands. You can use smaller LPARs instead of a few large ones and configure resources as needed.

Skytap on Azure promotes operational excellence through its native support for AIX on IBM Power9 systems that are hosted within Azure datacenters and managed by Microsoft.

### Scalability

One of the advantages of an Azure-based solution is the ability to scale out. Scaling makes nearly limitless compute capacity available to an application. Azure supports multiple methods to scale out compute power, such as [virtual machine scale sets](#) and [load balancing](#) across a cluster. Other platform as a service (PaaS) options scale compute resources dynamically. In addition, applications on Azure can also use [Kubernetes clusters](#) as compute services for specified resources.

To scale up on Azure, choose a [larger VM size](#) for your workload.

## Security

Skytap on Azure meets industry cloud security requirements, including System and Organization Controls for Service Organizations 2 (SOC 2) and SOC 3 attestations and compliance with ISO 27001 and PCI DSS 3.2. To learn more about how Skytap secures your workloads, you can get more information in the [Skytap Well-Architected Framework Security Pillar](#).

## Cost optimization

Running your AIX-based workloads in Skytap on Azure helps optimize costs compared to on-premises deployments. The consumption-based usage plans let you deploy AIX LPARs only as needed and scale them on demand to meet the needs of your workloads.

See more pricing information on the [Plans + Pricing](#) tab of Skytap on Azure in Azure Marketplace.

## Deploy this scenario

To get started running AIX applications on Azure, check out the [Skytap on Azure](#) template in Azure Marketplace. Learn more about the different Migration and Deployment options with the [Skytap Well-Architected Framework](#).

## Next steps

To learn more about Skytap on Azure, contact [legacy2azure@microsoft.com](mailto:legacy2azure@microsoft.com) or check out the following resources:

- See the [Cloud Migration for Apps Running IBM Power](#) demo.
- Learn how to [accelerate your cloud strategy with Skytap on Azure](#).
- Explore the [Skytap on Azure](#) template on Azure Marketplace.
- Learn about [Skytap Migration options](#).
- [Skytap Well-Architected Framework](#)
- [Skytap documentation](#)

## Related resources

- [Mainframe file replication and sync on Azure](#)
- [Modernize mainframe & midrange data](#)

# Run HP-UX workloads in Azure with Stromasys Charon-PAR

Azure Virtual Machines

Azure Virtual Network

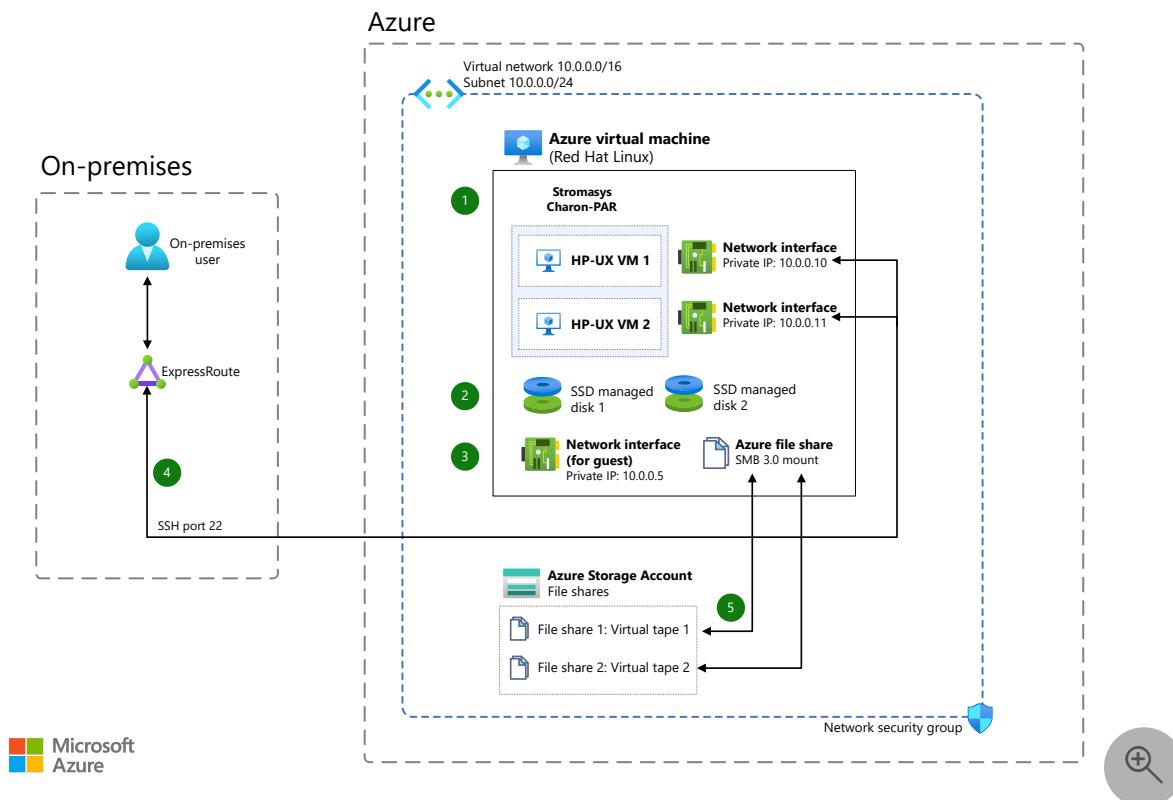
Azure ExpressRoute

Azure Storage

Azure Files

This article describes the lift-and-shift migration of an HP-UX workload to Azure. HP-UX is HP's Unix operating system for PA-RISC workstations and servers. The article shows how emulator software called Charon-PAR from Microsoft partner [Stromasys](#) can run HP-UX workloads on Azure.

## Architecture



Download a [Visio file](#) of this architecture.

## Workflow

Charon-PAR runs on Azure, emulating the PA-RISC systems for HP-UX. On this virtual system (Azure virtual machines), you install the Charon host operating system (Linux), the Charon emulator software, and your legacy operating system (HP-UX) and the associated applications. By using this configuration, you can run an HP-UX workload or

application unchanged in an emulation environment on a VM in Azure. The virtual system behaves as though you're using the original hardware.

1. The Charon-PAR software runs on Linux Azure VMs because Charon-PAR requires a Linux host. Charon-PAR emulates the PA-RISC processor architecture. The HP-UX workloads run on these emulated PA-RISC systems.
2. The HP-UX workloads can reside on the solid-state drive (SSD) managed disk of the host Azure VM.
3. One or more host network interface controllers (NICs) can be dedicated to the guest operating system. You can do that by dedicating physical NICs to the guest operating system. Each HP-UX VM gets its own Azure network interface, so it has its own dedicated private IP address. This host-specific network interface is normally used within the Charon configuration for the dedicated use of guest workloads.

Optionally, you can easily set up Azure public IP addresses on the same network interfaces. There must always be network interfaces dedicated to the guest OS. The host is allocated a network interface. PA9-32 720 allows only one network interface, but PA9-64 allows multiple network interfaces dedicated to the guest OS.

4. Users can connect via Secure Shell (SSH) directly to the HP-UX VMs (if SSH is supported by the HP-UX version). These VMs have their own dedicated network interface cards and IP addresses.
5. Azure storage account file shares that are mounted on the Linux VM allow mapping of the Charon-PAR virtual tape manager to a locally mounted device, which is backed by an Azure Files storage account in the cloud. This mapping enables low-cost storage of archived tapes for regulatory and compliance purposes.

## Components

- [Azure Virtual Machines](#) provides on-demand, scalable computing resources in Azure. An Azure VM gives you the flexibility of virtualization without requiring you to buy and maintain physical hardware. Azure VMs offer a choice of operating systems, including Windows and Linux.
- [Azure Virtual Network](#) is the fundamental building block for private networks on Azure. Virtual networks enable Azure resources like VMs to communicate with each other, the internet, and on-premises networks. Azure Virtual Network is like a traditional network in your own datacenter, but it provides the additional scale, availability, and isolation benefits of the Azure infrastructure.

- [Azure Virtual Network interface cards](#) enable an Azure VM to communicate with internet, Azure, and on-premises resources. As shown in the diagram, you can add additional network interface cards to a single Azure VM, which allows the child VMs to have their own dedicated network interface devices and IP addresses.
- [Azure SSD managed disks](#) are block-level storage volumes managed by Azure that are used with Azure VMs. Ultra disks, premium SSDs, standard SSDs, and standard hard disk drives (HDDs) are available. For this architecture, we recommend either premium SSDs or ultra disk SSDs.
- [Azure ExpressRoute](#) enables you to extend your on-premises networks into the Microsoft Cloud over a private connection that's facilitated by a connectivity provider. By using ExpressRoute, you can establish connections to Microsoft Cloud services like Azure and Microsoft 365.
- [Azure Storage](#) and [Azure Files](#) offer fully managed file shares in the cloud that can be accessed via the industry-standard Server Message Block (SMB) protocol. Azure file shares can be mounted concurrently by cloud or on-premises deployments of Windows, Linux, and macOS.
- [Stromasys Charon-PAR](#) re-creates the PA-RISC virtual hardware layer on industry standard x86-64 computer systems and VMs. The virtual hardware layer is compatible with a range of HP-UX software that runs on it, so there's no need for code conversion or source code. (See a list of [compatible versions](#).) Charon-PAR is a member of the Stromasys cross-platform hardware virtualization product family. It's a hardware virtualization layer that runs under Linux on industry standard servers. It emulates a range of historical 64-bit and 32-bit PA-RISC hardware and enables existing users of these systems to move to modern Intel-based server hardware.

## Alternatives

This solution works best with premium SSDs or ultra disk SSDs. We recommend premium SSD disks. Azure ultra SSD managed disks are an option for even higher input/output operations per second (IOPS).

For the best performance, we recommend a compute-optimized FX-series VM. You can use the Azure Fs-series for low-end spec servers, but the required minimum for PAR is 3.0 GHz. (3.4 GHz or more is recommended.) An FX-series instance is required for high-end servers.

## Scenario details

Frequently, the evolution and maintenance of business applications is stalled because of underlying legacy hardware. Sometimes the hardware is no longer compatible with newer upgrades and integrations, or, worse, it's no longer supported. Aging infrastructure for mission-critical applications is a concern. The longer the problem remains unsolved, the higher the risk and cost of mitigation.

These applications might have supported the organization's critical business and evolved over decades, gone through audits and certifications, and have well-established operations around them. Instead of a high-risk and complex re-engineering project, an alternative approach is a low-risk project that moves the applications as-is to a modern and less expensive platform, like Azure, with the help of an emulator. Such a project, often called *lift and shift*, preserves the business functionality of the application and replaces only the hardware, providing business continuity.

Running applications with an emulator on the cloud provides numerous benefits, like security, elasticity, disaster recovery, high availability, and failover. But the most significant benefits are the reduced operational costs and the ease of maintenance. No risky migration projects or changes to the operating system or middleware are required. A server virtualization on Azure can be the first step toward modernization. After the workload is on Azure, you can potentially take advantage of other benefits of the cloud.

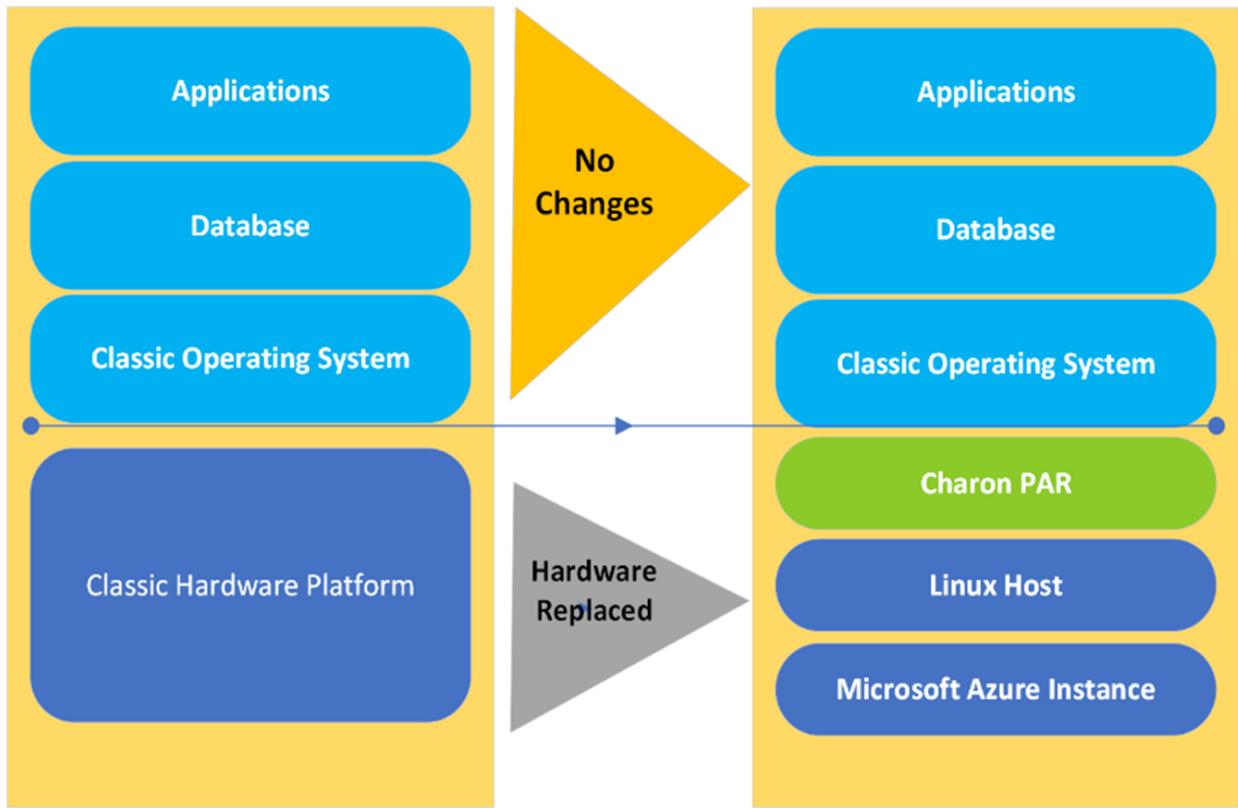
This article describes a migration of an HP-UX workload to Azure. It shows how emulator software Charon-PAR can run HP-UX workloads on Azure.

The core business of [Stromasys](#) centers on cross-platform virtualization / server virtualization software that allows owners of HP-UX legacy systems to continue running their mission-critical applications unchanged on new industry standard computer systems. Charon products preserve current application investments by enabling customers to continue to use their existing applications and business processes. Because everything continues to run without modification, no retraining or restaffing is required. Charon products dramatically reduce the cost of ownership by reducing computer footprint, energy consumption, and cooling costs while eliminating the risks and costs that are associated with running on aging hardware.

The Stromasys Charon environment provides a significantly higher level of platform stability. For the first time since the first HP-UX systems were introduced, replacing the actual physical server no longer requires changes to the HP-UX software environment. Charon also provides more platform stability and has virtually unlimited lifetime.

With the steady increase in the use of Azure-hosted systems in the typical corporate environment, an emulated HP-UX system that's hosted on Linux is the best way to host an HP-UX system in these environments.

The following image illustrates the migration approach that's recommended in this article:



Benefits of the lift-and-shift approach to migration include:

- Azure/Charon customers can continue to use existing critical applications without the cost of rewriting, porting, migrating, or retraining.
- Maintenance costs are reduced because these applications are moved to emulated systems that are hosted on Azure.

## Potential use cases

- Enable low-friction lift-and-shift to Azure of on-premises HP-UX workloads that run on PA-RISC server machines.
- Continue to use HP-UX applications that run on end-of-life PA-RISC servers without any changes, but free the applications from old hardware and continue to provide users with the same or better interfaces.
- Manage multiple server hosts and child VMs from a single interface.
- Use low-cost Azure storage to archive tapes for regulatory and compliance purposes.
- Migrate a database to the cloud and run the application in the cloud via emulation without any changes.

## Considerations

These considerations implement the pillars of the Azure Well-Architected Framework, which is a set of guiding tenets that you can use to improve the quality of a workload. For more information, see [Microsoft Azure Well-Architected Framework](#).

## Security

Security provides assurances against deliberate attacks and the abuse of your valuable data and systems. For more information, see [Overview of the security pillar](#).

- This solution uses an Azure network security group to manage traffic between Azure resources. For more information, see [Network security groups](#).
- [For increased security, consider using Azure Bastion](#). Azure Bastion maximizes admin access security by minimizing open ports. It provides secure and seamless RDP/SSH connectivity to virtual network VMs directly from the Azure portal over TLS.

## Cost optimization

Cost optimization is about reducing unnecessary expenses and improving operational efficiencies. For more information, see [Overview of the cost optimization pillar](#).

Azure avoids unnecessary costs by identifying the correct number of resource types, analyzing spending over time, and scaling to meet business needs without overspending. For example, with Azure, you pay as you go. When you don't need workloads, you can shut them down to save money. You can start Charon-PAR as a service manually or automatically when the Azure VM starts. You can stop the service manually or automatically when the host system shuts down. Ensure that you always first shut down the guest OS (HP-UX), then the emulator (Charon), and then the host VM. When you start up the system, do it in the reverse order. Here are a few other cost optimization considerations:

- [Azure Files](#) pricing depends on many factors: data volume, data redundancy, transaction volume, and the number of file sync servers that you use.
- [Azure Storage](#) costs depend on your data redundancy configurations and volume.
- The VMs in this architecture use either premium SSDs or ultra disk SSDs. For more information, see [Managed Disks pricing](#).
- For [ExpressRoute](#), you pay a monthly port fee and outbound data transfer charges.

To estimate the cost of Azure products and configurations, use the [Azure pricing calculator](#). To learn more about Stromasys products and their related services, see the

[Stromasys website](#).

## Operational excellence

Operational excellence covers the operations processes that deploy an application and keep it running in production. For more information, see [Overview of the operational excellence pillar](#).

For proactive monitoring and management, consider using [Azure Monitor](#) to monitor Azure services that host migrated HP-UX workloads.

## Performance efficiency

Performance efficiency is the ability of your workload to scale to meet the demands placed on it by users in an efficient manner. For more information, see [Performance efficiency pillar overview](#).

At least one CPU core for the host operating system and two cores per emulated CPU are required. This solution works best with [compute optimized Azure VMs](#). Compute optimized VMs have a high CPU-to-memory ratio. The [FX-series](#) virtual machine is a new addition to the F-Series. For the best performance, we recommend an FX-series VM. It's designed for high-frequency compute workloads. It features a base frequency of 3.4 GHz and an all-core-turbo clock speed of up to 4.0 GHz. We recommend FX-series for high-end HP-UX workloads.

Fx-series VMs are equipped with 2 GB of RAM and 16 GB of local SSD per CPU core.

## Contributors

*This article is maintained by Microsoft. It was originally written by the following contributors.*

Principal author:

- [Sunnyma Ghosh](#) | Senior Program Manager

Other contributors:

- [Mick Alberts](#) | Technical Writer
- [Bhaskar Bandam](#) | Senior Program Manager

*To see non-public LinkedIn profiles, sign in to LinkedIn.*

# Next steps

- [Charon-PAR ↗](#)
- [Charon on the Azure Cloud | Stromasys ↗](#)
- [What is Azure Virtual Network?](#)
- [Linux virtual machines in Azure](#)
- [What is Azure ExpressRoute?](#)
- [Create a Linux virtual machine in Azure](#)

For more information, contact [legacy2azure@microsoft.com](mailto:legacy2azure@microsoft.com).

## Related resources

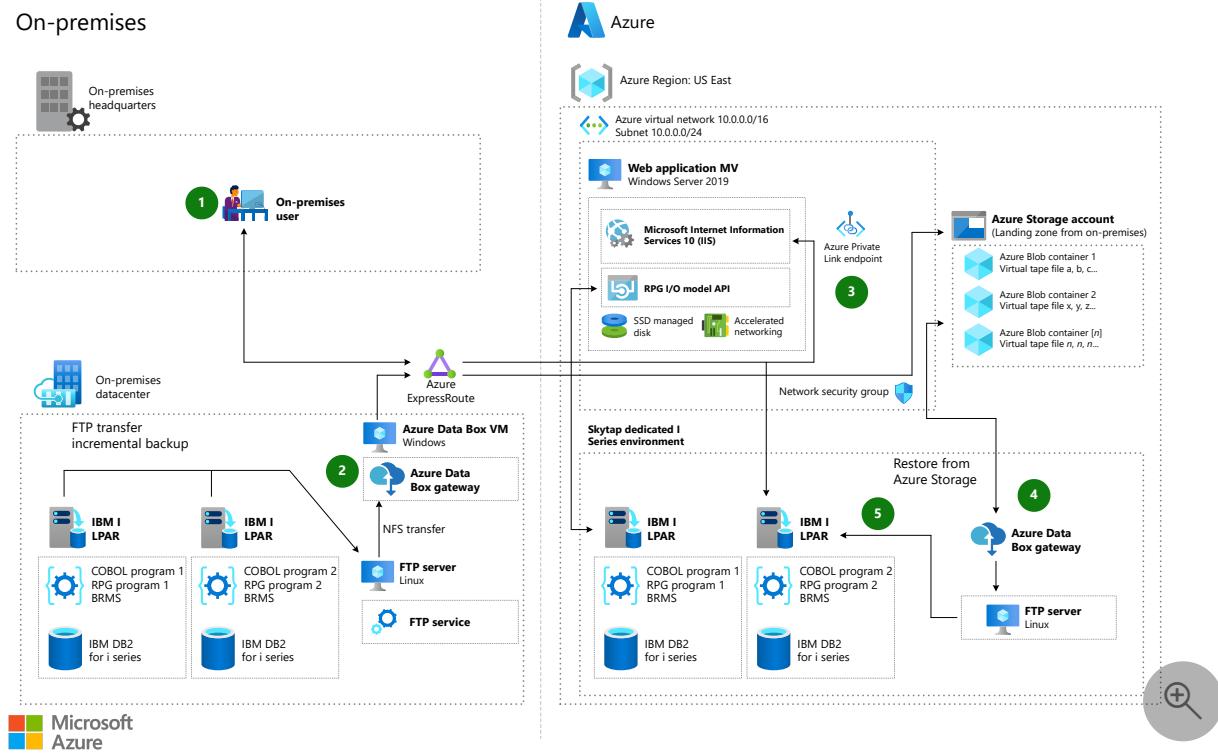
- [Mainframe migration overview](#)
- [Make the switch from mainframes to Azure](#)
- [Modernize mainframe and midrange data](#)
- [Azure mainframe and midrange architecture concepts and patterns](#)

# Migrate IBM i series to Azure with Skytap

Azure Virtual Network   Azure Private Link   Azure ExpressRoute   Azure Virtual Machines   Azure Data Box

This example architecture shows how to use the native IBM i backup and recovery services with Microsoft Azure components to quickly migrate IBM i workloads to Skytap on Azure. This native IBM Power9 infrastructure is hosted in an Azure datacenter, minimizing the latency between traditional workloads and those running natively on Azure. You get the reliability and reach of Azure, the flexibility to deploy and scale IBM i logical partitions (LPARs) on demand, plus full backup and recovery services through Azure Storage.

## Architecture



Download a [Visio file](#) of this architecture.

## Workflow

The numbers in the diagram correspond to the following data flow.

1. A user on-premises uses a web browser to connect to Azure through a private Azure ExpressRoute connection. This web-based app provides a modern interface for the services that run on the IBM i instances running in Skytap on Azure.
2. An FTP proxy and Azure Data Box Gateway are deployed on-premises next to the datacenter's existing IBM i infrastructure. Before migration, either GoSave or Backup, Recovery, and Media Services (BRMS) are used to back up the IBM i systems.
3. Data Box Gateway sends the data from the IBM i system through an Azure Private Link endpoint to an Azure Blob Storage account.
4. An FTP proxy and Data Box Gateway are deployed in the Skytap on Azure environment in the same network as the IBM i systems.
5. The IBM i systems are restored on Skytap on Azure using option 21 (restore system and user data), option 23 (restore user data), or BRMS if used for the original backup.

## Components

The architecture uses these components:

- [Skytap on Azure](#) is a service that runs IBM Power and x86 traditional workloads on hardware in Azure datacenters. Organizations of any size that run applications on IBM Power-based AIX, IBM i, or Linux operating systems can migrate them to Azure with little upfront effort.
- [Azure Virtual Machine](#) instances provide on-demand, scalable computing power. A virtual machine (VM) gives you the flexibility of virtualization without having to buy and maintain the physical hardware that runs it.
- [Azure Virtual Network](#) is the fundamental building block for your private network in Azure. As a software defined network, a virtual network (VNet) provides an isolated environment for VMs and other Azure resources to communicate with each other, the internet, and on-premises networks. Learn more information on how Skytap on Azure connectivity works in the [Skytap Well-Architected Framework](#).
- [Azure Private Link](#) creates your own private link service in your virtual network so the web client can consume resources from Skytap on Azure.
- [Azure Blob Storage](#) is an object storage solution designed for storing massive amounts of unstructured data, such as text and binary data.

- [Azure ExpressRoute](#) extends your on-premises networks to Microsoft cloud services, including Azure and Office 365, over a private connection facilitated by a connectivity provider. Learn more information on how Azure ExpressRoute works with Skytap in the [Skytap Getting Started with Azure Networking guide](#) ↗.
- [Azure Data Box Gateway](#) is a virtual device that you install on-premises. You write data to it using the Network File System (NFS) and Server Message Block (SMB) protocols, and Data Box Gateway sends the data to Azure.

## Alternatives

- You can connect to IBM i instances running in Skytap on Azure across a virtual private network (VPN) or the internet. For example, you can use SSH (Secure Shell) to access your IBM i applications on Azure.
- To maximize security and minimize the number of open ports, you can use VMs as bastion hosts for administrative access to the LPARs. The bastion host runs within the VNet on Azure. For example, administrators can use a 5250 terminal emulator to access their IBM i systems.
- You can use BRMS to back up your system before migration, and then use BRMS restore for incremental backups.
- In a high availability scenario, you can replicate journal data over your organization's ExpressRoute or VPN connection in near real-time. In case of a failure, you can perform a role swap for a near immediate failover.

## Scenario details

The IBM System i family of midrange computers was first introduced in 1988 as the AS/400. Until now, your choice was to rearchitect iSeries applications before moving them to the cloud or maintain them on-premises or in a co-located facility—both expensive options.

In this example, a web app on Azure gives users a modern interface for the resources running in Skytap on Azure. You can continue to run critical components or systems of record (SOR) on IBM i on-premises. You can also migrate complete IBM i workloads and modernize them using native Azure services, such as advanced analytics and machine learning. In this type of all-cloud scenario, Skytap on Azure helps you optimize costs.

## Potential use cases

- Enable easy, self-service lift-and-shift of on-premises workloads running IBM i to Azure.
- Modernize applications using native Azure services in a hybrid configuration that connects to earlier systems and data running on IBM i.
- Improve business continuity with cost-effective Azure solutions for backup and disaster recovery.
- Add scale by rapidly deploying IBM i instances on demand.

## Considerations

### Availability

Skytap on Azure has high reliability built on IBM Power9 Systems backed by SSD RAID 6+1 storage and 10 Gb/sec backplane networking.

### Performance

Skytap on Azure provides high performance and efficiency that support demanding workloads up to 44,000 CPWs and 512 GB of RAM, while providing the benefits of cloud scale. With capacity on demand and pay-as-you-go pricing, you save the expense of adding hardware on premises to meet changing demands. You can use smaller LPARs instead of a few large ones and configure resources as needed.

### Scalability

One of the advantages of an Azure-based solution is the ability to scale out. Scaling makes nearly limitless compute capacity available to an application. Azure supports multiple methods to scale out compute power, such as [virtual machine scale sets](#) and [load balancing](#) across a cluster. Other services scale compute resources dynamically. In addition, applications on Azure can also use [Kubernetes clusters](#) as compute services for specified resources.

Azure compute scale-up can be as simple as choosing the right [virtual machine](#) for your workload.

### Security

Skytap on Azure meets industry cloud security requirements, including System and Organization Controls for Service Organizations 2 (SOC 2) and SOC 3 attestations and compliance with ISO 27001 and PCI DSS 3.2. To learn more about how Skytap secures your workloads, you can get more information in the [Skytap Well-Architected Framework Security Pillar](#).

## Cost optimization

Running IBM i series workloads in Skytap on Azure helps optimize costs compared to on-premises deployments. The consumption-based usage plans let you deploy LPARs only as needed and scale them on demand to meet the needs of your workloads.

See more pricing information on the [Plans + Pricing](#) tab of Skytap on Azure in Azure Marketplace.

## Deploy this scenario

To get started running iSeries applications on Azure, check out the [Skytap on Azure](#) template in Azure Marketplace. Learn more about the different Migration and Deployment options with the [Skytap Well-Architected Framework](#).

## Next steps

To learn more about Skytap on Azure, contact [legacy2azure@microsoft.com](mailto:legacy2azure@microsoft.com) or check out the following resources:

- See the [Cloud Migration for Apps Running IBM Power](#) demo.
- Learn how to [accelerate your cloud strategy with Skytap on Azure](#).
- Explore the [Skytap on Azure](#) template on Azure Marketplace.
- Learn about [Skytap Migration options](#).
- [Skytap Well-Architected Framework](#)
- [Skytap documentation](#)

## Related resources

- [Mainframe file replication and sync on Azure](#)
- [Modernize mainframe & midrange data](#)

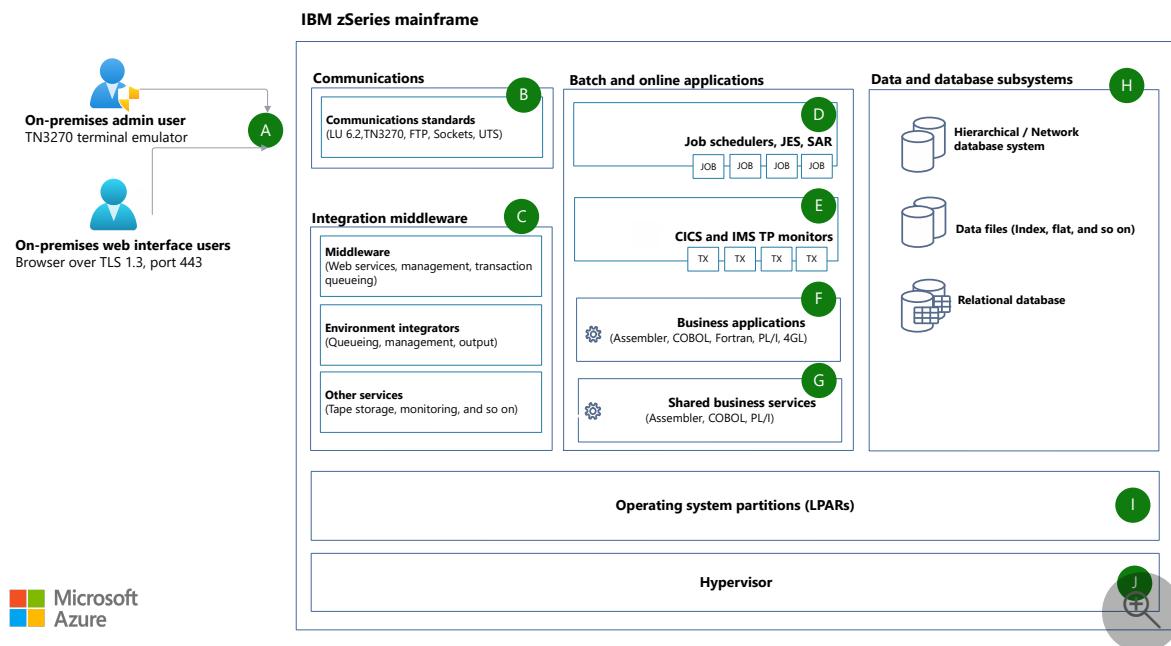
# Refactor mainframe architecture by using CloudFrame Renovate

Azure Virtual Machines    Azure Kubernetes Service (AKS)    Azure Virtual Network    Azure SQL Database

Azure Site Recovery

CloudFrame Renovate migrates COBOL code to Java Spring Boot Batch quickly, without compromising quality, precision, functional equivalency, or performance. Renovate is a DIY tool that uses guided actions and automation to help make code migration easy. Just provide the inputs and download Maven or Gradle Java projects. No specialized skills or staff are required.

## Legacy IBM zSeries architecture



Download a [Visio file](#) of the architectures in this article.

## Workflow

- Data is input over TCP/IP, including TN3270 and HTTP(S).
- Data is input into the mainframe via standard mainframe protocols.
- Middleware and utility services manage services like tape storage, queueing, output, and web services within the environment.

D. The batch application execution environment includes scheduling, workload management, and SPOOL operations.

E. Online transaction processing environments provide high availability, workload management, and XA-compliant transaction management.

F. Business applications written in COBOL, PL/I, or Assembler (or compatible languages) run in environments enabled for batch and online.

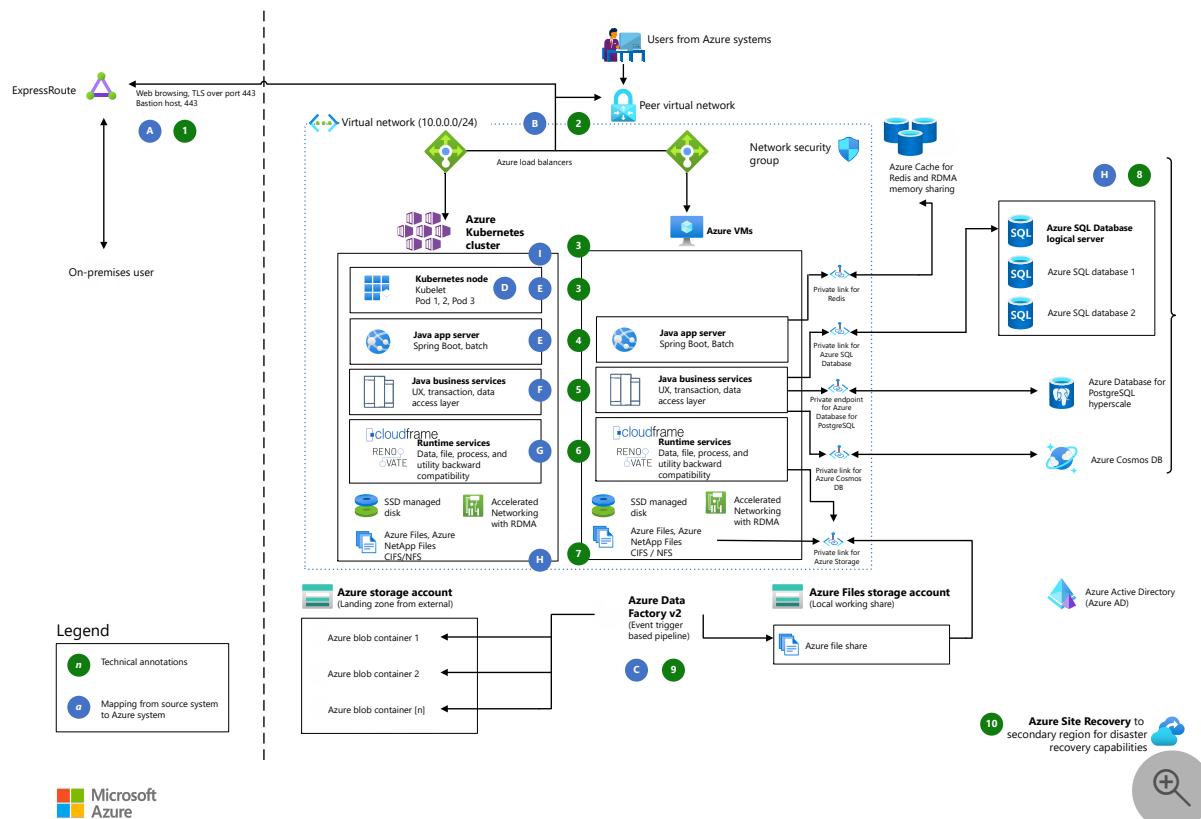
G. Shared business services standardize solutions for shared services like logging, error handling, I/O, and pre-SOA business services.

H. Data is stored in data and database services like hierarchical, network, and relational database subsystems and indexed and sequential data files.

I. Operating system partitions (virtual machines) provide the interface between the engine and the software.

J. The Processor Resource / System Manager (PR/SM) hypervisor performs direct hardware virtualization to partition physical machines into virtual machines (VMs).

## Migrated Azure architecture



Download a [Visio file](#) of the architectures in this article.

# Workflow

1. Data is typically input either via Azure ExpressRoute from remote clients or from other applications currently running Azure. In either case, TCP/IP is the primary means of connection to the system. TLS port 443 provides user access to web-based applications. You can use the web application presentation layer virtually unchanged to minimize the need for training. Or you can update the web application presentation layer with modern UX frameworks as needed. You can use Azure VM bastion hosts to provide admin access to the VMs. Doing so improves security by minimizing open ports.
2. In Azure, Azure load balancers manage access to the application compute clusters to provide high availability. This approach enables scale-out of compute resources to process the input work. Layer 7 (application layer) and Layer 4 (transport layer) load balancers are available. The type used depends on the application architecture and API payloads at the entry point of the compute cluster.
3. You can deploy to a VM in a compute cluster or in a pod that can be deployed in a Kubernetes cluster. Java Business Services and applications created via Renovate run equally well on Azure VMs and Azure Kubernetes containers. For a more detailed analysis of compute options, see [this Azure compute service decision tree](#).
4. Application servers receive the input in the compute clusters and share application state and data by using Azure Cache for Redis or Remote Direct Memory Access (RDMA).
5. Business services and applications in the application clusters allow for multiple connections to persistent data sources. These data sources can include PaaS services like Azure SQL Database and Azure Cosmos DB, databases on VMs, such as Oracle or Db2, and big data repositories like Azure Databricks and Azure Data Lake. Application data services can also connect to streaming data services like Kafka and Azure Stream Analytics.
6. Renovate runtime services provide backward compatibility with mainframe data architectures and emulation of mainframe QSAM and VSAM file systems, decoupling data migration to UTF-8 from refactoring to Java and rehosting in Azure. Additional runtime services include compatibility with SORT, IDCAMS, IE utilities, GDG retention management, and more.
7. Data services use a combination of high-performance storage (Ultra SSD / Premium SSD), file storage (Azure NetApp Files / Azure Files) and standard storage (blob, archive, backup) that can be either locally redundant or geo-redundant, depending on the use.

8. Azure platform as a service (PaaS) data services provide scalable, highly available geo-redundant data storage that's shared across compute resources in a cluster.
9. Azure Data Factory enables data ingestion and synchronization with multiple data sources both within Azure and from external sources. Azure Blob Storage is a common landing zone for external data sources.
10. Azure Site Recovery provides disaster recovery of the VM and container cluster components.

## Components

- [Azure Virtual Machines](#) is one of several types of on-demand, scalable computing resources that Azure provides. An Azure VM gives you the flexibility of virtualization, and you don't have to buy and maintain the physical hardware that runs it.
- [Azure Kubernetes Service \(AKS\)](#) can help you start developing and deploying cloud-native apps, with built-in code-to-cloud pipelines and guardrails.
- [Azure SSD managed disks](#) are block-level storage volumes that are managed by Azure and used with Azure VMs. The available types of disks are Ultra Disk, Premium SSD, Standard SSD, and Standard HDD. For this architecture, we recommend either Premium SSDs or Ultra Disk SSDs.
- [Azure Virtual Network](#) is the fundamental building block for your private network on Azure. Virtual Network enables many types of Azure resources, like Azure VMs, to communicate with each other, the internet, and on-premises networks, all with enhanced security. Virtual Network is like a traditional network that you'd operate in your own datacenter, but it provides additional benefits like scale, availability, and isolation.
- [Azure SQL Database](#) is a fully managed PaaS database engine that handles most database management functions, like upgrading, patching, backups, and monitoring, without your involvement. SQL Database always runs on the latest stable version of the SQL Server database engine and a patched OS.
- [Azure Cache for Redis](#) is a distributed, managed cache that helps you build highly scalable and responsive applications by providing fast access to your data.
- [Data Factory](#) is a cloud-based data integration service that orchestrates and automates the movement and transformation of data.
- [Azure Site Recovery](#) contributes to your business continuity and disaster recovery (BCDR) strategy by orchestrating and automating replication of Azure VMs between regions, on-premises VMs, and physical servers to Azure, and by replicating on-premises machines to a secondary datacenter.

# Scenario details

Using existing mainframe data and processes reduces risk and accelerates time to value. CloudFrame Renovate provides backward compatibility with mainframe data architectures and support for mainframe utilities like SORT. You can stage binary snapshots of VSAM and QSAM data in CloudFrame's emulated file systems, backed by Azure services like Blob Storage, Azure Cosmos DB, disk storage, and Azure SQL.

Refactoring mainframe applications by using Renovate moves application and infrastructure transformation from proprietary legacy solutions into standardized, benchmarked, open technologies. This transformation also moves teams toward Agile DevOps operating models.

Renovate-generated Java code is easy to understand, is rated A by SonarQube, and produces results that are functionally equivalent and data equivalent. The resulting code can be maintained by your current developers, using your DevOps processes and toolchains. Developers don't need knowledge about mainframes or COBOL to maintain the refactored application. The resulting code is highly maintainable, and the transformation risk is low.

By using Renovate's incremental modernization approach, you, and not the tool or tool vendor, can determine the granularity and speed of change. Refactoring with Renovate is a fast, low-risk way to move COBOL workloads to cloud-native Java on Azure.

## Potential use cases

Refactoring to Azure by using Renovate can help organizations and teams that want these benefits:

- More control of the modernization processes through the use of DIY tools.
- An incremental approach to modernization.
- Automated refactoring tools that can be configured according to custom requirements.
- Migration of mainframe workloads to the cloud without the consequential side effects of a complete rewrite.
- A modern infrastructure without the cost structures, limitations, and rigidity of mainframes.
- Migration of core applications while maintaining continuity with other on-premises applications.
- Solutions that offer various options for disaster recovery.
- The horizontal and vertical scalability that Azure provides.

# Considerations

These considerations implement the pillars of the Azure Well-Architected Framework, which is a set of guiding tenets that you can use to improve the quality of a workload. For more information, see [Microsoft Azure Well-Architected Framework](#).

## Reliability

Reliability ensures that your application can meet the commitments you make to your customers. For more information, see [Overview of the reliability pillar](#).

High availability and performance are built into this solution because of the load balancers and compute autoscaling. If one presentation, transaction, or batch server fails, the other server behind the load balancer handles the workload. The architecture uses [Site Recovery](#) to mirror Azure VMs. It uses PaaS storage and database services for replication to a secondary Azure region for quick failover and disaster recovery if an Azure datacenter fails. Finally, you can fully automate the deployment and operational architecture.

## Security

Security provides assurances against deliberate attacks and the abuse of your valuable data and systems. For more information, see [Overview of the security pillar](#).

Security in Azure is achieved through a layered approach of policy, process, automated governance and incident reporting, training, network vulnerability analysis, penetration testing, encryption, and DevSecOps operating models. Services like Microsoft Entra ID, Azure Virtual Network, Azure Private Link, and network security groups are fundamental to achieving this enhanced security.

## Cost optimization

Cost optimization is about reducing unnecessary expenses and improving operational efficiencies. For more information, see [Overview of the cost optimization pillar](#).

Azure provides cost optimization by running VMs and Kubernetes pods on commodity hardware, scripting a schedule to turn off VMs that aren't in use, and using Kubernetes pods to increase deployment density. Reserved and spot instances can further reduce costs. Microsoft Cost Management provides cost transparency by giving you a single, unified view of costs versus budgets. [Azure Reservations](#) and [Azure savings plan for compute](#) generate significant discounts off of pay-as-you-go pricing. You can use

these offerings separately or together to compound the savings. Use the [Azure pricing calculator](#) to estimate the cost of implementing the solution.

## Contributors

*This article is maintained by Microsoft. It was originally written by the following contributors.*

Principal author:

- Jim Dugan | Principal TPM

Other contributors:

- [Mick Alberts](#) | Technical Writer
- [Bhaskar Bandam](#) | Senior TPM

*To see non-public LinkedIn profiles, sign in to LinkedIn.*

## Next steps

- For more information about this architecture, contact [legacy2azure@microsoft.com](mailto:legacy2azure@microsoft.com).
- For more information about Renovate, see the [CloudFrame](#) website.
- For more information about the components of this architecture, see these articles:
  - [Virtual machines in Azure](#)
  - [Azure Kubernetes Service](#)
  - [What is Azure Virtual Network?](#)
  - [What is Azure SQL Database?](#)
  - [About Site Recovery](#)

## Related resources

- [Azure mainframe and midrange architecture design](#)
- [Mainframe migration overview](#)
- [Make the switch from mainframes to Azure](#)
- [Mainframe access to Azure databases](#)

# Refactor IBM z/OS mainframe coupling facility (CF) to Azure

Azure Kubernetes Service (AKS)

Azure Virtual Machines

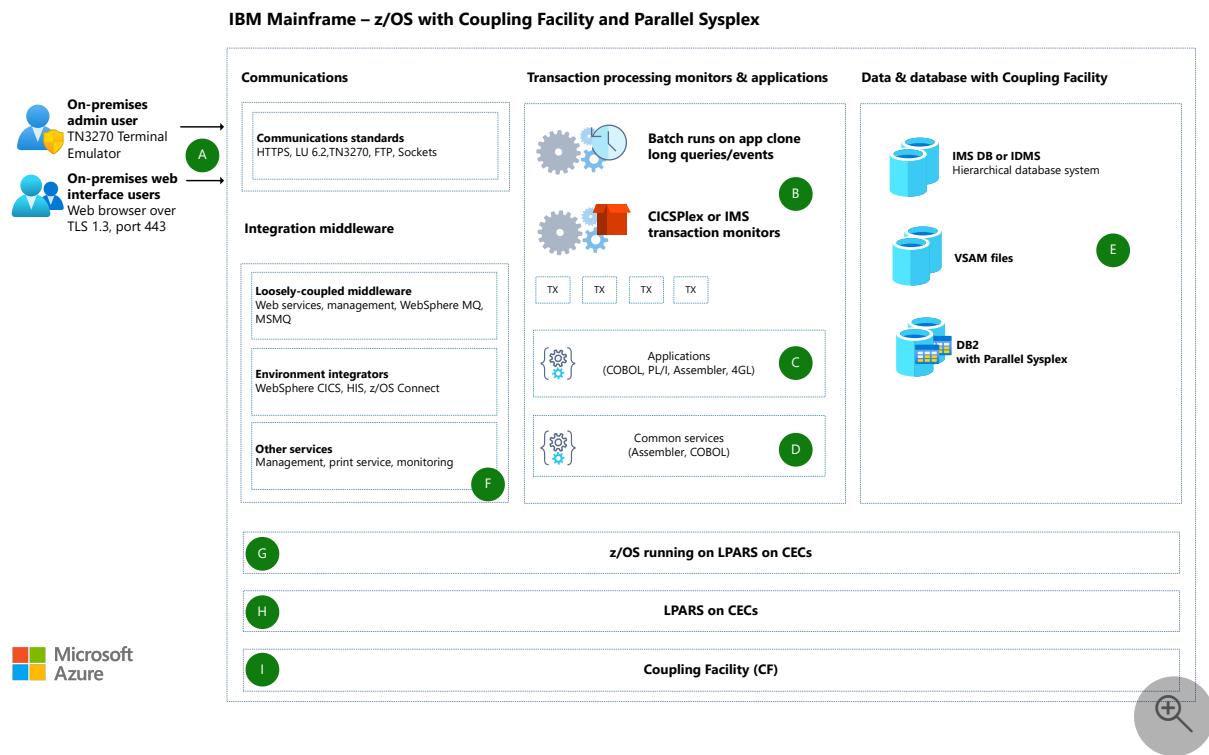
Azure Virtual Network

This architecture shows how Azure can provide scale-out performance and high availability that is similar to IBM z/OS mainframe systems with Coupling Facilities (CFs).

## Architecture

### Mainframe architecture

The following diagram shows the architecture of an IBM z/OS mainframe system with Coupling Facility and Parallel Sysplex:



Download a [Visio file](#) of this architecture.

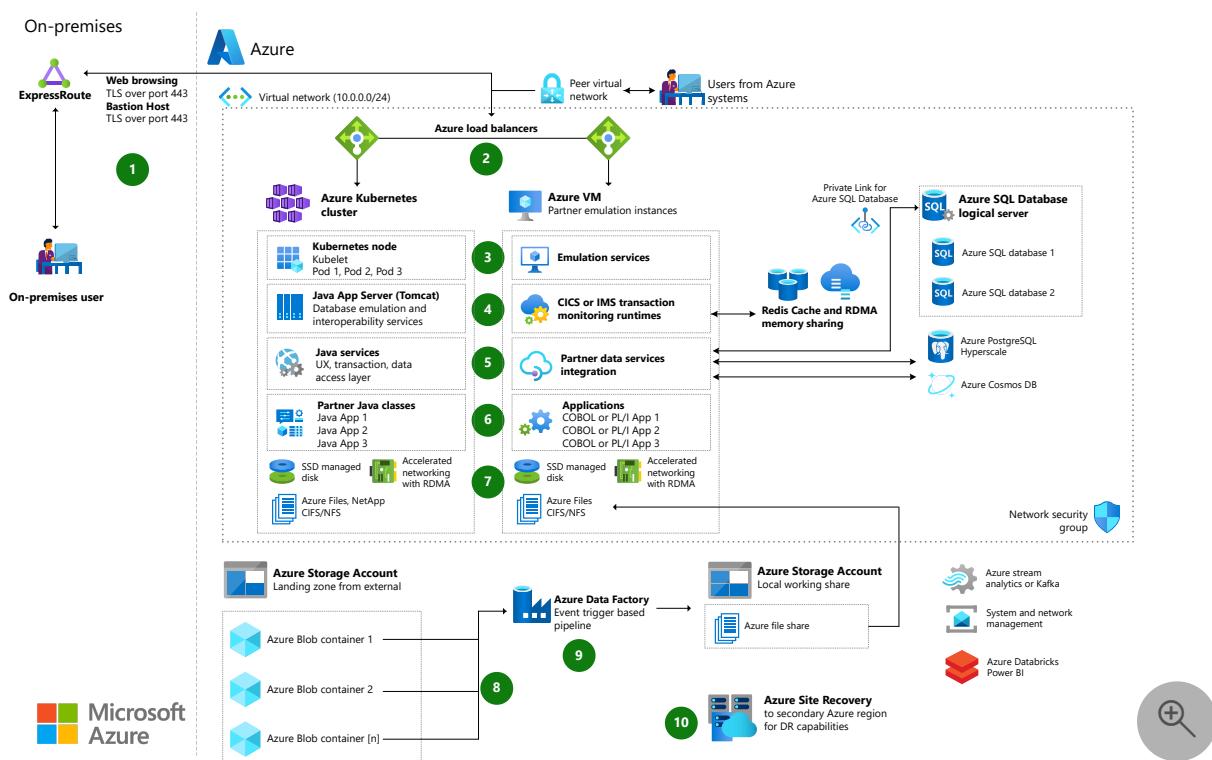
## Workflow

- Input travels into the mainframe over TCP/IP, using standard mainframe protocols like TN3270 and HTTPS (A).

- Receiving applications can be either batch or online systems (**B**). Batch jobs can spread or clone across multiple CECs that share data in the data tier. The online tier can spread a logical CICS region across multiple CECs by using Parallel Sysplex CICS or CICSplex.
- COBOL, PL/I, Assembler, or compatible applications (**C**) run in the Parallel Sysplex-enabled environment, such as a CICSplex.
- Other application services (**D**) can also use shared memory across a CF.
- Parallel Sysplex-enabled data services like Db2 (**E**) allow for scale-out data storage in a shared environment.
- Middleware and utility services like MQSeries, management, and printing (**F**) run on z/OS in each CEC.
- Logical partitions (LPARs) on each CEC (**G**) run z/OS. Other operating environments like z/VM or other engines like zIIP or IFL might also exist.
- A CEC connects via the CF (**H**) to shared memory and state.
- The CF (**I**) is a physical device that connects multiple CECs to share memory.

## Azure architecture

The next diagram shows how Azure services can provide similar functionality and performance to z/OS mainframes with Parallel Sysplex and CFs:



Download a [Visio file](#) of this architecture.

## Workflow

1. Input comes from remote clients via Express Route, or from other Azure applications. In either case, TCP/IP is the primary connection to the system.

A web browser to access Azure system resources replaces terminal emulation for demand and online users. Users access web-based applications over TLS port 443. Web applications' presentation layers can remain virtually unchanged, to minimize end user retraining. Or you can update the web application presentation layer with modern user experience frameworks.

For admin access to the Azure Virtual Machines (VMs), [Azure Bastion](#) hosts maximize security by minimizing open ports.

2. In Azure, access to the application compute clusters is through [Azure Load Balancer](#), allowing for scale-out compute resources to process the input work.
3. The type of application compute cluster to use depends on whether the application runs on VMs or in a container cluster like Kubernetes. Usually, mainframe system emulation for applications written in PL/I or COBOL use VMs, while applications refactored to Java or .NET use containers. Some mainframe system emulation software can also support deployment in containers.
4. Application servers, such as Tomcat for Java or CICS/IMS transaction processing monitor for COBOL, receive the input, and share application state and data using Azure Cache for Redis or Remote Direct Memory Access (RDMA). This capability is similar to CF for mainframes.
5. Data services in the application clusters allow for multiple connections to persistent data sources. These data sources can include platform-as-a-service (PaaS) data services like [Azure SQL Database](#) and Azure Cosmos DB, databases on VMs like Oracle or Db2, or Big Data repositories like Databricks and Azure Data Lake. Application data services can also connect to streaming data analytics services like Kafka and Azure Stream Analytics.

Azure PaaS data services provide scalable and highly available data storage that multiple compute resources in a cluster can share. These services can also be geo-redundant.

6. The application servers host various application programs based on language, such as Java classes in Tomcat, or COBOL programs with CICS verbs in CICS emulation VMs.
7. Data services use a combination of high-performance storage on Ultra or Premium solid-state disks (SSDs), file storage on Azure NetApp Files or [Azure Files](#), and

standard blob, archive, and backup storage that can be locally redundant or geo-redundant.

8. Azure Blob storage is a common landing zone for external data sources.
9. Azure Data Factory ingests and synchronizes data from multiple internal and external data sources.
10. Azure Site Recovery provides DR for the VM and container cluster components.

## Components

- [Azure ExpressRoute](#) extends your on-premises networks into the Microsoft cloud over a private connection that a connectivity partner provides. With ExpressRoute, you can establish connections to cloud services like Azure and Office 365.
- [Azure Bastion](#) is a fully managed PaaS that you provision inside your virtual network. Bastion provides secure and seamless RDP and SSH connectivity to the VMs in your virtual network directly from the Azure portal over TLS.
- [Azure Load Balancer](#) distributes inbound flows from the load balancer's front end to back-end pool instances, according to configured load-balancing rules and health probes. The back-end pool instances can be Azure VMs or instances in a virtual machine scale set. Load Balancer is the single point of contact for clients.

Load Balancer operates at layer 4 of the Open Systems Interconnection (OSI) model. Both level 7 application level and level 4 network protocol level load balancers are available. The type to use depends on how the application input reaches the compute cluster's entry point.

- [Azure Virtual Machines](#) provides on-demand, scalable computing resources that give you the flexibility of virtualization. Azure VMs give you a choice of operating systems, including Windows and Linux.

Most Azure high-performance computing (HPC) VM sizes feature a network interface for [RDMA connectivity](#).

- [Azure Virtual Networks](#) are the fundamental building blocks for Azure private networks. Virtual networks let Azure resources like VMs securely communicate with each other, the internet, and on-premises networks. An Azure Virtual Network is similar to a traditional on-premises network, but with the benefits of Azure infrastructure scalability, availability, and isolation.

Virtual network interfaces let Azure VMs communicate with internet, Azure, and on-premises resources. As in this architecture, you can add several network interface cards to one Azure VM, so child VMs can have their own dedicated network interface devices and IP addresses.

- [Azure Kubernetes Service \(AKS\)](#) is a fully managed Kubernetes service for deploying and managing containerized applications in container-based compute clusters.
- [Azure Cache for Redis](#) is a fully managed, in-memory cache that improves the performance and scalability of data-intensive architectures. The current architecture uses Azure Cache for Redis to share data and state between compute resources.
- [Azure SQL Database](#) is a fully managed PaaS database engine that always runs the latest stable version of SQL Server and patched OS, with 99.99% availability. SQL Database handles upgrading, patching, backups, monitoring, and most other database management functions without user involvement. These PaaS capabilities let you focus on business critical, domain-specific database administration and optimization.
- [Azure Private Link](#) for Azure SQL Database provides a private, direct connection from Azure VMs to Azure SQL Database that is isolated to the Azure networking backbone.
- [Azure Cosmos DB](#) is an Azure PaaS service for NoSQL databases.
- [Azure Database for PostgreSQL](#) is an Azure PaaS service for PostgreSQL databases.
- [Azure managed disks](#) are block-level storage volumes that Azure manages on Azure VMs. The available types of disks are ultra disks, premium SSDs, standard SSDs, and standard hard disk drives (HDDs). This architecture works best with Premium SSDs or Ultra Disk SSDs.
- [Azure Data Factory](#) is a fully managed, serverless data integration solution for ingesting, preparing, and transforming data at scale.
- [Azure Files](#) offers fully managed file shares in an Azure Storage Account that are accessible from the cloud or on-premises. Windows, Linux, and macOS deployments can mount Azure file shares concurrently, and access files via the industry-standard Server Message Block (SMB) protocol.
- [Azure Stream Analytics](#) is an Azure-based analytics service for streaming data.

- [Azure Databricks](#) is an Apache Spark PaaS service for Big Data analytics.

## Scenario details

Coupling Facilities (CFs) are physical devices that connect multiple mainframe servers or Central Electronics Complexes (CECs) with shared memory, letting systems scale out to increase performance. Applications written in languages like COBOL and PL/I seamlessly take advantage of these tightly coupled scale-out features.

IBM Db2 databases and Customer Information Control System (CICS) servers can use CFs with a mainframe subsystem called Parallel Sysplex that combines data sharing and parallel computing. Parallel Sysplex allows a cluster of up to 32 systems to share workloads for high performance, high availability, and disaster recovery (DR). Mainframe CFs with Parallel Sysplex typically reside in the same datacenter, with close proximity between the CECs, but can also extend across datacenters.

Azure resources can provide similar scale-out performance with shared data and high availability. Azure compute clusters share memory through data caching mechanisms like Azure Cache for Redis, and use scalable data technologies like Azure SQL Database and Azure Cosmos DB. Azure can implement availability sets and groups, combined with geo-redundant capabilities, to extend scale-out compute and high availability to distributed Azure datacenters.

## Considerations

These considerations implement the pillars of the Azure Well-Architected Framework, which is a set of guiding tenets that can be used to improve the quality of a workload. For more information, see [Microsoft Azure Well-Architected Framework](#).

## Availability

This architecture uses [Azure Site Recovery](#) to mirror Azure VMs to a secondary Azure region for quick failover and DR if an Azure datacenter fails.

## Resiliency

Resiliency is built into this solution because of the Load Balancers. If one presentation or transaction server fails, other servers behind the Load Balancer can run the workloads.

## Scalability

You can scale out the server sets to provide more throughput. For more information, see [Virtual machine scale sets](#).

## Security

Security provides assurances against deliberate attacks and the abuse of your valuable data and systems. For more information, see [Overview of the security pillar](#).

- This solution uses an Azure network security group (NSG) to manage traffic between Azure resources. For more information, see [Network security groups](#).
- [Private Link for Azure SQL Database](#) provides a private, direct connection isolated to the Azure networking backbone from the Azure VMs to Azure SQL Database.
- [Azure Bastion](#) maximizes admin access security by minimizing open ports. Bastion provides secure and seamless RDP/SSH connectivity to virtual network VMs directly from the Azure portal over TLS.

## Cost optimization

Cost optimization is about looking at ways to reduce unnecessary expenses and improve operational efficiencies. For more information, see [Overview of the cost optimization pillar](#).

- Azure SQL Database should use [Hyperscale or Business Critical](#) SQL Database tiers for high input/output operations per second (IOPS) and high uptime SLA.
- This architecture works best with Premium SSDs or Ultra Disk SSDs. For more information, see [Managed Disks pricing](#).

## Next steps

- For more information, please contact [legacy2azure@microsoft.com](mailto:legacy2azure@microsoft.com).
- [Azure ExpressRoute](#)
- [Azure Bastion](#)
- [Azure Load Balancer](#)
- [Azure managed disks](#)
- [Azure Virtual Networks](#)
- [Create, change, or delete a network interface](#)

## Related resources

- Azure mainframe and midrange architecture concepts and patterns
- Mainframe migration overview
- Mainframe rehosting on Azure virtual machines
- IBM z/OS online transaction processing on Azure
- Integrate IBM mainframe and midrange message queues with Azure

# Refactor IBM z/TPF mainframe systems to Azure

Azure Kubernetes Service (AKS)

Azure Cosmos DB

Azure Virtual Network

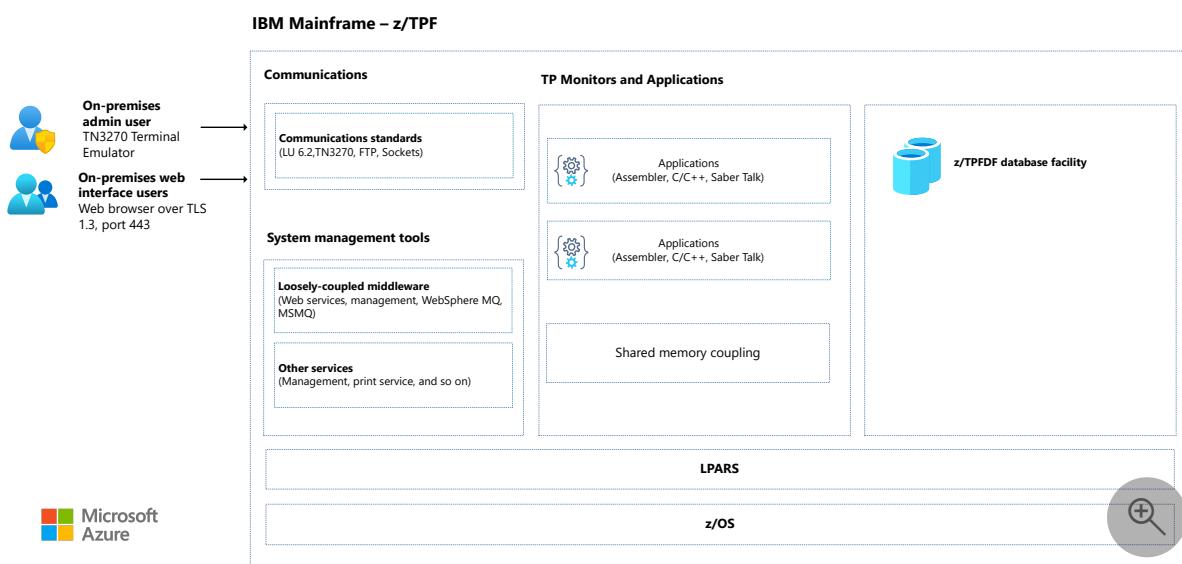
Azure Cache for Redis

Organizations migrating from z/TPF mainframes to the cloud need a robust platform that can process high volume transactions. This solution delivers cloud-enabled applications and databases that are functionally equivalent to the z/TPF legacy counterparts.

## Architecture

### Mainframe architecture

IBM z/TPF Mainframe System



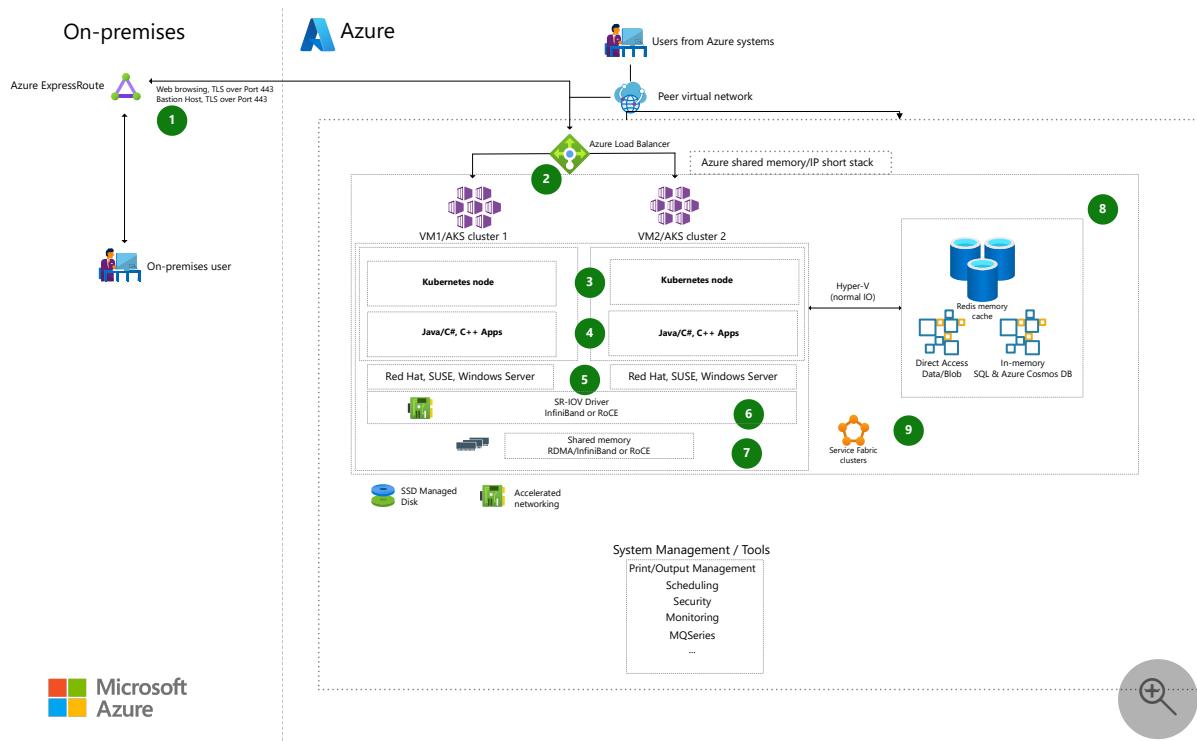
Download a [Visio file](#) of this architecture.

## Dataflow

- Users input data over TCP/IP, including TN3270 and HTTP(S).
- Data is input into the mainframe via standard mainframe protocols.
- Applications receive the data. These applications are usually only online systems. Assembler, C++, or Saber Talk run in an enabled environment.
- NonSQL data and database services, like Azure Cosmos DB, store data.

- Middleware and utility services manage tasks like tape storage, queueing, output, and web services within the environment.
- Shared memory coupling coordinates multiple processors.
- Partitions are used to run separate workloads or segregate work types within the environment.
- Operating systems provide interfaces between the engine and the software it runs.

## Azure architecture



Download a [Visio file](#) of this architecture.

Migrating mainframe systems to Azure requires a platform that supports a high-performance memory-sharing mechanism and a high-performance IP stack with low latency (microseconds) to deliver performance at the level of the z/TPF.

The z/TPF mainframe uses a shared memory feature called a *coupling facility* together with a low-latency IP stack called *HiperSockets*. This architecture uses drivers that provide shared I/O and memory across multiple nodes to implement similar functionality.

The Azure Cosmos DB NoSQL database is used for high-performance storage. This storage solution provides high speed and high-performance data persistence and retrieval.

## Dataflow

1. Input, typically via either Azure ExpressRoute from remote clients or via other applications currently running in Azure. In either case, TCP/IP connections provide the primary means of connection to the system. User access for web-based applications is provided over TLS port 443. To improve security by minimizing open ports, you can use Azure Bastion hosts for admin access to the VMs.
2. On Azure, an Azure load balancer is used to access the application compute clusters. Kubernetes provides robust load balancing and scaling. In this case, the front-end load balancer provides another level of failover capability to maintain business continuity if an entire cluster service goes down.
3. VMs, Kubernetes, or virtual machine scale sets are used for deployment.
4. Application servers receive the input in the compute clusters and share application state and data by using Azure Cache for Redis or Remote Direct Memory Access (RDMA).
5. The architecture runs on Red Hat Enterprise Linux, SUSE Linux, or Windows.
6. A single root I/O virtualization (SR-IOV) driver is used to meet performance requirements. The SR-IOV enables multiple VMs to share the same PCIe physical hardware resources. The driver used here is either RDMA over Converged Ethernet (RoCE) or InfiniBand over Ethernet (IBoE). These drivers allow communication between two hosts in the same Ethernet broadcast domain via an Ethernet link layer.
7. RDMA/InfiniBand or RoCE drivers allow the two hosts to share memory as one pool.
8. Azure Cache for Redis provides a caching solution that improves application response time by storing copies of the most frequently used data and the session state.
9. Service Fabric clusters provide container orchestration.

## Components

This solution features the following Azure components:

- [Azure ExpressRoute](#) extends your on-premises networks into Azure over a private, dedicated fiber connection from a connectivity provider. ExpressRoute establishes connections to cloud services like Azure and Microsoft 365.
- [Azure Bastion](#) is a fully managed service that helps secure remote access to your virtual machines.
- [Azure Load Balancer](#) distributes incoming traffic to the compute resource clusters. You can define rules and other criteria to distribute the traffic.
- [Azure Kubernetes Service \(AKS\)](#) is a fully managed Kubernetes service for deploying and managing containerized applications. AKS offers serverless

Kubernetes, an integrated continuous integration and continuous delivery (CI/CD) experience, and enterprise-grade security and governance.

- [Azure Virtual Network](#) is the fundamental building block of Azure private networks. Azure VMs in virtual networks can communicate, with enhanced security, with each other, the internet, and on-premises networks. A virtual network is like a traditional on-premises network, but it provides Azure infrastructure benefits like scalability, high availability, and isolation.
- [Azure Cache for Redis](#) adds a quick caching layer to application architecture to handle large volumes at high speed. Azure Cache for Redis scales performance simply and cost-effectively, providing the benefits of a fully managed service.
- [Azure databases](#) offer a choice of fully managed relational and NoSQL databases to fit modern application needs. Automated infrastructure management provides scalability, availability, and security.
  - [Azure Cosmos DB](#) is a fully managed, fast NoSQL database with open APIs for any scale.

## Alternatives

This solution supports deployment in containers, VMs, or virtual machine scale sets. Unlike VMs, containers and scale sets can scale in and out rapidly. Because the unit of scaling is containers, infrastructure utilization is optimized.

You can use the legacy web-application presentation layer virtually unchanged to minimize user retraining. Alternatively, you can update the web-application presentation layer with modern UX frameworks.

## Scenario details

Mainframe systems are expensive to maintain, and the pool of developers who understand these systems is diminishing. But organizations migrating from z/TPF mainframes to the cloud need a robust platform that can process high volume transactions. This solution delivers cloud-enabled applications and databases that are functionally equivalent to the z/TPF legacy counterparts.

The solution is designed to meet these requirements:

- Refactored applications must remain functionally equivalent to their original counterparts.
- Refactored applications must perform as well as or better than the original applications.

- Refactored applications must be cloud-ready and delivered via a standard DevOps toolchain and implement DevOps best practices.

## Potential use cases

Here are some scenarios that can benefit from refactoring to Azure:

- Modernize infrastructure and avoid the high costs, limitations, and rigidity that are associated with mainframes.
- Reduce operational and capital expenditure.
- Move mainframe workloads to the cloud without the side effects of rewrites.
- Migrate mission-critical applications but maintain continuity with other on-premises applications.
- Take advantage of the horizontal and vertical scalability that Azure provides.
- Implement solutions that provide disaster recovery.

## Considerations

These considerations implement the pillars of the Azure Well-Architected Framework, which is a set of guiding tenets that can be used to improve the quality of a workload. For more information, see [Microsoft Azure Well-Architected Framework](#).

## Performance efficiency

This architecture is designed for high volume transactions. It uses Azure compute and shared I/O and memory to create a coupled environment that meets these needs.

To meet the requirements of z/TPS performance, this architecture uses Kubernetes clusters.

## Operations

In addition to supporting faster cloud adoption, refactoring also promotes the adoption of DevOps and Agile working principles. It provides full flexibility in development and production deployment options.

## Resiliency

Kubernetes provides a cluster autoscaler that adjusts the number of nodes required based on the requested compute resources in the node pool. The cluster autoscaler

monitors the Metrics API server every 10 seconds to determine if changes are needed in the node count.

## Security

This architecture is primarily built on Kubernetes, which includes security components like [Pod Security Standards](#) and [Secrets](#). Azure provides additional security features, like Microsoft Entra ID, Microsoft Defender for Containers, Azure Policy, Azure Key Vault, network security groups, and orchestrated cluster upgrades.

[Azure Bastion](#) improves security for admin access by minimizing open ports. Azure Bastion provides highly secure RDP or SSH connectivity to virtual network VMs directly from the Azure portal, over TLS.

## Cost optimization

Use the [Azure pricing calculator](#) to estimate the costs for your implementation of this solution.

## Contributors

*This article is maintained by Microsoft. It was originally written by the following contributors.*

Principal author:

- Marlon Johnson | Senior Program Manager

Other contributors:

- [Mick Alberts](#) | Technical Writer
- [Bhaskar Bandam](#) | Senior Program Manager

## Next steps

For more information, contact [legacy2azure@microsoft.com](mailto:legacy2azure@microsoft.com).

See these additional resources:

- [Azure Kubernetes Service \(AKS\)](#)
- [Welcome to Azure Cosmos DB](#)
- [Azure databases](#)

- What is Azure Virtual Network?

## Related resources

- Mainframe application migration
- Make the switch from mainframes to Azure
- Mainframe access to Azure databases
- Modernize mainframe and midrange data
- Moving archive data from mainframe systems to Azure
- Re-engineer mainframe batch applications on Azure

# Refactor mainframe applications with Astadia

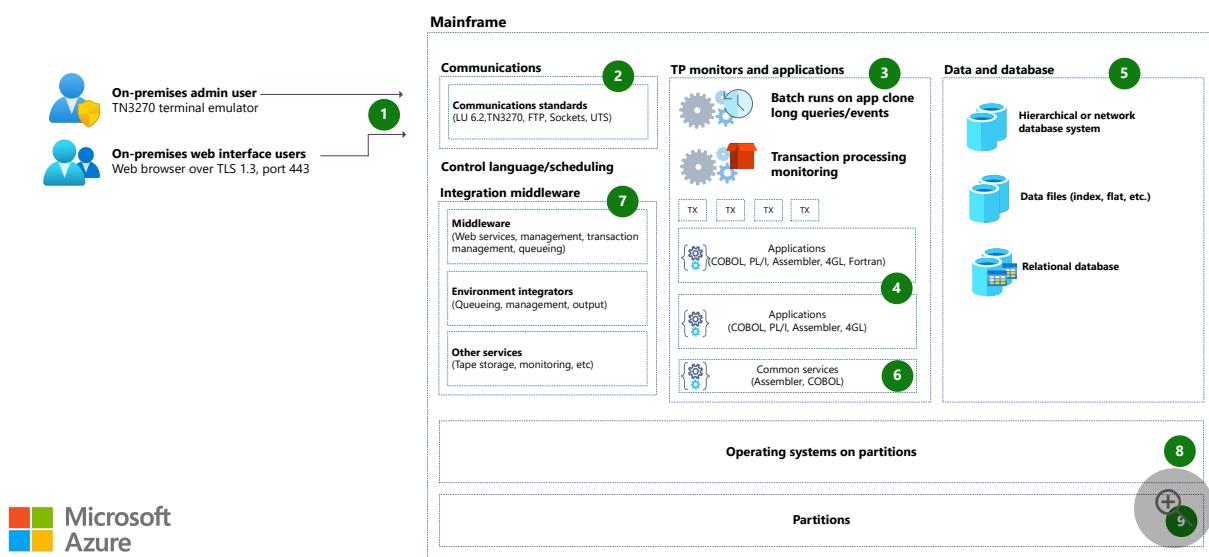
Azure ExpressRoute   Azure Bastion   Azure Load Balancer   Azure Private Link   Azure Site Recovery

Astadia's automated COBOL refactoring solution delivers cloud-enabled applications and databases that do the same things as their legacy counterparts. The refactored applications run as Azure applications in virtual machines provided by Azure Virtual Machines. Azure ExpressRoute makes them available to users, and Azure Load Balancer distributes the load.

## Mainframe architecture

Here's a mainframe architecture that represents the kind of system that's suitable for the Astadia refactoring solution.

Example mainframe system architecture



Download a [Visio file](#) of this architecture.

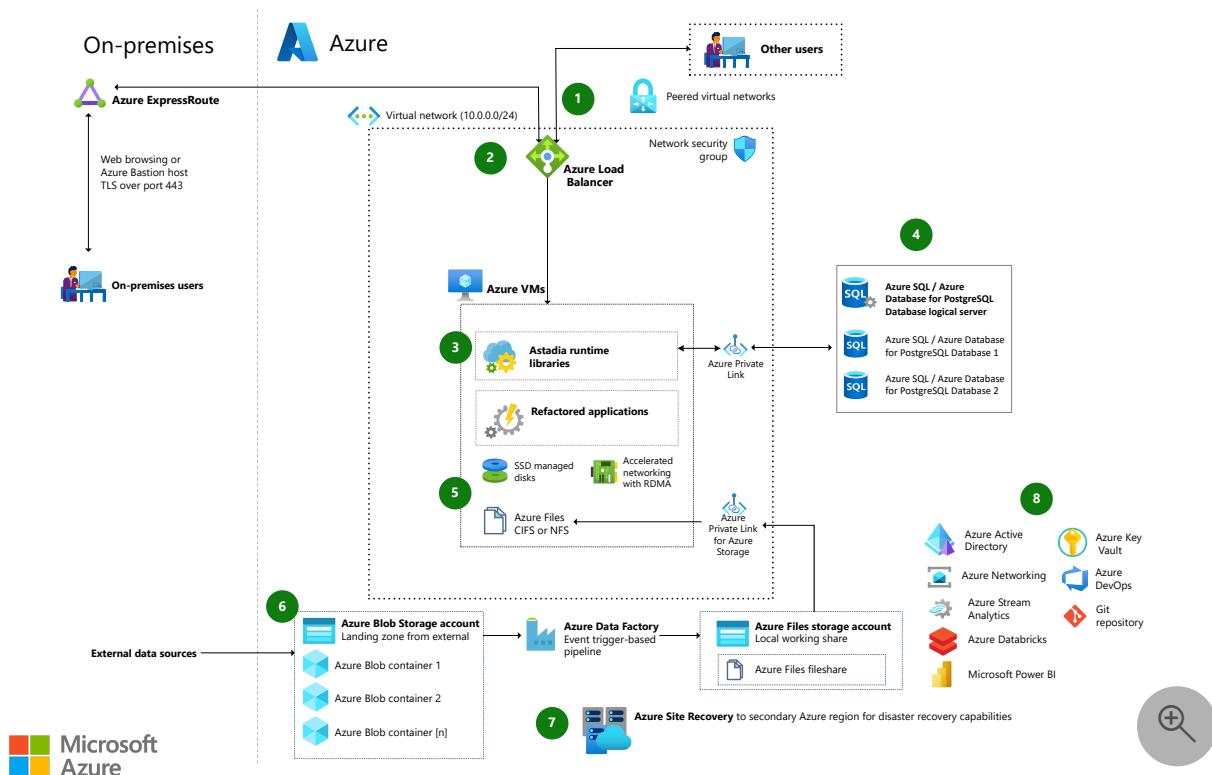
## Dataflow

1. TN3270 and HTTP(S) user input arrives over TCP/IP.
2. Mainframe input uses standard mainframe protocols.
3. There are batch and online applications.

4. Applications written in COBOL, PL/I, Assembler, and other languages run in an enabled environment.
5. Data is held in files and in hierarchical, network, and relational databases.
6. Commonly used services include program execution, I/O operations, error detection, and protection within the environment.
7. Middleware and utility services manage tape storage, queueing, output, and web activity.
8. Each operating system runs in its own partition.
9. Partitions segregate different workloads or work types.

## Azure architecture

Here's an Azure architecture to replace the mainframe functionality with refactored applications.



Download a [Visio file](#) of this architecture.

## Dataflow

1. Input comes from remote clients and other users via ExpressRoute. TCP/IP is the primary way to connect to the system.
  - On-premises users access web-based applications over Transport Layer Security (TLS) port 443. The user interfaces stay the same to minimize end user impact.

user retraining.

- On-premises administrative access uses Azure Bastion hosts.
- Azure users connect to the system via virtual network peering.

2. Load Balancer manages access to the application compute clusters. Load Balancer supports scale-out compute resources to handle input. It operates at level-7, application level, or level-4, network level, depending on the application input.
3. Astadia runtime libraries run refactored applications on Azure Virtual Machines. Compute resources use Azure Premium SSD or Azure Ultra Disk Storage managed disks with accelerated networking.
4. Data services in the application clusters support multiple connections to persistent data sources. Azure Private Link provides private connectivity from inside the virtual network to Azure services. Data sources include data services such as Azure SQL Database and Azure PostgreSQL.
5. Data storage is local-redundant or geo-redundant, depending on usage. It's a mixture of:
  - High-performance storage:
    - Premium SSD
    - Ultra Disk Storage
  - Azure Standard SSD, including blob, archive, and backup storage
6. Azure data services provide scalable and highly available data storage that compute clusters share. The storage can be geo-redundant.
  - Azure Blob Storage serves as a landing zone for data from external data sources.
  - Azure Data Factory ingests data and synchronizes multiple Azure and external data sources.
7. Azure Site Recovery provides disaster recovery for virtual machines (VMs) and container cluster components.
8. Services like Microsoft Entra ID, Azure Networking, Azure DevOps, Azure Stream Analytics, Azure Databricks, GitHub, and Power BI are easily integrated with the modernized system.

## Components

- [ExpressRoute](#) extends on-premises networks into Azure over a private, dedicated fiber connection from a connectivity provider. ExpressRoute establishes connections to Microsoft cloud services like Azure and Microsoft 365.
- [Azure Bastion](#) provides seamless Remote Desktop Protocol (RDP) or secure shell (SSH) connectivity to virtual network VMs from the Azure portal over TLS. Azure

Bastion maximizes administrative access security by minimizing open ports.

- [Load Balancer](#) distributes incoming traffic to the compute resource clusters. It uses configurable rules and other criteria to distribute the traffic.
- [Azure Virtual Machines](#) offers many sizes and types of on-demand, scalable VMs. With Azure Virtual Machines, you get the flexibility of virtualization and you don't have to buy and maintain physical hardware.
- [Azure Virtual Network](#) is the fundamental building block of Azure private networks. VMs within virtual networks communicate securely with each other, with the internet, and with on-premises networks. A virtual network is like a traditional on-premises network, but with Azure infrastructure benefits like scalability, high availability, and isolation.
- [Private Link](#) provides private connectivity from virtual networks to Azure services. Private Link simplifies network architecture and secures the connection between Azure endpoints by eliminating public internet exposure.
- [Azure Storage](#) is scalable, secure cloud storage for all your data, applications, and workloads.
  - [Azure Disk Storage](#) is high-performance, durable block storage for business-critical applications. Azure managed disks are block-level storage volumes that are managed by Azure on VMs. The available types of disks are Ultra Disk Storage, Premium SSD, Standard SSD, and Azure Standard HDD. This architecture uses either Premium SSD or Ultra Disk Storage.
  - [Azure Files](#) provides fully managed file shares in the cloud that are accessed via the industry standard Server Message Block (SMB) protocol. Cloud and on-premises Windows, Linux, and macOS deployments share access by mounting file shares concurrently.
  - [Azure NetApp Files](#) provides enterprise grade Azure file shares that are powered by NetApp. NetApp Files makes it easy for enterprises to migrate and run complex, file-based applications without changing code.
  - [Blob Storage](#) is scalable and secure object storage for archives, data lakes, high-performance computing, machine learning, and cloud-native workloads.
- Azure has fully managed relational, NoSQL, and in-memory databases to fit modern application needs. Automated infrastructure management provides scalability, availability, and security. For an overview of the database types, see [Types of Databases on Azure](#).
  - [SQL Database](#) is a fully managed database engine. SQL Database always runs on the latest stable version of SQL Server and a patched OS with high availability. Built-in database management capabilities include upgrading, patching, backups, and monitoring. With these tasks taken care of, you can focus on domain-specific, business-critical database administration and optimization.

- [Azure Database for PostgreSQL](#) is a fully managed database that's based on the open-source Postgres relational database engine. For applications that require greater scale and performance, the [Hyperscale \(Citus\) deployment option](#) scales queries across multiple machines by sharding them.
- [Azure Cosmos DB](#) is a fully managed, fast NoSQL database with open APIs for any scale.
- [Site Recovery](#) mirrors VMs to a secondary Azure region for quick failover and disaster recovery if an Azure datacenter fails.
- [Data Factory](#) is an extract, transfer, and load (ETL) service for scale-out serverless data integration and data transformation. It offers a code-free UI for intuitive authoring and single-pane-of-glass monitoring and management.

## Scenario details

There are important reasons why companies should replace their COBOL and mainframe systems:

- **Scarcity of domain experience:** Developers who understand COBOL and mainframe technology are retiring, and few developers are trained to replace them. The talent pool gets steadily smaller and the costs and risks of relying on COBOL rise.
- **Limited flexibility:** COBOL and the underlying systems that support it weren't designed for modern cloud-based applications. They're inflexible and hard to integrate.
- **Exorbitant costs:** IBM mainframe hardware and software costs are high. Licensing and maintenance fees for ancillary mainframe applications and databases are rising.

There *is* a way forward for COBOL and mainframe systems. Astadia's automated COBOL refactoring solution delivers cloud-enabled applications and databases that do the same things as their legacy counterparts. The refactored applications run as Azure applications in virtual machines provided by Azure Virtual Machines. Azure ExpressRoute makes them available to users, and Azure Load Balancer distributes the load.

Refactoring reduces costs and allows for deeper integration and for customization to meet business requirements. The hassles and costs of COBOL and the mainframe give way to a new world of quality and scalability that includes:

- Automated testing and quality assurance.
- Docker and Kubernetes for containerized deployment and orchestration.

The refactoring solution creates applications that:

- Are functionally equivalent to their original counterparts.
- Are written in your choice of Java or C#.
- Follow object-oriented concepts and paradigms.
- Are easy to maintain.
- Perform as well as the applications they replace, or better.
- Are cloud-ready.
- Are delivered using a standard DevOps toolchain and best practices.

The refactoring process includes flow normalization, code restructuring, data layer extraction, data remodeling, and packaging for reconstruction. It identifies cloned code and replaces it with shared objects for simpler maintenance and manageability. The process also identifies and removes dead code by analyzing data and control dependencies.

Java and C# developers adapt refactored applications for cloud optimization by using standard DevOps tools and continuous integration and continuous delivery (CI/CD) concepts. Such tools and methods aren't available for mainframe applications. Optimization delivers efficiencies and business benefits such as elasticity, granular service definition, and easy integration with cloud-native services.

## Potential use cases

Automated refactoring is available for most COBOL dialects and platforms, including z/OS, OpenVMS, and VME. Candidates for using it include organizations seeking to:

- Modernize infrastructure and escape the high costs, limitations, and rigidity of mainframe systems.
- Avoid the risks of shortages of COBOL and mainframe developers.
- Reduce operational costs and capital expenditures.
- Move mainframe workloads to the cloud without the costs and risks of prolonged manual rewrites.
- Migrate mission-critical applications to the cloud while maintaining continuity with other on-premises applications.
- Make their systems horizontally and vertically scalable.
- Implement disaster recovery techniques.

## Considerations

The considerations in this section, based on the [Microsoft Well-Architected Framework](#), apply to this solution.

## DevOps

Refactoring not only supports faster cloud adoption, but also promotes adoption of DevOps and agile development principles. You have full flexibility in development and production deployment options.

## Reliability

- The architecture uses Site Recovery to mirror VMs to a secondary Azure region for quick failover and disaster recovery if an Azure datacenter fails.
- The auto-failover groups feature of SQL Database provides data protection by managing database replication and failover to the secondary region. For more information, see [Auto-failover groups overview & best practices \(Azure SQL Database\)](#).
- Resiliency is built into this solution by using Load Balancer. If one presentation or transaction server fails, other servers run the workloads.
- We recommend that you create availability sets for your VMs to increase availability. For more information, see [Availability sets overview](#).
- We recommend that you use geo-replication to increase reliability. For more information, see [Azure Storage redundancy](#).

## Scalability

This solution supports deployment in containers, VMs, or Virtual Machine Scale Sets. Containers and Virtual Machine Scale Sets, unlike VMs, scale out and in rapidly. Shifting the unit of scaling to containers optimizes infrastructure utilization.

## Security

- This solution uses an Azure network security group to manage traffic to and from Azure resources. For more information, see [Network security groups](#).
- Private Link for Azure SQL Database provides a private, direct connection that's isolated to the Azure networking backbone and that runs between VMs and SQL Database.
- Azure Bastion maximizes admin access security by minimizing open ports. It provides secure and seamless RDP/SSH connectivity to virtual network VMs directly from the Azure portal over TLS.

## Cost optimization

- Azure avoids unnecessary costs by identifying the correct number of resource types, analyzing spending over time, and scaling in advance to meet business needs without overspending.
- Azure minimizes costs by running on VMs. You can turn off the VMs that aren't being used, and provide a schedule for known usage patterns. For more information about cost optimization for VMs, see [Virtual Machines](#).
- The VMs in this architecture use either Premium SSD or Ultra Disk Storage. For more information about disk options and pricing, see [Managed Disks pricing](#).
- SQL Database optimizes costs with serverless compute and Hyperscale storage resources that automatically scale. For more information about SQL Database options and pricing, see [Azure SQL Database pricing](#).
- Use the [Pricing calculator](#) to estimate costs for your implementation of this solution.

## Contributors

*This article is maintained by Microsoft. It was originally written by the following contributors.*

Principal author:

- [Richard Cronheim](#) | Senior Program Manager

Other contributor:

- [Bhaskar Bandam](#) | Senior Program Manager

## Next steps

- For more information, contact [legacy2azure@microsoft.com](mailto:legacy2azure@microsoft.com).

## Azure

- [What is Accelerated Networking?](#)
- [How network security groups filter network traffic.](#)
- [Types of Databases on Azure](#)

## Astadia website

- [Migrating Mainframe Applications to Azure](#)
- [United States Air Force \(case study\)](#)

- Jefferson County (case study) ↗

## Other

- Automated Data Migration ↗

## Related sources

- High-volume batch transaction processing
- General mainframe refactor to Azure
- IBM z/OS mainframe migration with Avanade AMT
- IBM z/OS online transaction processing on Azure
- Micro Focus Enterprise Server on Azure VMs
- Refactor IBM z/OS mainframe coupling facility (CF) to Azure
- Refactor mainframe applications with Advanced
- Refactor mainframe computer systems that run Adabas & Natural
- Rehost mainframe applications to Azure with Raincode compilers
- Use LzLabs Software Defined Mainframe (SDM) in an Azure VM deployment
- Migrate IBM mainframe applications to Azure with TmaxSoft OpenFrame

# Refactor mainframe applications with Advanced

Azure Files

Azure Load Balancer

Azure SQL Database

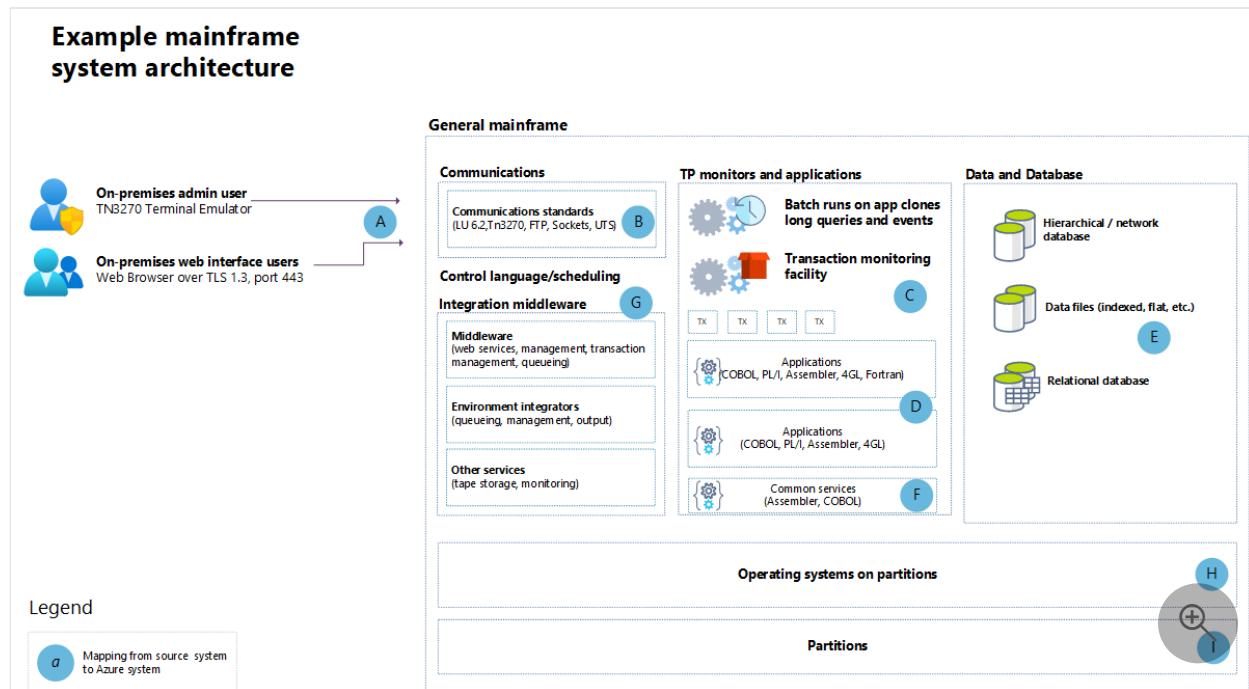
Azure Storage

Azure Virtual Machines

Advanced's Automated COBOL Refactoring solution refactors COBOL applications, as well those written in CA-Gen, CA-Telon, Natural, ADSO and other legacy languages, to deliver cloud-enabled applications and databases that are functionally equivalent to their legacy counterparts. This reduces costs, allows for deeper integration, and enables customization to meet business requirements. In addition, it unlocks a whole new world of quality and scalability, from automated testing to quality assurance, and the ability to benefit from containerized deployments and orchestration with Docker and Kubernetes.

## Mainframe architecture

Here's an example system where automated factoring can be used:



## Workflow

- Users provide input over TCP/IP, using protocols such as TN3270, HTTP, and HTTPS.
- Input arrives using standard mainframe protocols.
- Batch and online applications process the input.

D. COBOL, PL/I, Assembler, and compatible languages run in an enabled environment.

E. Files and databases provide data storage. The database types include hierarchical, network, and relational.

F. Services perform tasks for the applications. Services that are commonly enabled include program execution, I/O operations, error detection, and protection.

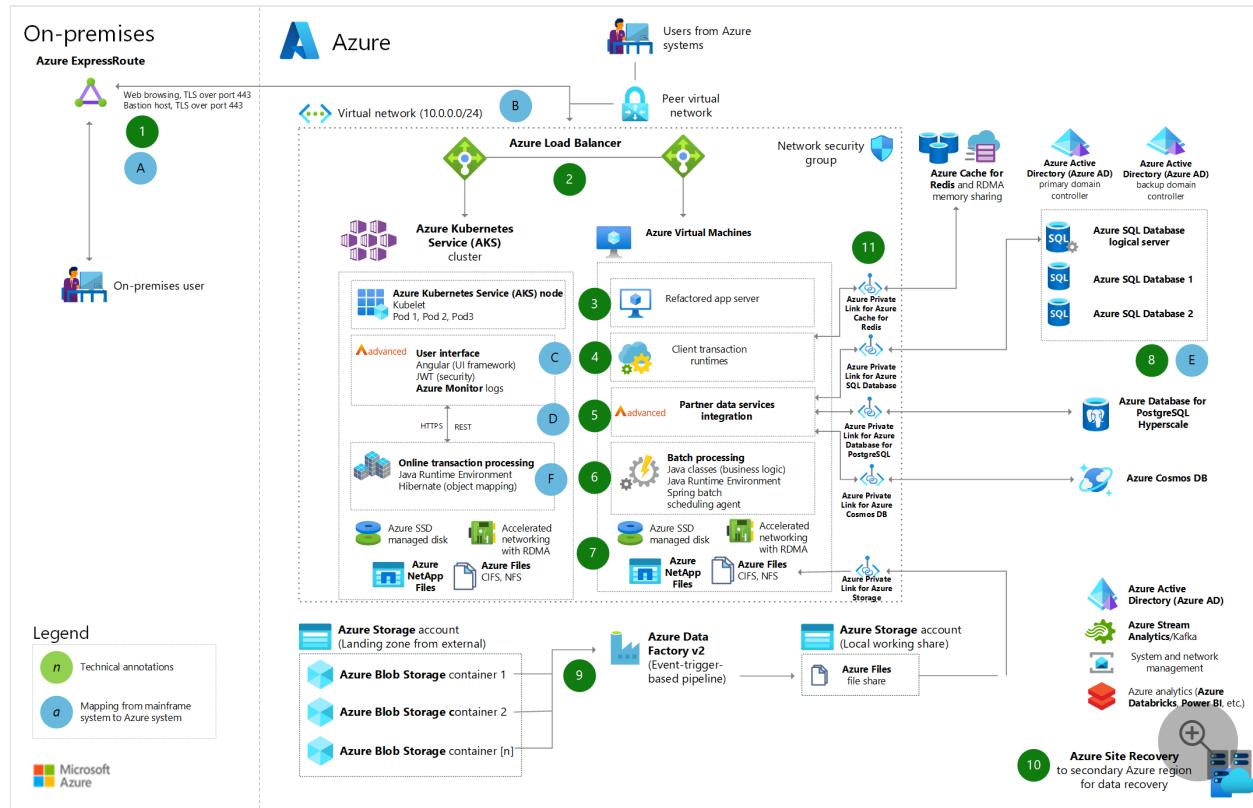
G. Middleware and utility services manage such tasks as tape storage, queueing, output, and web support.

H. Operating systems provide the interface between the engine and the software that it runs.

I. Partitions run separate workloads, or segregate work types within the environment.

## Azure architecture

This is the architecture of the example system shown above when refactored for Azure. Note that the letter callouts in the diagrams reveal where the refactored solution handles the corresponding mainframe functionality.



Download a [Visio file](#) of this architecture.

## Workflow

1. Input typically comes either through Azure ExpressRoute from remote clients, or from other Azure applications. In either case, TCP/IP connections are the primary means of connecting to the system. User access to web applications is over TLS port 443. You can keep the UI of the web applications the same to minimize end user retraining, or you can update it by using modern UX frameworks. Azure Bastion provides admin access to the virtual machines (VMs), maximizing security by minimizing open ports.
2. Once in Azure, access to the application compute clusters is through an Azure load balancer. This approach allows for scale-out compute resources to process the input work. Depending on input, you can load balance at either the application level or the network-protocol level.
3. Advanced supports deployment in containers, VMs, or Virtual Machine Scale Sets. Containers and Virtual Machine Scale Sets, unlike VMs, can scale out and in rapidly. Shifting the unit of scaling to containers optimizes infrastructure utilization.
4. Application servers receive the input in the compute clusters, and share application state and data using Azure Cache for Redis or Remote Direct Memory Access (RDMA).
5. Data services in the application clusters allow for multiple connections to persistent data sources. Possible data sources include:
  - Azure SQL Database.
  - Azure Cosmos DB.
  - Databases on VMs, such as Oracle and Db2.
  - Big data repositories such as Azure Databricks and Azure Data Lake.
  - Streaming data services such as Kafka and Azure Stream Analytics.
6. The application servers host various application programs based on the language's capability, such as Java classes or COBOL programs.
7. Data services use a combination of:
  - a. **High-performance storage:** Azure Premium SSD and Azure Ultra Disk Storage.
  - b. **File storage:** Azure NetApp Files and Azure Files.
  - c. **Standard storage:** Azure Blob Storage, archive, and backup. The backup can be:
    - i. Locally Redundant Storage (LRS).
    - ii. Zone-redundant storage (ZRS).
    - iii. Geo-redundant storage (GRS).
    - iv. Geo-zone-redundant storage (GZRS).

For more information on redundancy, see [Azure Storage redundancy](#).

8. Azure platform as a service (PaaS) data services provide scalable and highly available data storage to share across multiple compute resources in a cluster. These can also be geo-redundant.
9. Azure Data Factory can ingest data and synchronize with multiple data sources both within Azure and from external sources. Azure Blob storage is a common landing zone for external data sources.
10. Azure Site Recovery provides for disaster recovery of the VM and container cluster components.
11. Applications connect to private endpoints of the various PaaS services.

## Components

This example features the following Azure components. Several of these components and workflows are interchangeable or optional depending on your scenario.

- [Azure ExpressRoute](#) extends your on-premises networks into Azure over a private, dedicated fiber connection from a connectivity provider. ExpressRoute establishes connections to Microsoft cloud services like Azure and Microsoft 365.
- [Azure Bastion](#) provides seamless Remote Desktop Protocol (RDP) or secure shell (SSH) connectivity to virtual network VMs from the Azure portal over Transport Layer Security (TLS). Azure Bastion maximizes admin access security by minimizing open ports.
- [Azure Load Balancer](#) distributes incoming traffic to the compute resource clusters. You can define rules and other criteria to distribute the traffic.
- [Azure Kubernetes Service \(AKS\)](#) is a fully managed Kubernetes service to deploy and manage containerized applications. AKS offers serverless Kubernetes, an integrated continuous integration and continuous delivery (CI/CD) experience, and enterprise-grade security and governance.
- [Azure Virtual Machines](#) offers many sizes and types of on-demand, scalable computing resources. With Azure VMs, you get the flexibility of virtualization without having to buy and maintain physical hardware.
- [Azure Virtual Network](#) is the fundamental building block of Azure private networks. VMs within virtual networks can communicate securely with each other, the internet, and on-premises networks. A virtual network is like a traditional on-premises network, but with Azure infrastructure benefits like scalability, high availability, and isolation.

- [Azure Private Link](#) provides private connectivity from a virtual network to Azure services. Private Link eliminates public internet exposure to simplify network architecture and secure the connections between Azure endpoints.
- [Azure Cache for Redis](#) adds a quick caching layer to application architecture to handle large volumes at high speed. Azure Cache for Redis scales performance simply and cost-effectively, with the benefits of a fully managed service.
- [Azure Storage](#) is scalable, secure cloud storage for all your data, applications, and workloads.
  - [Azure Disk Storage](#) is high-performance, durable block storage for business-critical applications. Azure managed disks are block-level storage volumes that are managed by Azure on Azure VMs. The available types of disk storage are Ultra Disk Storage, Premium SSD, Standard SSD, and Standard HDD. This architecture uses either Premium SSD or Ultra Disk Storage.
  - [Azure Files](#) offers fully managed file shares in the cloud that are accessible via the industry standard Server Message Block (SMB) protocol. Cloud and on-premises Windows, Linux, and macOS deployments can mount file shares concurrently.
  - [Azure NetApp Files](#) provides enterprise-grade Azure file shares that are powered by NetApp. Azure NetApp Files makes it easy for enterprises to migrate and run complex, file-based applications with no code changes.
  - [Azure Blob Storage](#) is scalable and secure object storage for archives, data lakes, high-performance computing, machine learning, and cloud-native workloads.
- [Azure databases](#) offer a choice of fully managed relational and NoSQL databases to fit modern application needs. Automated infrastructure management provides scalability, availability, and security.
  - [Azure SQL Database](#) is a fully managed PaaS database engine. SQL Database always runs on the latest stable version of SQL Server and a patched OS with high availability. Built-in PaaS database management capabilities include upgrading, patching, backups, and monitoring. You can focus on domain-specific, business-critical database administration and optimization.
  - [Azure Database for PostgreSQL](#) is a fully managed database based on the open-source PostgreSQL relational database engine. The Hyperscale (Citus) deployment option scales queries across multiple machines by using sharding, for applications that require greater scale and performance.
  - [Azure Cosmos DB](#) is a fully managed, fast NoSQL database with open APIs for any scale.
- [Azure Site Recovery](#) mirrors Azure VMs to a secondary Azure region for quick failover and data recovery if an Azure datacenter fails.

- [Azure Data Factory](#) is an extract, transform, and load (ETL) service for scale-out, serverless data integration and data transformation. It offers a code-free UI for intuitive authoring and single-pane-of-glass monitoring and management.

## Scenario details

There are many reasons to look for alternatives to the COBOL-based mainframe applications that are still common:

- COBOL and CA-Gen/Natural/Telon/ASDO developers are retiring and no one is trained to replace them, resulting in a steadily diminishing talent pool. As the talent shortage grows, the costs and risks of relying on COBOL and other legacy languages increase.
- The applications weren't designed for modern IT, resulting in difficult integrations and limited flexibility.
- IBM mainframe hardware and software are expensive, and licensing and maintenance fees for ancillary mainframe applications and databases are rising.

Advanced's Automated COBOL Refactoring solution refactors COBOL applications, as well those written other legacy languages, to deliver cloud-enabled applications and databases that are functionally equivalent to their legacy counterparts. This reduces costs, allows for deeper integration, and enables customization to meet business requirements. In addition, it unlocks a whole new world of quality and scalability, from automated testing to quality assurance, and the ability to benefit from containerized deployments and orchestration with Docker and Kubernetes.

The refactored applications:

- Are functionally equivalent to the originals.
- Are easy to maintain—they attain SonarQube A ratings, and follow object-oriented concepts and paradigms.
- Perform as well as, or better than, the originals.
- Are cloud-ready and delivered by using a standard DevOps toolchain and best practices.

The refactoring process includes flow normalization, code restructuring, data layer extraction, data remodeling, and packaging for reconstruction. The process identifies cloned code and creates shared replacement objects, simplifying maintenance and manageability. Complex data and control dependency analysis locates and removes dead code.

Once the Advanced solution refactors the COBOL applications and associated databases, Java and C# developers can use standard DevOps tools and CI/CD concepts to extend application functionality. The refactoring process preserves business logic and optimizes performance. Additional benefits include elasticity, granular service definition, and easy integration with cloud-native services.

Automated COBOL Refactoring is available for most COBOL dialects and platforms, including z/OS, OpenVMS, and VME.

## Potential use cases

Advanced refactoring benefits many scenarios, including:

- Businesses seeking to:
  - Modernize infrastructure and escape the exorbitant costs, limitations, and rigidity associated with mainframes.
  - Avoid the risk associated with skills shortages around legacy systems and applications by going cloud-native and DevOps.
  - Reduce operational and capital expenditure costs.
- Organizations wanting to migrate mainframe workloads to the cloud without costly and error-prone manual rewrites.
- Organizations that need to migrate business-critical applications while maintaining continuity with other on-premises applications.
- Teams looking for the horizontal and vertical scalability that Azure offers.
- Businesses that favor solutions that have disaster recovery options.

## Considerations

Incorporate the following pillars of the [Microsoft Azure Well-Architected Framework](#) for a highly available and secure system:

### Availability

- The architecture uses [Azure Site Recovery](#) to mirror Azure VMs to a secondary Azure region for quick failover and disaster recovery if an Azure datacenter fails.
- [Azure auto-failover group replication](#) manages the database replication and failover to the secondary region.

### Operations

Refactoring not only supports faster cloud adoption, but also promotes adoption of DevOps and Agile working principles. You have full flexibility in development and production deployment options.

## Security

This solution uses an Azure network security group to manage traffic between Azure resources. For more information, see [Network security groups](#).

Private Link for Azure SQL Database provides a private, direct connection that's isolated to the Azure networking backbone from the Azure VMs to Azure SQL Database.

[Azure Bastion](#) maximizes admin access security by minimizing open ports. Bastion provides secure and seamless RDP/SSH connectivity to virtual network VMs directly from the Azure portal over TLS.

## Resiliency

Resiliency is built into this solution by the load balancers. If one presentation or transaction server fails, other servers behind the load balancers can run the workloads according to the rules and health probes. Availability sets and geo-redundant storage are highly recommended.

## Cost optimization

Azure avoids unnecessary costs by identifying the correct number of resource types, analyzing spending over time, and scaling to meet business needs without overspending.

- Azure provides cost optimization by running on VMs. You can turn off the VMs when they're not in use, and script a schedule for known usage patterns. See the [Azure Well-Architected Framework](#) for more information about cost optimization for VM instances.
- The VMs in this architecture use either Premium SSD or Ultra Disk Storage. For more information about disk options and pricing, see [Managed Disks pricing](#).
- SQL Database optimizes costs with serverless compute and Hyperscale storage resources that automatically scale. For more information about SQL Database options and pricing, see [Azure SQL Database pricing](#).
- Use the [Pricing calculator](#) to estimate costs for your implementation of this solution.

# Contributors

*This article is maintained by Microsoft. It was originally written by the following contributors.*

Principal author:

- [Bhaskar Bandam](#) | Senior TPM

*To see non-public LinkedIn profiles, sign in to LinkedIn.*

## Next steps

- For more information, please contact [legacy2azure@microsoft.com](mailto:legacy2azure@microsoft.com).
- [Modernization Platform as a Service \(ModPaaS\)](#)
- [Advanced's Automated COBOL Refactoring solution](#)
- [Case study: Modernizing to the Cloud While Racing the Clock](#).

## Related resources

- [Azure mainframe and midrange architecture concepts and patterns](#)
- [IBM z/OS mainframe migration with Avanade AMT](#)
- [Rehost mainframe applications to Azure with Raincode compilers](#)

# Refactor mainframe computer systems that run Adabas & Natural

Azure Kubernetes Service (AKS)

Azure ExpressRoute

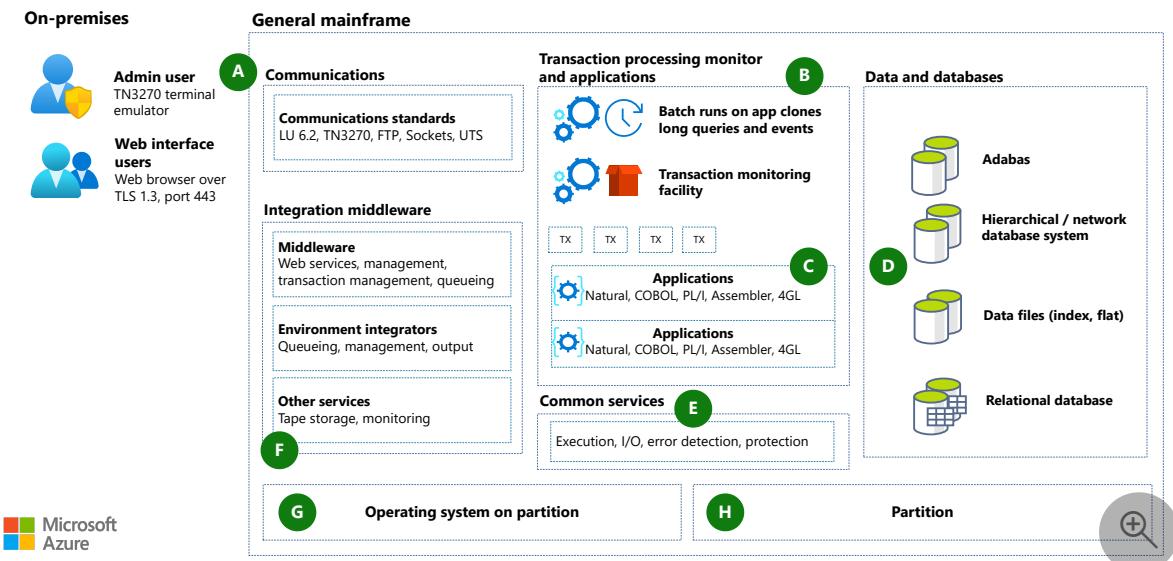
Azure Managed Disks

Azure NetApp Files

Software AG provides a popular 4GL mainframe platform that's based on the Natural programming language and the Adabas database. This article provides an architecture for organizations that are using mainframe computers that run Adabas & Natural and that are looking for ways to modernize these workloads and move them to the cloud.

## Mainframe architecture

This diagram illustrates an example of a mainframe with Software AG's Adabas & Natural modules installed, before migration to Azure. This example shows an IBM z/OS architecture.



Download a [Visio file](#) of this architecture.

## Workflow

A. Input occurs over TCP/IP, including TN3270 and HTTP(S). Input into the mainframe uses standard mainframe protocols.

B. Receiving applications can be either batch or online systems.

C. Natural, COBOL, PL/I, Assembler, or other compatible languages run in an enabled environment.

D. Data and database services commonly used are hierarchical/network database systems and relational database types.

E. Common services include program execution, I/O operations, error detection, and protection within the environment.

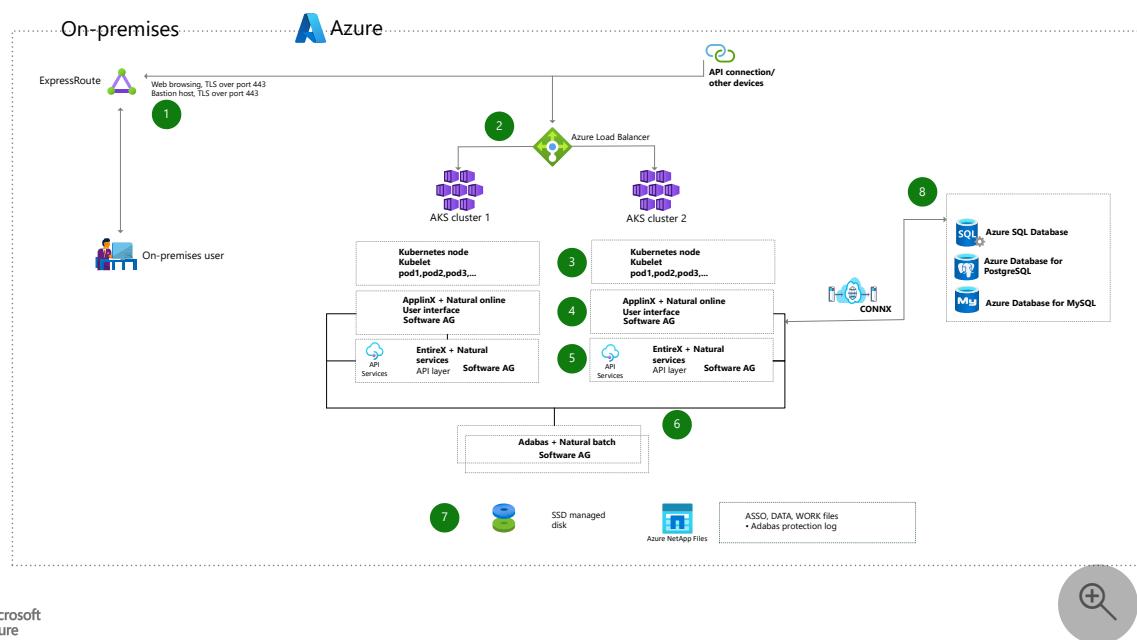
F. Middleware and utilities manage services like tape storage, queueing, output, and web services within the environment.

G. Operating systems provide the interface between the engine and the software that it runs.

H. Partitions are needed to run separate workloads and to segregate work types within the environment.

## Azure architecture

This diagram shows how you can migrate the legacy architecture to Azure by using a refactoring approach to modernize the system:



Download a [Visio file](#) of this architecture.

## Workflow

- 1. Input.** Input typically occurs either via Azure ExpressRoute from remote clients or via other applications currently running Azure. In either case, TCP/IP connections are the primary means of connection to the system. TLS port 443 provides access to web-based applications. You can leave the web-based applications presentation layer virtually unchanged to minimize user retraining. Alternatively, you can update this layer with modern UX frameworks per your requirements. For admin access to the VMs, you can use Azure Bastion hosts to maximize security by minimizing open ports.
- 2. Access in Azure.** In Azure, access to the application compute clusters is provided via an Azure load balancer. This approach allows scale-out compute resources to process the input work. Both level 7 (application level) and level 4 (network protocol level) load balancers are available. The type that you use depends on how the application input reaches the entry point of the compute cluster.
- 3. Application compute clusters.** The architecture supports applications that run in a container that can be deployed in a container orchestrator like Kubernetes. Adabas & Natural components can run inside container technology operated on top of a Linux operating system. You can re-architect your legacy applications to modern container-based architectures and operate on top of Azure Kubernetes Services.
- 4. ApplinX terminal emulation** (Software AG). ApplinX is a server-based technology that provides web connectivity and integration into core system applications without requiring changes to the applications. **Natural Online** enables online users to connect to Natural applications via a web browser. Without ApplinX, users need to connect with terminal emulation software by using SSH. Both systems run in containers.
- 5. EntireX** (Software AG). EntireX enables you to easily connect services that run on Integration Server to mission-critical programs that are written in languages like COBOL and Natural. **Natural Business Services** enables API access to business functions that are programmed in Natural. Both systems run in containers.
- 6. Adabas** (Software AG). Adabas is a high performance NoSQL database management system. **Natural batch** (Software AG) is a dedicated component for running batch jobs. Natural batch jobs, which are scheduled by a batch job scheduling system that you choose, should run on the same node as the Adabas database to avoid performance impact.
- 7. Storage.** Data services use a combination of high performance storage (ultra / premium SSD), file storage (NetApp), and standard storage (Blob, archive, backup) that can be either local redundant or geo-redundant, depending on the use. Node

operating systems use managed disk storage. All persistent data, like database files, protection logs, application data, and backup, use Azure NetApp Files. AKS manages operating system volumes that are stored in managed disks. All business-critical data from the databases, including ASSO, DATA, WORK files, and Adabas protection logs, should be written to separate volumes that can be provided by Azure NetApp Files.

8. **CONNX.** The CONNX for Adabas module provides highly secure, real-time read/write access to Adabas data sources on OS/390, z/OS, VSE, Linux, Solaris, HP-UX, AIX, and Windows via .NET, ODBC, OLE DB, and JDBC. CONNX connectors provide access to Adabas data sources and expose them to more common databases, like Azure SQL Database, Azure Database for PostgreSQL, and Azure Database for MySQL.

## Components

- [Azure ExpressRoute](#) ↗ extends your on-premises networks into the Microsoft cloud over a private connection that's facilitated by a connectivity provider. You can use ExpressRoute to establish connections to Microsoft cloud services like Azure and Office 365.
- [Azure Kubernetes Service](#) ↗ is a fully managed Kubernetes service for deploying and managing containerized applications. AKS provides serverless Kubernetes, integrated continuous integration and continuous delivery (CI/CD), and enterprise-grade security and governance.
- [Azure managed disks](#) are block-level storage volumes that are managed by Azure and used with Azure Virtual Machines. Various types are available: ultra disks, premium SSD, standard SSD, and standard HDD. SSD disks are used in this architecture.
- [Azure NetApp Files](#) ↗ provides enterprise-grade Azure file shares powered by NetApp. Azure NetApp Files makes it easy to migrate and run complex, file-based applications without changing code.

## Scenario details

Applications running on mainframe computers have been at the core of most business operations for almost 50 years. While these mainframe systems have provided remarkable reliability over the years, they've become somewhat problematic because they're rigid and, in some cases, hard to maintain and costly to operate.

Many organizations are looking for ways to modernize these systems. They're looking for ways to free up the constrained resources that are required to maintain these systems, control their costs, and gain more flexibility in interactions with the systems.

Software AG provides a popular 4GL mainframe platform that's based on the Natural programming language and the Adabas database.

There are two patterns that allow you to run Adabas & Natural applications on Azure: [rehost and refactor](#). This article describes how to refactor an application by using containers that are managed in Azure Kubernetes Service (AKS). For more information, see [Container-based approach](#), later in this article.

## Potential use cases

This architecture applies to any organization that uses mainframe computers running Adabas & Natural and that plans to modernize these workloads and move them to the cloud.

## Considerations

### Container-based approach

To make the most of the flexibility, reliability, and capabilities of Azure, you need to rearchitect mainframe applications. We recommend that you rewrite monolithic applications as microservices and use a container-based approach to deployment. A container bundles all the software that's needed for execution into one executable package. It includes an application's code together with the related configuration files, libraries, and dependencies that are required to run the app. Containerized applications are quick to deploy and support popular DevOps practices like continuous integration (CI) and continuous deployment (CD).

Adabas & Natural containers run in pods, each of which performs a specific task. Pods are units of one or more containers that stay together on the same node and share resources like the host name and IP address. Because they're decoupled from the underlying platform, components in pods scale independently and support higher availability. A containerized application is also portable: it runs uniformly and consistently on any infrastructure.

Containerized services and their associated networking and storage components need to be orchestrated and managed. We recommend AKS, a managed Kubernetes service that automates cluster and resource management. You designate the number of nodes

you need, and AKS fits your containers onto the right nodes to make the best use of resources. AKS also supports automated rollouts and rollbacks, service discovery, load balancing, and storage orchestration. And AKS supports self-healing: if a container fails, AKS starts a new one. In addition, you can safely store secrets and configuration settings outside of the containers.

The architecture diagram in this article shows a container-based implementation of Adabas & Natural. When you set up AKS, you specify the Azure VM size for your nodes, which defines the storage CPUs, memory, and type, like high-performance solid-state drives (SSDs) or regular hard disk drives (HDDs). In this example, Natural runs on three VM instances (nodes) to boost scalability and availability of the user interface (Natural online plus ApplinX) and the API layer (Natural services plus EntireX).

In the data layer, Adabas runs in the AKS cluster, which scales in and out automatically based on resource use. You can run multiple components of Adabas in the same pod or, for greater scale, AKS can distribute them across multiple nodes in the cluster. Adabas uses Azure NetApp Files, a high-performance, metered file storage service, for all persistent data, like database files, protection logs, app data, and backup.

## Operations

Refactoring supports faster cloud adoption. It also promotes adoption of DevOps and Agile working principles. You have full flexibility of development and production deployment options.

## Performance efficiency

Kubernetes provides a cluster autoscaler. The autoscaler adjusts the number of nodes based on the requested compute resources in the node pool. It monitors the Metrics API server every 10 seconds for any required changes in node count. If the cluster autoscaler determines that a change is required, the number of nodes in your AKS cluster is increased or decreased accordingly.

## Security

This architecture is primarily built on Kubernetes, which includes security components like pod security standards and secrets. Azure provides additional features like Microsoft Entra ID, Microsoft Defender for Containers, Azure Policy, Azure Key Vault, network security groups, and orchestrated cluster upgrades.

# Contributors

*This article is maintained by Microsoft. It was originally written by the following contributors.*

Principal author:

- Marlon Johnson | Senior TPM

*To see non-public LinkedIn profiles, sign in to LinkedIn.*

## Next steps

For more information, contact [legacy2azure@microsoft.com](mailto:legacy2azure@microsoft.com).

Here are some additional resources:

- [Adabas & Natural ↗](#)
- [Azure Kubernetes Service](#)
- [Azure NetApp Files documentation](#)
- [Mainframe rehosting on Azure virtual machines](#)
- [Move mainframe compute to Azure Virtual Machines](#)

## Related resources

- [Azure mainframe and midrange architecture concepts and patterns](#)
- [Mainframe migration overview](#)
- [General mainframe refactor to Azure](#)
- [AIX UNIX on-premises to Azure Linux migration](#)

# Rehost mainframe applications to Azure with Raincode compilers

Azure Virtual Machines

Azure Kubernetes Service (AKS)

Azure Files

Azure ExpressRoute

Azure Load Balancer

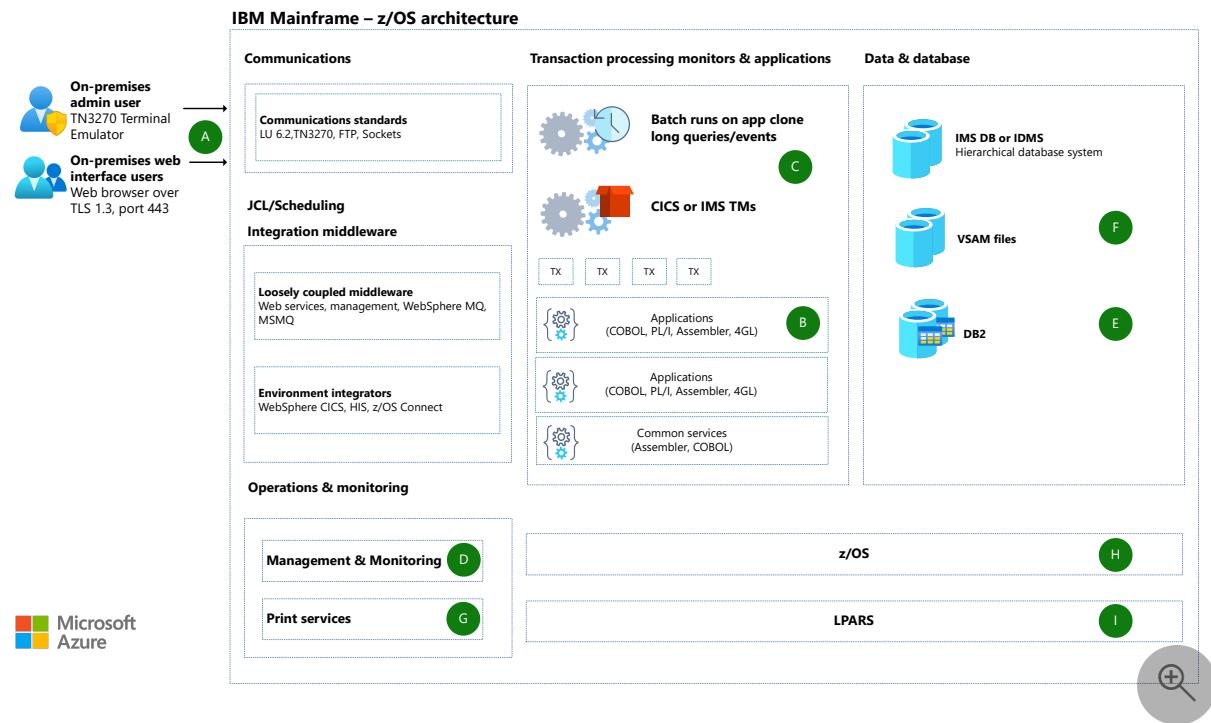
This architecture shows how the Raincode COBOL compiler modernizes mainframe legacy applications by seamlessly migrating and integrating them with a modern, Azure-based technology stack without changing a single line of code. With Raincode's compiler technology, you can keep current optimized mainframe applications and deploy them on the cloud, allowing you to preserve decades of development while greatly enhancing performance and flexibility. Raincode's solution is aimed at transforming the mainframe to an Azure-native architecture by preserving the business logic while transforming the entire architecture. Raincode supports application flexibility across Linux and Windows with containerized or virtual machine (VM) deployments on Azure.

## Architecture

### Legacy IBM z/OS architecture

The following diagram shows an example of a legacy COBOL-based mainframe architecture, before migration to Azure.

## IBM z/OS architecture



Download a [Visio file](#) of this architecture.

## Workflow

The following annotations map from the source IBM z/OS to Azure:

- A. IBM 3270 terminal emulation for demand and online users is replaced by a web browser to access system resources in Azure.
- B. COBOL and other legacy application code is converted to C#/.NET. Raincode generates 100-percent thread-safe and managed code for .NET and .NET Core.
- C. Raincode COBOL compiler modernizes mainframe legacy applications by seamlessly migrating and integrating them with a modern, cloud-based technology stack without changing a single line of code.
- D. Workload automation, scheduling, reporting, and system monitoring functions can retain current platforms, as they are Azure capable today.
- E. Legacy database structures like Db2 and IDMS can be migrated to Azure SQL Database with all the DR/HA capabilities that Azure provides. Raincode also supports static or dynamic SQL queries through SQL Server or on Azure SQL DB.
- F. File structures (VSAM, flat files, virtual tape, and the like) map easily to Azure data constructs within structured files and/or blob storage. Features like redundant

geographic replication and Azure Auto Failover Group Replication are available to provide data protection.

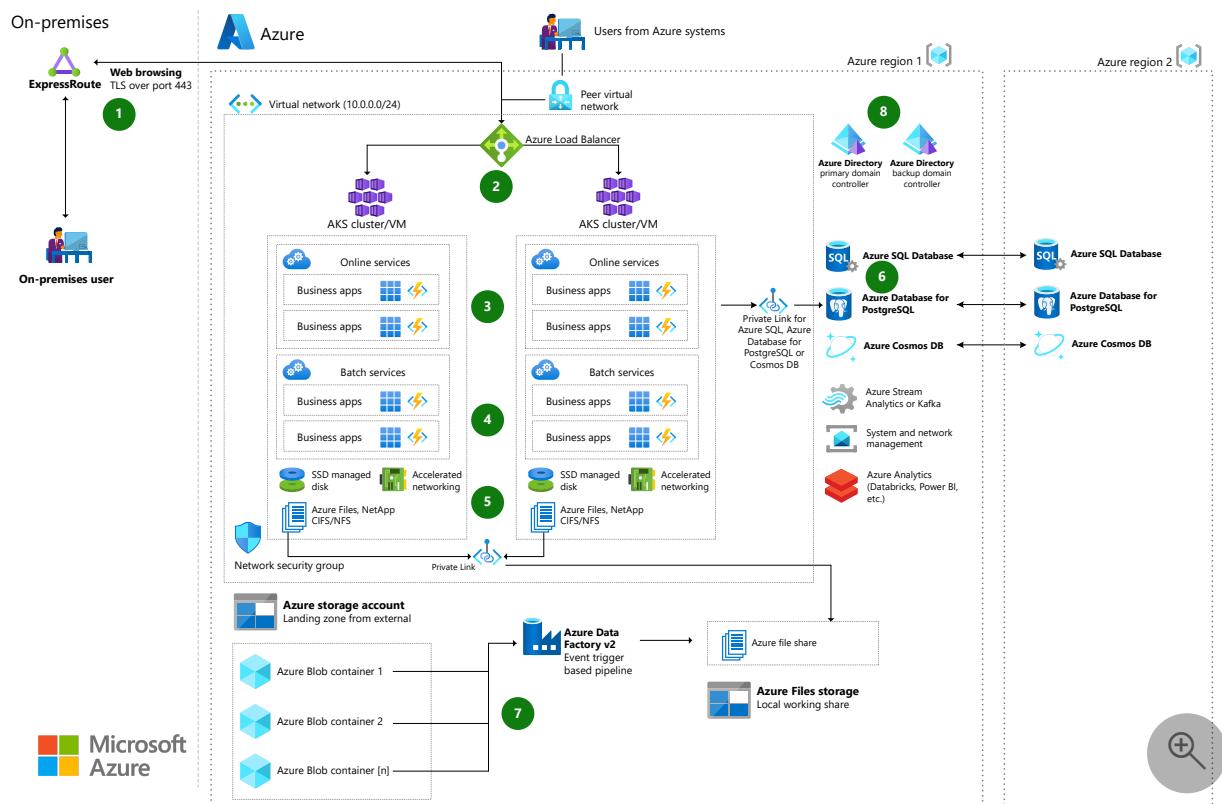
G. An optional printer subsystem manages on-premises printers.

H. z/OS running on Logical Partitions (LPARs).

I. LPARs represent a subset of a computer's hardware resources. Each LPAR can host a separate OS. While this example shows only Z/OS instances, other LPARs running on the same hardware can host other operating environments, like z/VM, or other engines, like zIIP or IFL.

## Postmigration, Azure-based architecture

This diagram shows how the legacy architecture can be migrated to Azure, taking advantage of the Raincode compiler and many other modern Azure services.



Download a [Visio file](#) of this architecture.

## Workflow

1. User access provided over TLS port 443 for accessing web-based applications. Web-based Applications presentation layer can be kept virtually unchanged to minimize end user retraining. Alternatively, the web application presentation layer can be updated with modern UX frameworks as requirements necessitate.

2. In Azure, access to the application compute clusters is through Azure Load Balancer, allowing for scale-out compute resources to process the input work.
3. Raincode system emulation software can also support deployment in containers. With Raincode's cutting-edge compiler technology, you can keep current optimized mainframe applications and deploy them on .NET Core.
4. Cloud-native applications are a collection of independent and autonomous services packaged as lightweight containers.

Unlike virtual machines, containers can scale out and scale in rapidly. Since the unit of scaling shifts to containers, infrastructure utilization is optimized.
5. Data services use a combination of high-performance storage on Ultra or Premium solid-state disks (SSDs), file storage on Azure NetApp Files or Azure Files, and standard blob, archive, and backup storage that can be locally redundant or geo-redundant.
6. Azure SQL Database using either Hyperscale or Business Critical tiers for both high IOPS and high uptime SLA. Further, Private Link for Azure SQL Database is used to provide a private, direct connection isolated to the Azure Networking Backbone from the Azure VM to the Azure SQL Database. Raincode data migration tools can convert DMS/RDMS schemas to SQL.
7. Azure Blob Storage is a common landing zone for external data sources.
8. An implementation of Active Directory needs to be created or already in place. Raincode provides RACF and Top Secret identity integration using Active Directory extensions.

## Components

- [Azure Kubernetes Service \(AKS\)](#) is a fully managed Kubernetes service for deploying and managing containerized applications in container-based compute clusters.
- [Azure Virtual Network \(VNet\)](#) is the fundamental building block for your private network in Azure. VNet enables many types of Azure resources, such as Azure Virtual Machines (VM), to securely communicate with each other, the internet, and on-premises networks. VNet is similar to a traditional network that you'd operate in your own datacenter, but it brings more benefits of Azure's infrastructure, such as scale, availability, and isolation.

- [Azure Files](#) offers fully managed file shares in the cloud that are accessible via the industry-standard Server Message Block (SMB) protocol. Azure file shares can be mounted concurrently by cloud or on-premises deployments of Windows, Linux, and macOS.
- [Azure ExpressRoute](#) lets you extend your on-premises networks into the Microsoft cloud over a private connection facilitated by a connectivity provider. With ExpressRoute you can establish connections to Microsoft cloud services, such as Microsoft Azure and Office 365.
- [Azure Load Balancer](#) operates at layer four of the Open Systems Interconnection (OSI) model. It's the single point of contact for clients. Load Balancer distributes inbound flows that arrive at the load balancer's front end to back-end pool instances. These flows are according to configured load balancing rules and health probes. The back-end pool instances can be Azure Virtual Machines or instances in a virtual machine scale set.
- [Azure SQL Database](#) is a fully managed platform as a service (PaaS) database engine that always runs the latest stable version of SQL Server and patched OS, with 99.99-percent availability. SQL Database handles upgrading, patching, backups, monitoring, and most other database management functions without user involvement. These PaaS capabilities let you focus on business-critical, domain-specific database administration and optimization.
- [Azure Cosmos DB](#) is an Azure PaaS service for NoSQL databases.
- [Azure Database for PostgreSQL](#) is an Azure PaaS service for PostgreSQL databases.

## Scenario details

This architecture illustrates how the Raincode solution runs on Azure. Raincode on Azure supports the following features:

- 100-percent thread-safe and managed code for .NET and .NET Core.
- A solution primarily aimed at transforming mainframes to a cloud-native architecture.
- Native support for static or dynamic SQL queries through SQL Server either on-premises or on Azure SQL DB.
- Support for Db2 (through Microsoft's HIS) and SQL Server.

- Visual Studio integration, featuring a debugger, compiler, configurations, IntelliSense, code colorizer, and project management.
- Support for all COBOL data types, with mainframe memory representation.
- Seamless integration with PL/I and ASM370 compilers.
- A repository with call graphs, statistics, and other compile-time information.
- Native EBCDIC support at compile time and runtime.

Migrating to a modern, distributed cloud infrastructure using Raincode allows you to:

- Facilitate new development and maintenance in C#.
- Free yourself from the financial burden of COBOL licensing costs.
- Adopt a flexible and scalable platform by using the latest technologies through .NET Core.
- Integrate with modern applications such as web and mobile to improve customer experience.
- Transform your monolithic legacy applications into micro- or service-oriented architecture (SOA).
- Control your total cost of ownership (TCO) by using Azure's scalability and availability features.

## Potential use cases

Many use cases can benefit from the Raincode compiler; possibilities include:

- Businesses seeking to modernize infrastructure and escape the high costs, limitations, and rigidity associated with mainframes.
- Reducing Technical Debt by going cloud native and DevOps.
- Reducing operational and capital expenditure costs.
- Organizations opting to move IBM zSeries mainframe workloads to the cloud without the side effects of a complete redevelopment.
- IBM zSeries mainframe customers who need to migrate mission-critical applications while maintaining continuity with other on-premises applications.
- Teams looking for the horizontal and vertical scalability that Azure offers.

- Businesses that favor solutions offering disaster recovery options.
- Taking advantage of the latest software development innovations: tools, frameworks, languages, and practices.

## Considerations

The following considerations apply to this solution.

### Availability

- Raincode architecture uses [Azure Site Recovery](#) to mirror Azure VMs to a secondary Azure region for quick failover and disaster recovery (DR) if an Azure datacenter fails.

### Operations

- Each service of a cloud-native application goes through an independent life cycle, which is managed through an agile DevOps process.
- Multiple continuous integration/continuous delivery (CI/CD) pipelines can work in tandem to deploy and manage a cloud-native application.

### Performance efficiency

- Cloud-native applications are a collection of independent and autonomous services that are packaged as lightweight containers.
- Unlike virtual machines, containers can rapidly scale out and scale in.
- Since the unit of scaling shifts to containers, infrastructure usage is optimized.

### Security

- This solution uses an [Azure network security group \(NSG\)](#) to manage traffic between Azure resources.
- [Private Link for Azure SQL Database](#) provides a private, direct connection that is isolated to the Azure networking backbone, from the Azure VMs to Azure SQL Database.

## Cost optimization

- The Raincode COBOL compiler facilitates new development in C# and eliminates the financial burden of COBOL licensing costs.
- Native support for SQL and CICS. The source code debugged is the same as the source being maintained, rather than the output of a pre-processor.

## Contributors

*This article is maintained by Microsoft. It was originally written by the following contributors.*

Principal author:

- [Jonathon Frost](#) | Principal Software Engineer

*To see non-public LinkedIn profiles, sign in to LinkedIn.*

## Next steps

For more information, please contact [legacy2azure@microsoft.com](mailto:legacy2azure@microsoft.com) or check out the following resources:

- Read the [Raincode technical landscape](#).
- [Mainframe and midrange migration](#)
- [Mainframe rehosting on Azure virtual machines](#)

## Related resources

- [Modernize mainframe & midrange data](#)
- [Mainframe file replication and sync on Azure](#)
- [Replicate and sync mainframe data in Azure](#)
- [Refactor IBM z/OS mainframe Coupling Facility \(CF\) to Azure](#)
- [IBM z/OS mainframe migration with Avanade AMT](#)
- [Migrate IBM mainframe applications to Azure with TmaxSoft OpenFrame](#)

# Rehost a general mainframe on Azure

Azure Virtual Machines

Azure Virtual Network

Azure Container Registry

Azure Kubernetes Service (AKS)

Azure Site Recovery

Rehosting is a way to run legacy mainframe applications, intact, on an open system. This path is the fastest way to take applications off your mainframe hardware and run them on a Windows or Linux platform in a cloud-native environment. Application code written in legacy languages like COBOL or PL/1 are migrated as is and recompiled in the new environment with no change to the business logic. Rehosting helps to preserve the application's logic. At the same time, rehosting minimizes the risk and cost that comes with recoding the application for the new environment.

Rehosting is a cost-effective method to address the challenges of maintaining old mainframe hardware. Commonly referred to as *lift and shift*, rehosting moves mission-critical and core applications off the mainframe and migrates them to the cloud. With this approach, the underlying hardware changes, for example from an IBM mainframe to x86. However, the functional and business logic is untouched. This migration is the quickest and least impactful from an end-user perspective. The application retains the same interfaces and look and feel that the users are comfortable with.

For teams exploring cloud features, rehosting applications is a great way to use cloud capabilities like auto-scaling, managed storage, and containers. This architecture shows a general rehosting example that highlights two methodologies to deploy workloads. You can use Azure Kubernetes Service (AKS) or Azure Virtual Machines. Which method you use depends on the portability of the application, and on your preference.

## Potential use cases

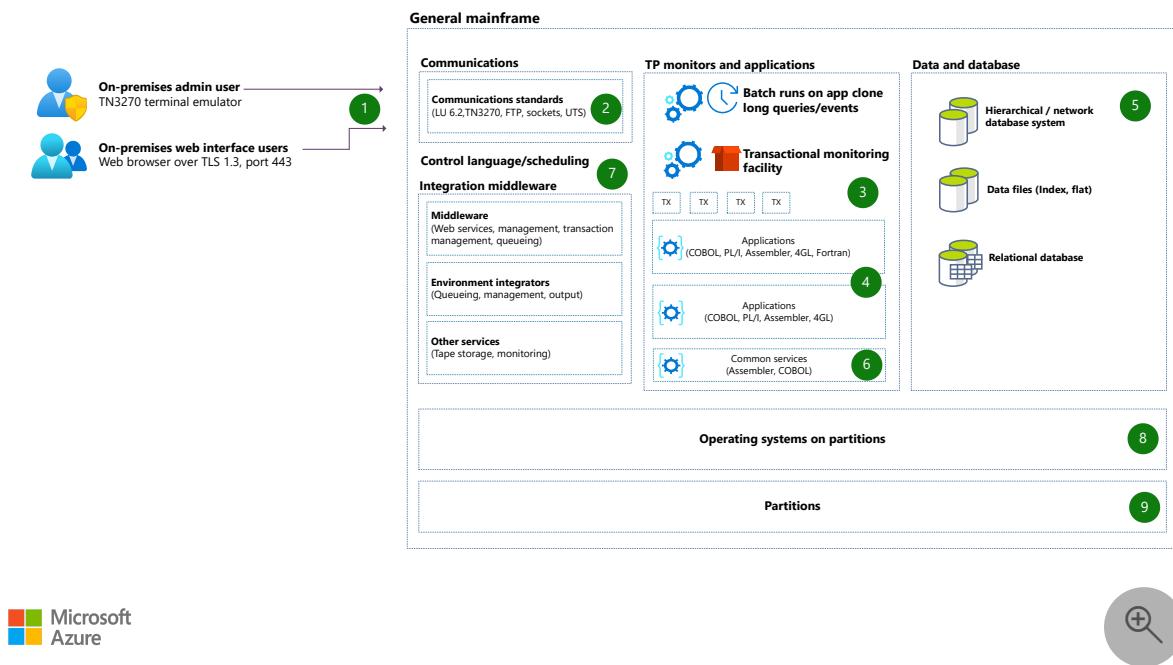
Many scenarios can benefit from rehosting on Azure. Here are some possible use cases:

- **Cost optimization:** You want to significantly reduce the high operating and maintenance costs of mainframes hardware and its associated licenses or software.
- **Location agnostic:** You're planning for a datacenter exit and want a highly available, secure, and reliable alternative platform to host your legacy applications.
- **Least disruption:** You need to migrate mission-critical mainframe applications while maintaining the continuity of day-to-day business operations.

- **Minimal user impact:** Move your applications from old hardware but continue to provide your users with the same or better interfaces.
- **Negligible upskilling:** Applications are rehosted in the cloud with no significant code changes. They continue to provide your development team with the familiar code base, and at the same time eliminate costly development, testing, and reskilling on a newer language.

# Mainframe architecture

This is the pre-migration architecture.



Download a [Visio file](#) of this architecture.

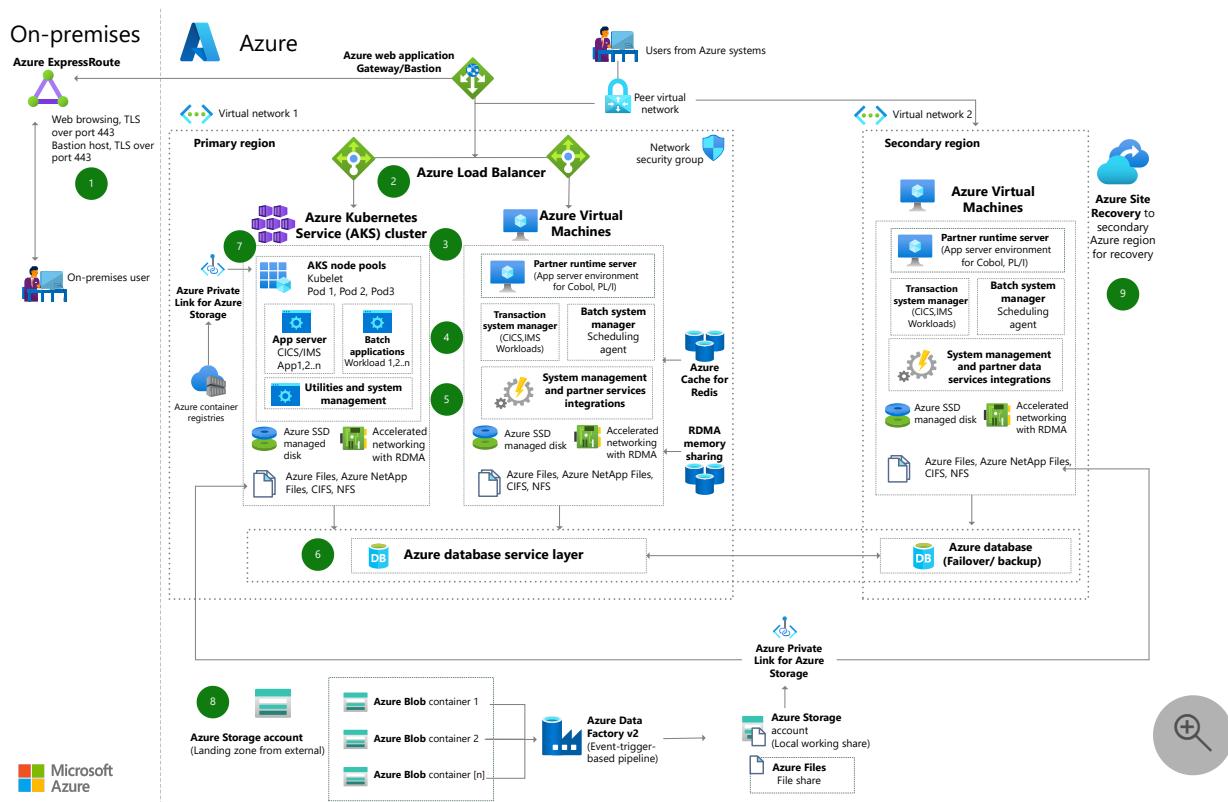
## Mainframe dataflow

1. Input occurs over TCP/IP, including TN3270, HTTP, and HTTPS.
2. Input into the mainframe uses standard mainframe communication protocols.
3. Receiving applications can be batch or online systems.
4. COBOL, PL/I, Assembler, and other compatible languages run in an enabled environment.
5. Data and database services used are hierarchical, network, and relational databases.

6. Common services include program execution, I/O operations, error detection, and protection within the environment.
7. Middleware and utilities manage services like tape storage, queueing, output, and web services within the environment.
8. Operating systems provide the interface between the engine and the software that it runs.
9. Partitions are necessary to run separate workloads and to segregate work types within the environment.

# Architecture

This architecture showcases a solution that is rehosted on Microsoft Azure.



Download a [Visio file](#) of this architecture.

## Dataflow

1. Input typically comes via ExpressRoute from remote clients, or by other applications that currently run in Azure. In either case, TCP/IP connections are the primary means of connection to the system. User access is provided over TLS port 443 to access web-based applications. The presentation layer of the web-based application can be kept unchanged to minimize end user retraining. For admin

access to the VMs, you can use Azure Bastion hosts to maximize security by minimizing open ports.

2. Access to the application compute clusters is done by using an Azure load balancer. With this approach, you can scale out compute resources to process the input work. Both the level 7 application level and level 4 network protocol level load balancers are available. The type you use depends on how the application input reaches the entry point of the computer cluster.
3. The use of application compute clusters depends on whether the application supports virtual machines (VMs) in a compute cluster, or the application runs in a container that you deploy in a container compute cluster like Kubernetes. Most mainframe partner software for applications written in legacy languages prefers to use VMs. Some mainframe systems partner software can also support deployment in containers.
4. Application servers receive the input in the compute clusters and share application state and data using Azure Redis Cache or remote direct memory access (RDMA). The application servers host various COBOL or PL/1 application programs. A transaction system manager is an emulator on Azure that can handle customer information control systems (CICS) and information management systems (IMS) workloads. A batch system emulator on Azure does the role of job control language (JCL).
5. You can use Azure services or other partner software hosted in VMs for system, utilities, and data management.
6. Mainframe data is migrated to Azure databases. Azure provides various efficient data storage services like Azure SQL Database, SQL Server on Azure Virtual Machines, and Azure SQL Managed Instance. There are many factors to consider when making a choice, like type of workload, cross-database queries, and two-phase commit requirements. Azure data services provide scalable and highly available data storage that you can share across multiple compute resources in a cluster. You can make these services geo-redundant, and then configure them so that if failover occurs, the disaster recovery database instance becomes active.
7. AKS enables you to scale out and scale down your mainframe modernization workloads in Azure, to take advantage of the cloud platform. When you deploy multiple AKS clusters, choose regions where AKS is available. Then you can use paired regions for a highly resilient architecture. It's important to run multiple instances of an AKS cluster across multiple regions and in highly available configurations.

8. Azure Data Factory provides data ingestion and synchronization with multiple data sources, both within Azure and from external sources. Azure Blob Storage is a common landing zone for external data sources.
9. Use Azure Site Recovery for disaster recovery of the VM and container cluster components. Azure Site Recovery replicates and syncs the production environment to the failover region.

## Components

- [Virtual Machines](#) : Virtual Machines is an on-demand, scalable computing resource. An Azure VM gives you the flexibility of virtualization without having to buy and maintain the physical hardware that runs it.
- [Azure Virtual Network](#) : Virtual Network is the fundamental building block for your private network in Azure. Virtual Network enables many types of Azure resources, like Virtual Machines, to securely communicate with each other, the internet, and on-premises networks. Virtual Network is like a traditional network that you operate in your own data center. However, it brings with it the benefits of Azure's infrastructure such as scale, availability, and isolation.
- [Azure Virtual Network Interface Cards](#): A network interface enables an Azure VM to communicate with internet, Azure, and on-premises resources. As shown in this architecture, you can add more network interface cards to the same Azure VM. This way, the Solaris child-VMs have their own dedicated network interface device and IP address.
- [Azure Disk Storage](#) : Managed disks are block-level storage volumes that are managed by Azure and used with Azure VMs. The available types of disks are Azure Ultra Disk Storage, Azure Premium SSD, Azure Standard SSD, and Azure Standard HDD. For this architecture, we recommend either Premium SSD or Ultra Disk Storage.
- [Azure Files](#) : Azure Files offers fully managed file shares in the cloud that are accessible via the industry standard Server Message Block (SMB) protocol. You can mount Azure file shares concurrently by cloud or on-premises deployments of Windows, Linux, and macOS.
- [Azure ExpressRoute](#) : With ExpressRoute, you can extend your on-premises networks into the Microsoft cloud over a private connection facilitated by a connectivity provider. You can also establish connections to Microsoft cloud services, like Microsoft Azure and Microsoft 365.

- [AKS](#) : Deploy and manage containerized applications more easily with a fully managed Kubernetes service. Azure Kubernetes Service (AKS) offers serverless Kubernetes, an integrated continuous integration and continuous delivery (CI/CD) experience, and enterprise-grade security and governance. Unite your development and operations teams on a single platform to rapidly build, deliver, and scale applications with confidence.
- [Azure Container Registry](#) : Build, store, secure, scan, replicate, and manage container images and artifacts with a fully managed, geo-replicated instance of OCI distribution. Connect across environments like AKS and Azure Red Hat OpenShift, and across Azure services like App Service, Machine Learning, and Batch.
- [Site Recovery](#) : Site Recovery offers ease of deployment, cost effectiveness, and dependability. Deploy replication, failover, and recovery processes through Site Recovery to help keep your applications running during planned and unplanned outages.

## Considerations

These considerations implement the pillars of the Azure Well-Architected Framework, which is a set of guiding tenets that can be used to improve the quality of a workload. For more information, see [Microsoft Azure Well-Architected Framework](#).

## Reliability

Reliability ensures your application can meet the commitments you make to your customers. For more information, see [Overview of the reliability pillar](#).

- To make the most of Azure's capabilities, use a container-based approach to deployment. This approach helps if the application needs to scale on demand and achieve elastic provisioning of capacity without the need to manage the infrastructure. It also enables you to add event-driven autoscaling and triggers. A container bundles all the software that's needed for execution into one executable package. It includes an application's code together with the related configuration files, libraries, and dependencies necessary to run the app.
- You need to orchestrate and manage containerized services and their associated networking and storage components. AKS is an excellent option because it automates cluster and resource management. You designate the number of nodes you need, and AKS fits your containers onto the right nodes to make the best use of resources. AKS also supports automated rollouts and rollbacks, service

discovery, load balancing, and storage orchestration. And AKS supports self-healing. If a container fails, AKS starts a new one. You can also safely store secrets and configuration settings outside of the containers.

- The architecture uses Site Recovery to mirror Azure VMs to a secondary Azure region for quick failover and disaster recovery if an Azure datacenter fails.
- To maximize the uptime of the workloads on the AKS deployment approach, it's a best practice for business continuity to deploy the application into multiple AKS clusters across different regions. Your application state can be available across multiple clusters because AKS allows storage replication across multiple regions.
- To maximize the uptime of the workloads on a VM-based deployment approach, consider using Azure Virtual Machine Scale Sets. With Virtual Machine Scale Sets, you can create and manage a group of load balanced VMs. The number of VM instances can automatically increase or decrease in response to demand or a defined schedule. Scale sets provide high availability to your applications, and allow you to centrally manage, configure, and update many VMs.

## Security

Security provides assurances against deliberate attacks and the abuse of your valuable data and systems. For more information, see [Overview of the security pillar](#).

- This solution uses an Azure network security group to manage traffic between Azure resources. For more information, see [Network security groups](#).
- [Azure Bastion](#) maximizes admin access security by minimizing open ports. Bastion provides secure and seamless RDP/SSH connectivity to virtual network VMs directly from the Azure portal over TLS.

## Cost optimization

To help optimize costs, look for ways to reduce unnecessary expenses and improve operational efficiencies. For more information, see [Overview of the cost optimization pillar](#).

Azure provides cost optimization by running on Windows VMs. With Windows VMs, you can turn off the VMs when not in use and script a schedule for known usage patterns. Azure identifies the right number or resource types, analyzes spending over time, and scales to meet business needs without overspending.

Use the [Azure pricing calculator](#) to estimate the cost of the services in this architecture.

# Operational excellence

Operational excellence covers the operations processes that deploy an application and keep it running in production. For more information, see [Overview of the operational excellence pillar](#).

- The target architecture is functional with Azure Cloud Services.
- The container-based deployment promotes adoption of DevOps and Agile working principles.
- You have full flexibility of development and production deployment options.

# Performance efficiency

Performance efficiency is the ability of your workload to scale to meet the demands placed on it by users in an efficient manner. For more information, see [Performance efficiency pillar overview](#).

- Performance efficiency is built into this solution because of the load balancers. If one presentation or transaction server fails, the server behind the load balancer shoulders the workload.
- Kubernetes provides a cluster autoscaler. The autoscaler adjusts the number of nodes based on the requested compute resources in the node pool.

# Contributors

*This article is maintained by Microsoft. It was originally written by the following contributors.*

Principal author:

- [Sunnyma Ghosh](#) | Senior program manager

Other contributor:

- [Bhaskar Bandam](#) | Senior program manager

# Next steps

- [Azure white papers about mainframe topics](#)
- [Mainframe rehosting on Azure virtual machines](#)
- [Mainframe workloads supported on Azure](#)

For more information, contact [legacy2azure@microsoft.com](mailto:legacy2azure@microsoft.com).

## Related resources

- [Azure mainframe and midrange architecture concepts and patterns](#)
- [Mainframe and midrange data replication to Azure using Qlik](#)
- [Mainframe modernization using Model9](#)
- [Rehost mainframe applications by using NTT DATA UniKix](#)

# Rehost Adabas & Natural applications in Azure

Azure Virtual Network

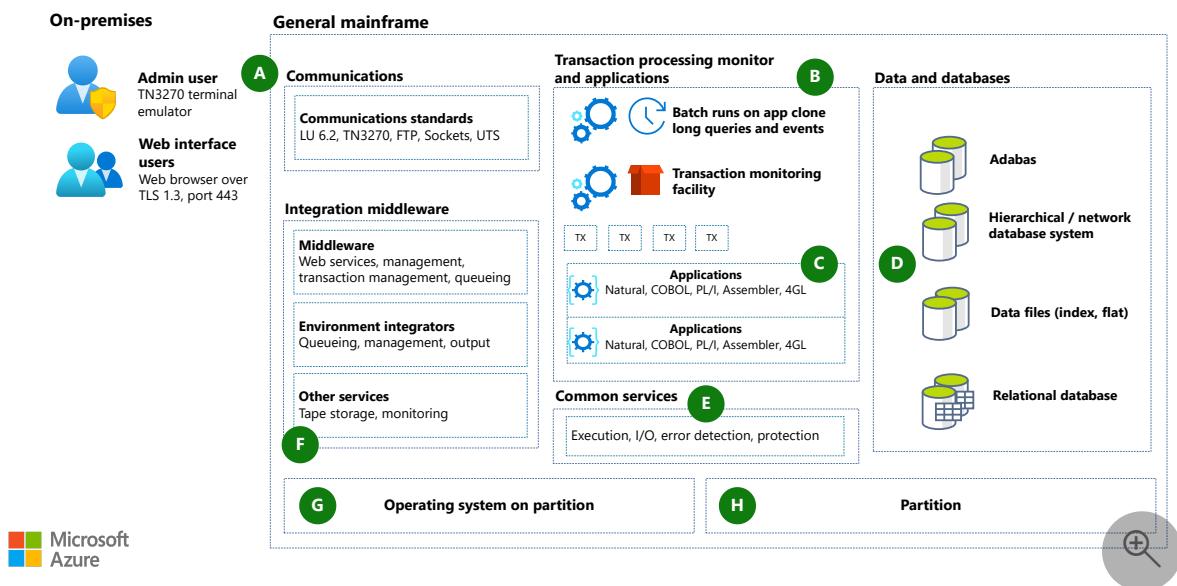
Azure Virtual Machines

Azure ExpressRoute

For decades, Software AG Adabas has been the adaptable database system behind many large mission-critical business applications. Now you can bring the convenience of cloud computing to these applications without giving up your Adabas database or the Natural programming language. This architecture presents the option to rehost your system on Azure. It provides a high-level look at what's possible, whether you keep the green screen or go modern.

## Mainframe architecture

This architecture shows a legacy Adabas & Natural architecture, before a rehost to the cloud:



Download a [Visio file](#) of this architecture.

## Workflow

A. Users input data over TCP/IP, including TN3270 and HTTP(S). Data is input into the mainframe via standard mainframe protocols.

B. Applications receive the data. These applications can be either batch or online systems.

C. Natural, COBOL, PL/I, Assembler, or compatible languages run in an enabled environment.

D. Database services, commonly hierarchical/network database systems and relational databases, store data.

E. Common services, like program execution, I/O operations, error detection, and protection within the environment, provide support.

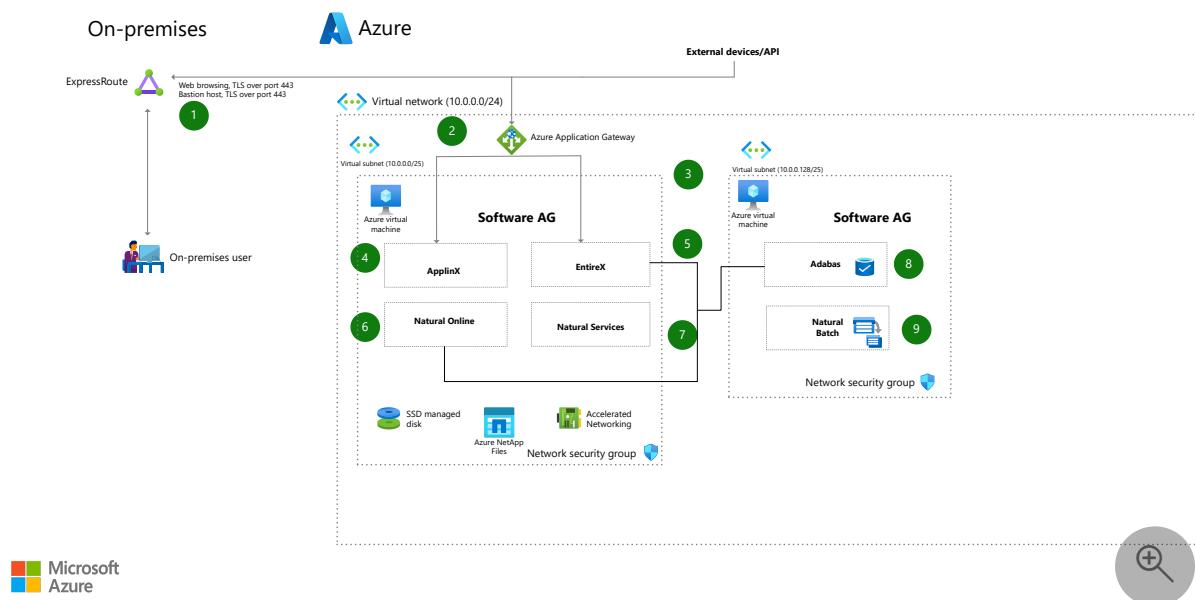
F. Middleware and utility services manage functions like tape storage, queueing, output, and web services within the environment.

G. Operating systems run on partitions.

H. Partitions are used to run separate workloads or segregate work types within the environment.

## Azure architecture

This diagram shows the legacy architecture migrated to Azure. A rehost approach is used to migrate the system:



Download a [Visio file](#) of this architecture.

## Workflow

1. Data is input, typically via either Azure ExpressRoute from remote clients or via other applications currently running in Azure. In either case, TCP/IP connections provide the primary means of connection to the system. User access for web-based applications is provided over TLS port 443. You can use the legacy web-application presentation layer virtually unchanged to minimize user retraining. Alternatively, you can update the web-application presentation layer with modern UX frameworks. To improve security by minimizing open ports, you can use Azure Bastion hosts for admin access to the VMs.
2. Azure Application Gateway is used to access the application compute clusters. It provides Layer 7 load balancing services. It can also make routing decisions based on additional attributes in an HTTP request, like a URI path or host headers. For example, you can route traffic based on the incoming URL. In this case, you route traffic to the correct Software AG component (ApplinX or EntireX).
3. For application compute clusters, you can use one VM for the Adabas & Natural software. We recommend that you use separate VMs for the application and database for more than 200 MIPS. This example uses two VMs. You can deploy a distributed architecture (Adabas & Natural running on multiple VMs) to provide scalable Natural applications with higher availability and higher consistency for Adabas storage.
4. ApplinX provides web connectivity and integration into system applications. No changes to the applications are required.
5. EntireX connects services that run on Integration Server to mission-critical programs that are written in languages like COBOL or Natural.
6. Online users connect to the Natural application by using Natural Online. Natural Online enables connection via SSH or a web browser.
7. Natural Services provides API access to business functions that are programmed in Natural.
8. An Adabas NoSQL database stores data.
9. Software AG Natural Batch runs batch jobs.

## Components

- [Azure Virtual Machines](#) . Virtual Machines is one of several types of on-demand, scalable computing resources that Azure offers. An Azure virtual machine (VM) provides the flexibility of virtualization without the need to buy and maintain physical hardware.
- [Azure Virtual Network](#) . Virtual Network is the fundamental building block for your private network on Azure. Virtual Network enables many types of Azure resources, like VMs, to communicate with each other, the internet, and on-premises networks via a highly secure connection. A virtual network is like a

traditional network that you might operate in your own datacenter, but it provides the benefits of the Azure infrastructure, like scalability, availability, and isolation.

- [Azure Application Gateway](#). Application Gateway provides a customizable Layer 7 load-balancing solution.
- [Virtual network interfaces](#). A network interface enables a VM to communicate with internet, Azure, and on-premises resources. You can add network interface cards to a VM to provide child VMs with their own dedicated network interface device and IP address.
- [Azure managed disks](#). Azure managed disks are block-level storage volumes that are managed by Azure and used with Azure Virtual Machines. Ultra disks, premium solid-state drives (SSD), standard SSDs, and standard hard disk drives (HDD) are available. For this architecture, we recommend either premium SSDs or ultra disk SSDs.
- [Azure ExpressRoute](#). You can use ExpressRoute to extend your on-premises networks into the Azure cloud via a private connection that's facilitated by a connectivity provider. By using ExpressRoute, you can establish connections to Microsoft cloud services like Azure and Office 365.

## Scenario details

For decades, Software AG Adabas has been the adaptable database system behind many large mission-critical business applications. Now you can bring the convenience of cloud computing to these applications without giving up your Adabas database, the Natural programming language, or even your green screen, unless you want to.

Most organizations are pragmatic in their approach to digital transformation. They want to reuse what they can and make cost-effective choices about the rest. That's why the rehost approach to cloud migration is so popular. You simply move your workload as is, if possible, to Azure virtual machines (VMs), a type of infrastructure as a service (IaaS). VMs run in Azure datacenters that are managed by Microsoft, so you benefit from the efficiency, scalability, and performance of a distributed platform without the overhead of hardware management.

This architecture presents the rehost option. It provides a high-level look at what's possible, whether you keep the green screen or go modern.

## Potential use cases

This architecture is appropriate for organizations that want to use a *rehost* approach for a cost-effective mainframe migration to Azure that optimizes reuse of legacy systems.

To gain the full benefits of cloud computing, consider a [refactor](#) approach that uses modern techniques like container-based microservices. This type of migration is more complex than a rehost approach, but the payoff is increased flexibility and scalability.

## Considerations

The following considerations implement the pillars of the Azure Well-Architected Framework, a set of guiding tenets that you can use to improve the quality of your workloads. For more information, see [Microsoft Azure Well-Architected Framework](#).

### Cost optimization

Azure helps you avoid unnecessary costs by identifying the correct number of resources, analyzing spending over time, and scaling to meet business needs without overspending.

Azure also provides cost optimization by running on VMs. You can turn off the VMs when they're not being used and script a schedule for known usage patterns. For more information about cost optimization for [VM instances](#), see the [Azure Well-Architected Framework](#).

The VMs in this architecture use either premium SSDs or ultra disk SSDs. For more information about disk options and pricing, see [Managed Disks pricing](#).

### Operational excellence

In addition to supporting faster cloud adoption, rehosting also promotes the adoption of DevOps and Agile working principles. It provides flexibility in development and production deployment options.

### Performance efficiency

Load balancers and redundant VMs in a distributed environment provide performance efficiency and resiliency in this architecture. If one presentation or transaction server fails, the other server behind the load balancer handles the workload.

### Security

This solution uses an Azure network security group (NSG) to manage traffic between Azure resources in different subnets. For more information, see [Network security groups](#).

Azure Bastion improves security for admin access by minimizing open ports. Azure Bastion provides highly secure RDP or SSH connectivity to virtual network VMs directly from the Azure portal, over TLS.

## Contributors

*This article is maintained by Microsoft. It was originally written by the following contributors.*

Principal author:

- Marlon Johnson | Senior Program Manager

Other contributors:

- [Mick Alberts](#) | Technical Writer
- [Bhaskar Bandam](#) | Senior Program Manager

## Next steps

For more information, contact [legacy2azure@microsoft.com](mailto:legacy2azure@microsoft.com).

See these additional resources:

- [What is Azure Virtual Network?](#)
- [Configure virtual networks](#)
- [What is Azure ExpressRoute?](#)
- [What is Azure Application Gateway?](#)
- [Windows virtual machines in Azure](#)
- [Mainframe rehosting on Azure virtual machines](#)

## Related resources

- [Refactor mainframe computer systems that run Adabas & Natural](#)
- [Azure mainframe and midrange architecture concepts and patterns](#)
- [Mainframe migration overview](#)
- [Move mainframe compute to Azure](#)
- [General mainframe refactor to Azure](#)
- [AIX UNIX on-premises to Azure Linux migration](#)

# Rehost IMS workloads to virtual machines by using IMSql

Azure Virtual Machines

Azure Virtual Network

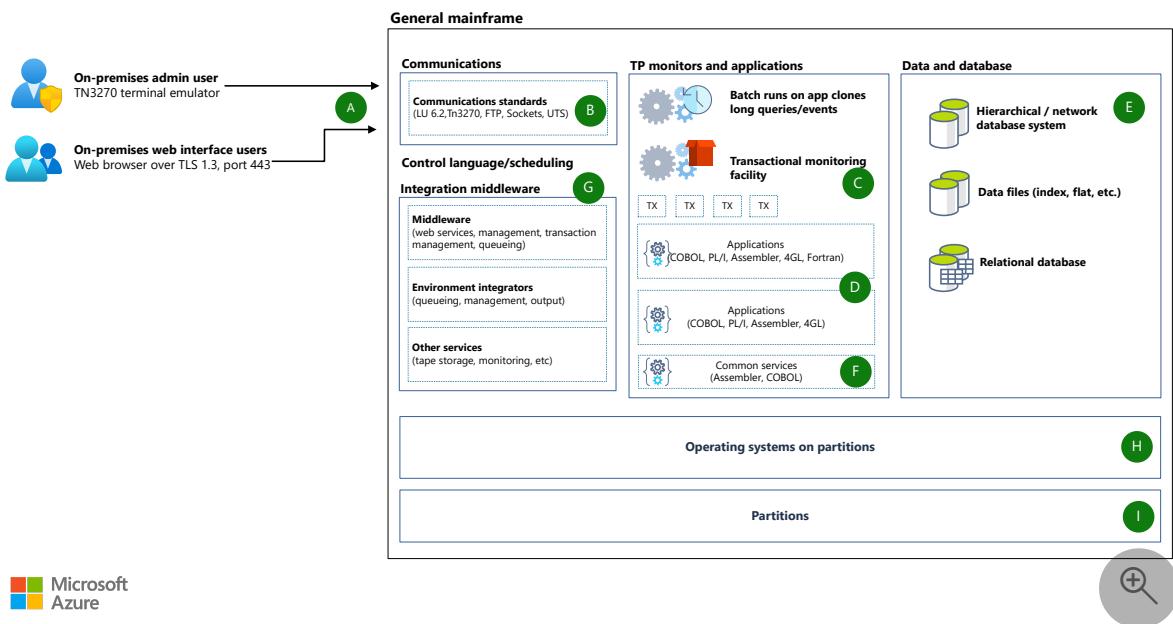
Azure Virtual Machine Scale Sets

Azure SQL Managed Instance

This architecture shows how to use Raincode's IMSql to rehost IMS Database Manager (IMS DB) and IMS Transaction Manager (IMS TM) systems on .NET and SQL Server in the simplest way: by using virtual machines. You can recompile legacy applications to target .NET and interact with IMSql in the same way that they interact with IMS on a mainframe. IMSql transitions mainframe applications to an Azure-native architecture while thoroughly preserving the business logic.

## Architecture

### IBM z/OS architecture, before migration



Download a [Visio file](#) of this architecture.

## Dataflow

- A. Users connect via TCP/IP by using protocols like TN3270 and HTTPS.
- B. Input into the mainframe uses standard mainframe communication protocols.

C. Applications receive the data. These applications are either batch or online systems.

D. COBOL, PL/I, Assembler, or other compatible languages run in an enabled environment.

E. Database systems, commonly hierarchical/network and relational systems, store data.

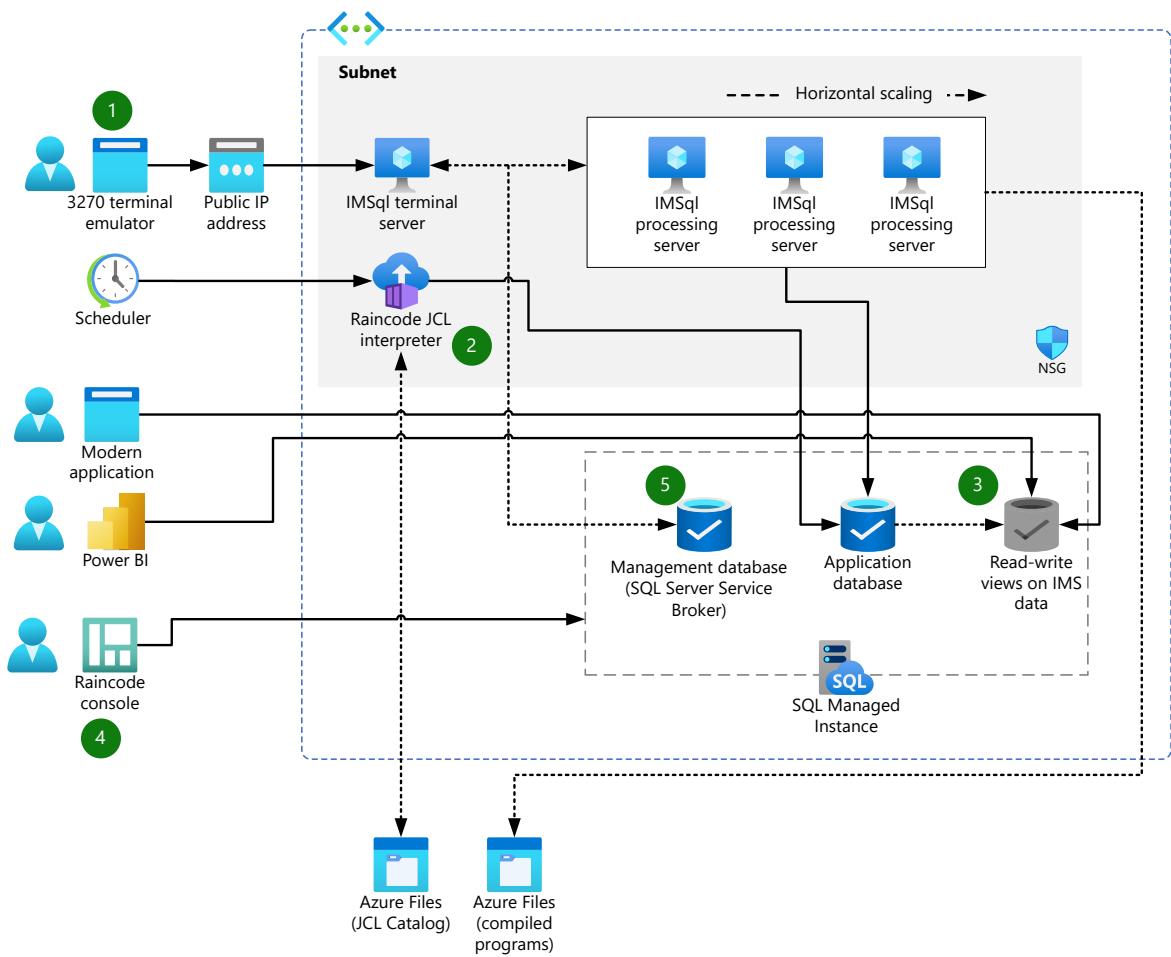
F. Common services, like program execution, I/O operations, error detection, and protection within the environment, provide support.

G. Middleware and utilities manage services like tape storage, queueing, output, and web services within the environment.

H. Operating systems run on partitions.

I. Partitions run separate workloads and segregate work types within the environment.

## Azure architecture, after migration



Download a [Visio file](#) of this architecture.

## Dataflow

1. IBM 3270 terminal emulators connect to IMS TM applications that are deployed on Azure unchanged via the IMSql Terminal Server.
2. Batch processes written in JCL are run unchanged via transient Azure container instances that run the Raincode JCL interpreter. Compiled legacy programs access IMS DB by using standard IMS APIs. Raincode JCL can store its catalog on any file-based storage.
3. Read/write SQL Server views on the IMS data enable modern applications or business intelligence (like Power BI) to communicate directly with IMS applications, abstracting away mainframe elements like data structures and character encodings.
4. Raincode Console provides a web-based administration environment for IMSql.
5. SQL Server Service Broker is the communications backbone for IMSql components.

## Components

- [Azure Virtual Network](#) is the fundamental building block for your private network in Azure. Virtual Network enables many types of Azure resources, like virtual machines (VMs), to communicate with each other, the internet, and on-premises networks, all with improved security. Virtual Network is like a traditional network that you operate in your own datacenter, but it provides more of the benefits of the Azure infrastructure, like scale, availability, and isolation.
- [Azure Virtual Machine Scale Sets](#) provides automated and load-balanced VM scaling that simplifies the management of your applications and increases availability.
- [Azure SQL Managed Instance](#), part of the Azure SQL service portfolio, is a managed, highly secure, always up-to-date SQL instance in the cloud.

## Alternatives

- You can use SQL Server in an Azure virtual machine as an alternative to SQL Managed Instance. We recommend SQL Managed Instance in this architecture because of benefits like high availability, seamless integration with various Azure services, and management of underlying security patches and maintenance.
- You can use an Azure single-VM architecture as an alternative to Virtual Machine Scale Sets. You might want to use single VMs for workloads that have constant load and performance demands and don't need scaling. This architecture uses Virtual Machine Scale Sets to handle typical IMS workloads.

# Scenario details

This architecture shows how to seamlessly rehost to Azure a mainframe workload that has critical IMS features and capabilities. You don't need to translate or modify your existing application. The architecture uses IMSql and Azure SQL.

- Raincode compilers generate 100 percent thread-safe managed code for .NET. The .NET assemblies are loaded dynamically and called by IMSql processing servers.
- IMSql is intrinsically non-transformational. It keeps the source (COBOL, PL/I) as is. The IMS-specific CBLTDLI and PLITDLI calls and EXEC DLI statements aren't changed. This capability ensures optimal maintainability of the resulting system. It extends to IMS DB data: the data is imported as is, in bulk, with no changes, cleansing, or normalization.
- IMSql uses the robust, versatile, and scalable SQL Server as a database, transaction processor, and execution platform.
- IMSql operates in three modes:
  - Online
  - Batch
  - Load and Unload (for data migration or for JCLs that produce or consume sequential files)
- On mainframes, Database Descriptions (DBDs) and Program Specification Blocks (PSBs) are compiled to create the database and the program's description. Similarly, on IMSql, DBDs and PSBs are compiled into an XML representation. This representation enables IMS-aware programs to determine which database segments pertain to them. It also drives the generation of various server-side artifacts for IMSql, like the database schema and stored procedures.

## Potential use cases

- Modernize infrastructure and eliminate the high costs, limitations, and rigidity associated with IMS, or, more generally, with mainframes.
- Reduce technical debt by implementing cloud-native solutions and supporting a DevOps strategy.
- Move IMS workloads to the cloud without the side effects of a complete redevelopment.
- Move IMS business-critical applications while maintaining continuity with other on-premises applications.

## Considerations

These considerations implement the pillars of the Azure Well-Architected Framework, a set of guiding tenets that you can use to improve the quality of a workload. For more information, see [Microsoft Azure Well-Architected Framework](#).

## Reliability

Reliability ensures your application can meet the commitments you make to your customers. For more information, see [Overview of the reliability pillar](#).

- This OLTP architecture can be deployed in multiple regions and can incorporate a geo-replication data layer.
- The Azure database services support zone redundancy and can fail over to a secondary node during outages or to enable maintenance activities.

## Security

Security provides assurances against deliberate attacks and the abuse of your valuable data and systems. For more information, see [Overview of the security pillar](#).

This solution uses an Azure network security group to manage traffic to and from Azure resources. For more information, see [Network security groups](#).

These security options are available in Azure database services:

- Data encryption at rest
- Dynamic data masking
- Always Encrypted data

For general guidance on designing highly secure SQL solutions, see [Azure security recommendations](#).

## Cost optimization

Cost optimization is about reducing unnecessary expenses and improving operational efficiencies. For more information, see [Overview of the cost optimization pillar](#).

- Azure provides cost optimization by running on Windows VMs. You can turn off the VMs when they're not being used and script a schedule for known usage patterns. Azure helps you avoid unnecessary costs by identifying the right number of resource types, analyzing spending over time, and scaling to meet business needs without overspending.

- SQL Managed Instance provides various pricing tiers, like general purpose and business critical, to optimize costs based on usage and business criticality.
- Use [Azure Reservations](#) and [Azure savings plan for compute](#) with a one-year or three-year contract and receive significant savings off pay-as-you-go prices.

Use the [Azure pricing calculator](#) to estimate the cost of implementing this solution.

## Contributors

*This article is maintained by Microsoft. It was originally written by the following contributors.*

Principal author:

- [Bhaskar Bandam](#) | Senior Program Manager

Other contributor:

- [Mick Alberts](#) | Technical Writer

*To see non-public LinkedIn profiles, sign in to LinkedIn.*

## Next steps

- [IMSsql user guide](#)
- [What is Azure Virtual Network?](#)

For more information, contact [legacy2azure@microsoft.com](mailto:legacy2azure@microsoft.com).

## Related resources

See the companion architecture:

- [Rehost IMS DC and IMS DB on Azure by using IMSsql](#)

More related resources:

- [General mainframe refactor to Azure](#)
- [Mainframe access to Azure databases](#)
- [Re-engineer mainframe batch applications on Azure](#)
- [Rehost a general mainframe on Azure](#)

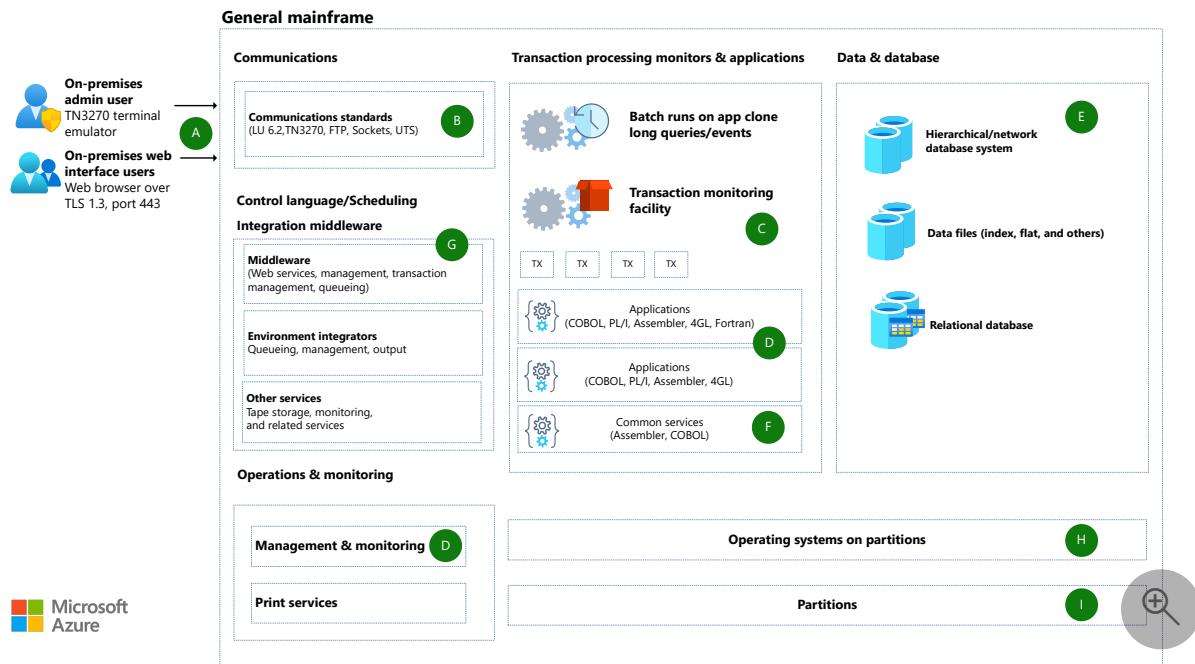
# Rehost mainframe applications by using NTT DATA UniKix

Azure ExpressRoute   Azure Load Balancer   Azure Site Recovery   Azure SQL Database   Azure Storage

UniKix is a mainframe-rehosting software suite from NTT DATA. This suite provides a way to run migrated legacy assets on Azure. Example assets include IBM CICS transactions, IBM IMS applications, batch workloads, and JCL workloads. This article outlines a solution for rehosting mainframe applications on Azure. Besides UniKix, the solution's core components include Azure ExpressRoute, Azure Site Recovery, and Azure storage and database services.

## Mainframe architecture

The following diagram shows a legacy mainframe system before it's rehosted to the cloud:



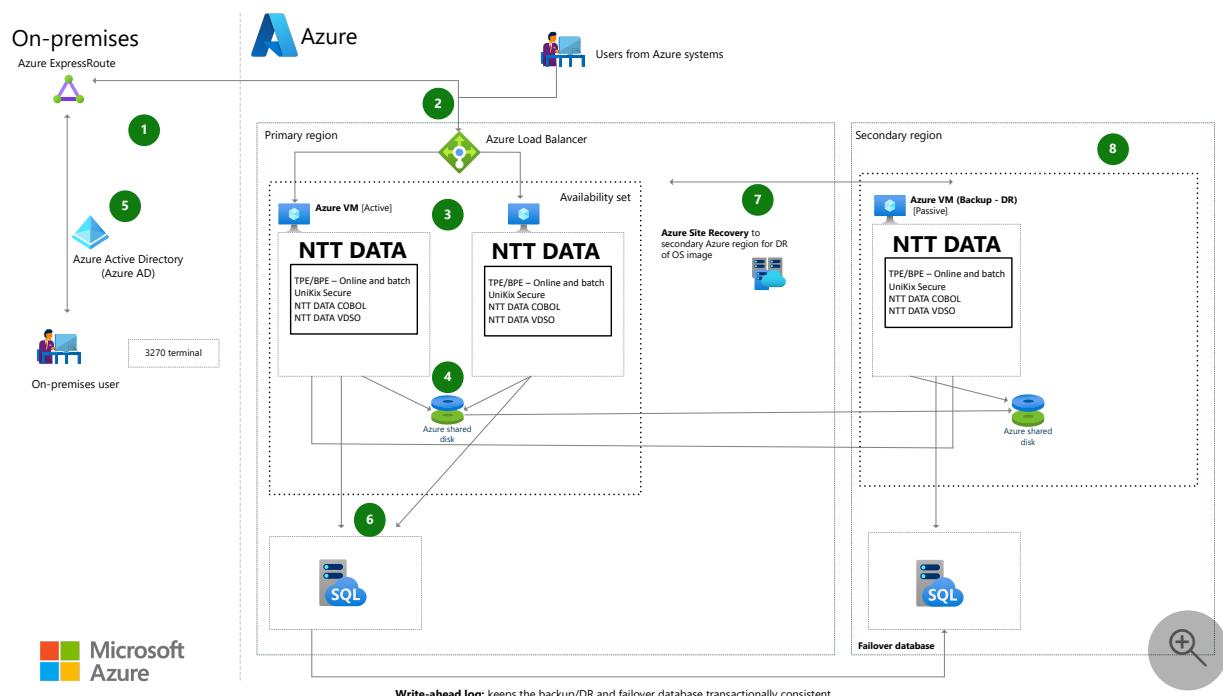
Download a [Visio file](#) of this architecture.

## Workflow

- On-premises users interact with the mainframe by using TCP/IP (A):
  - Admin users interact through a TN3270 terminal emulator.

- Web interface users interact via a web browser over TLS 1.3 port 443.
- Mainframes use communication protocols such as LU 6.2, TN3270, FTP, Sockets, and UTS to receive input (B).
- Batch and online applications process the input (C).
- Mainframe applications are in COBOL, PL/I, Assembler, 4GL, and Fortran. These languages and compatible ones run in an enabled environment (D).
- Mainframes use hierarchical, network, and relational databases (E).
- Services perform tasks for the applications. Services that typically run include program execution, I/O operations, error detection, and protection. (F).
- Middleware and utility services manage tasks like tape storage, queueing, output, and web support (G).
- Operating systems provide an interface between the engine and the software that it runs (H).
- Partitions run separate workloads or segregate work types within the environment (I).

## Azure architecture



Download a [Visio file](#) of this architecture.

# Workflow

1. ExpressRoute connects an on-premises corporate network to NTT DATA's UniKix mainframe rehosting software suite. Traffic from users and external interfaces that aren't on the Azure platform flows through this ExpressRoute connection to the Azure instances.
2. Azure Load Balancer distributes online transactions across two or more Azure virtual machines (VMs). Port 4444 is used to connect with x3270. For a single-host alternative, see [Alternatives](#).
3. The application server runs the following NTT DATA products:
  - TPE. This environment runs:
    - Rehosted online IBM CICS transactions.
    - IBM IMS/TM applications.
    - Transformed IDMS DC programs.
    - Related resources.
  - These workloads run on industry-standard servers and operating systems such as Red Hat Linux.
  - BPE. This environment provides a complete job entry subsystem (JES) environment for the administration, execution, and management of batch workloads.
  - UniKix Secure, which was previously known as Transaction Security Facility (TSF). This external security manager provides role-based access control that's based on security for online TPE-based transactions.
  - NTT DATA COBOL. This technology produces optimized, portable object code that you can deploy in Azure. NTT DATA COBOL supports ANSI-85 standard and legacy COBOL dialects.
  - NTT DATA VDSO. This mechanism provides a way to store VSAM key-sequenced dataset (KSDS) data in a SQL database rather than local disk files. NTT DATA VDSO supports many database technologies such as SQL Server, DB2, Oracle, and MySQL.
4. Azure managed disks provide storage for shared files.
5. UniKix Secure uses Microsoft Entra ID to provide authentication. This security manager replaces security systems like Resource Access Control Facility (RACF), Access Control Facility 2 (ACF2), and Top Secret.

6. The solution stores database tables and, optionally, VSAM files, in Azure SQL Database. This data is replicated to another Azure region for disaster recovery purposes.
7. Site Recovery replicates the Azure production application VMs. This replication helps ensure business continuity by keeping business apps and workloads running during outages.
8. The second Azure region mirrors the configuration of the primary Azure region for disaster recovery.

## Components

- [ExpressRoute](#) extends on-premises networks into Azure over a private, dedicated fiber connection from a connectivity provider. ExpressRoute establishes connections to Microsoft cloud services like Azure and Microsoft 365.
- [Load Balancer](#) distributes incoming traffic to compute resource clusters. You can define rules and other criteria to distribute the traffic.
- [Azure Virtual Machines](#) offers many sizes and types of on-demand, scalable computing resources. With Azure VMs, you get the flexibility of virtualization without having to buy and maintain physical hardware.
- [Azure Storage](#) offers scalable, secure cloud storage for all your data, applications, and workloads:
  - [Azure Disk Storage](#) is high-performance, durable block storage for business-critical applications. Azure managed disks are block-level storage volumes that are managed by Azure on Azure VMs. The available types of disks are Ultra Disk Storage, Premium SSD, Standard SSD, and Standard HDD. This solution uses either Premium SSD or Ultra Disk Storage.
  - [Azure Files](#) offers fully managed file shares in the cloud that are accessible via the industry standard Server Message Block (SMB) protocol. Cloud and on-premises Windows, Linux, and macOS deployments can mount Azure Files file shares concurrently.
  - [Azure Blob Storage](#) provides scalable and secure object storage. It can manage large amounts of unstructured data, such as archives and data lakes. Blob Storage is a good fit for high-performance computing, machine learning, and cloud-native workloads.
- [Azure databases](#) offer a choice of fully managed relational and NoSQL databases to fit modern application needs. Automated infrastructure management provides scalability, availability, and security.

- [SQL Database](#) is a fully managed platform as a service (PaaS) database engine. SQL Database runs on the latest stable version of SQL Server and a patched operating system. Automated functionality includes upgrading, patching, backups, and monitoring. Because SQL Database offers built-in PaaS capabilities, you can focus on domain-specific, business-critical database administration and optimization.
- [Site Recovery](#) mirrors Azure VMs to a secondary Azure region. If the primary datacenter fails, the secondary region provides quick failover and disaster recovery.

## Alternatives

- Sometimes scaling isn't possible, due to licensing constraints or your application's design. In those cases, you can mirror the mainframe setup with a single host.
- For disaster recovery, the solution replicates the SQL Server data to another region. As another option, you can use the Always On availability groups feature of SQL Server as a disaster recovery solution.
- In some scenarios, some of the solution's components and workflows are optional or interchangeable.

## Scenario details

UniKix is a mainframe-rehosting software suite from NTT DATA. This suite provides a way to run migrated legacy assets on Azure. Example assets include IBM CICS transactions, IBM IMS applications, batch workloads, and JCL workloads.

The NTT DATA software offers many useful features:

- A means for converting Integrated Database Management System (IDMS), Natural, and other application environments so that they operate within UniKix
- A robust, logically threaded NTT DATA engine that provides a rich online transaction processing environment (TPE)
- A complete, native batch processing environment (BPE)
- A powerful COBOL compiler
- A streamlined runtime environment
- A graphical source-level debugger
- A portable indexed file system

By using UniKix to rehost mainframe applications, you can take advantage of these features. You can also:

- Avoid licensing fees for mainframe software.

- Reduce infrastructure maintenance and operating costs.
- Minimize risk and disruption by retaining existing user interfaces and business logic.
- Modernize your IT environment.
- Capitalize on Azure solutions for scalability, high availability, and disaster recovery.
- Implement a modern DevOps workflow with NTT DATA tools and select Azure tools.

This article outlines a solution for rehosting mainframe applications on Azure. Besides UniKix, the solution's core components include Azure ExpressRoute, Azure Site Recovery, and Azure storage and database services.

## Potential use cases

Industries that use mainframes can benefit from UniKix rehosting solutions. The following sectors that process large volumes of transactions on a daily basis are possibilities:

- Banking and finance
- Insurance
- Healthcare
- The military and government
- E-commerce and retail

## Considerations

The following considerations, based on the [Azure Well-Architected Framework](#), apply to this solution:

### Reliability

The solution uses Site Recovery to mirror Azure VMs to a secondary Azure region. If the primary datacenter fails, the secondary region provides quick failover and disaster recovery.

### Security

This solution uses an Azure network security group to manage traffic between Azure resources. For more information, see [Network security groups](#).

# Cost optimization

- Azure provides cost optimization by running on VMs. You can turn off the VMs when not in use, and script a schedule for known usage patterns. For more information about cost optimization for [VM instances](#), see the [Azure Well-Architected Framework](#).
- For managed disks, the VMs in this solution use either Premium SSD or Ultra Disk Storage. For more information about disk options and pricing, see [Managed disks pricing](#).
- To estimate the cost of implementing this solution, use the [Pricing calculator](#).

## Contributors

*This article is maintained by Microsoft. It was originally written by the following contributors.*

Principal author:

- [Richard Berry](#) | Senior Program Manager

Other contributors:

- [Bhaskar Bandam](#) | Senior Program Manager
- [Jonathon Frost](#) | Principal Program Manager

## Next steps

- For more information about rehosting on Azure, contact [legacy2azure@microsoft.com](mailto:legacy2azure@microsoft.com).
- For more information about using NTT DATA software for rehosting, contact [NTTReHost.Cloud@nttdata.com](mailto:NTTReHost.Cloud@nttdata.com).
- To see how to deploy UniKix in Azure, see these resources:
  - [Deploying NTT DATA UniKix in Azure, Part 1](#)
  - [Deploying NTT DATA UniKix in Azure, Part 2](#)
- To learn more about components in the solution, see these articles:
  - [Azure ExpressRoute](#)
  - [What is Azure Load Balancer?](#)
  - [Introduction to Azure managed disks](#)

- [What is Azure SQL Database?](#)
- [About Site Recovery](#)

## Related resources

- [Azure mainframe and midrange architecture concepts and patterns](#)
- [Mainframe migration overview](#)
- [Mainframe rehosting on Azure virtual machines](#)
- [Move mainframe compute to Azure](#)
- [General mainframe refactor to Azure](#)
- [AIX UNIX on-premises to Azure Linux migration](#)

# Unisys ClearPath Forward OS 2200 enterprise server virtualization on Azure

Azure Virtual Machines

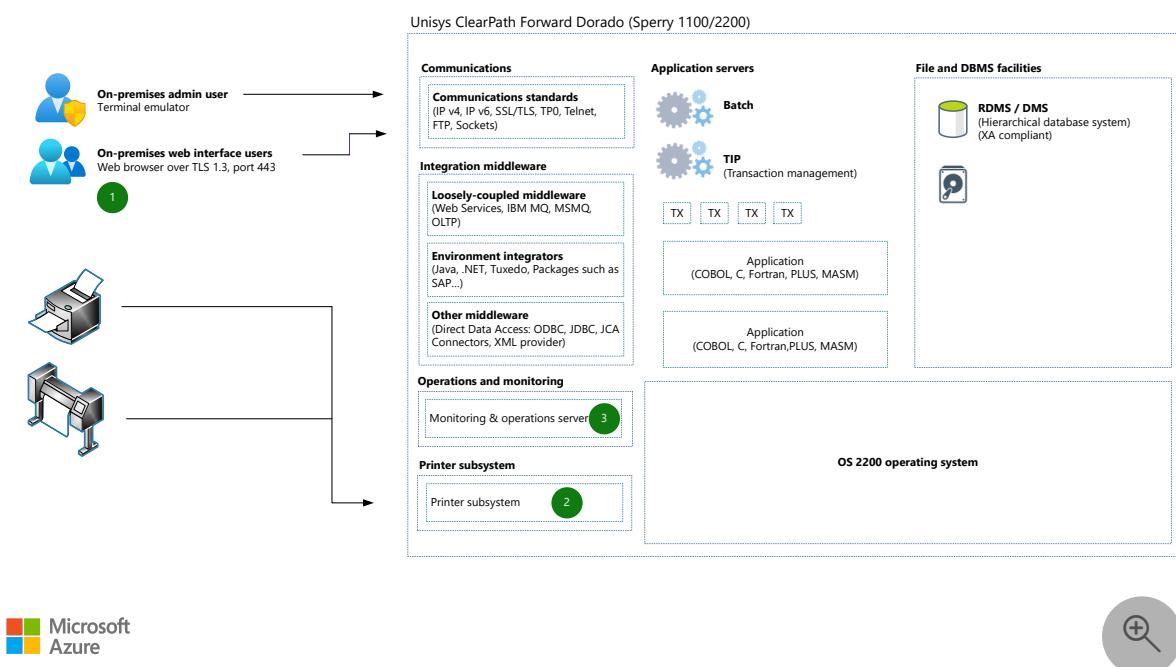
Azure Virtual Network

Azure ExpressRoute

This article describes how to use virtualization technologies from Unisys, a Microsoft partner, with an existing Unisys ClearPath Forward (CPF) Dorado enterprise server. With this approach, you can accelerate your move into Azure and eliminate the need to rewrite the application code or redesign the database architecture. Existing code is maintained in its original form. The application screens, user interactions, and data structures behind the scenes are unchanged, which eliminates the need to retrain your users.

## Architecture

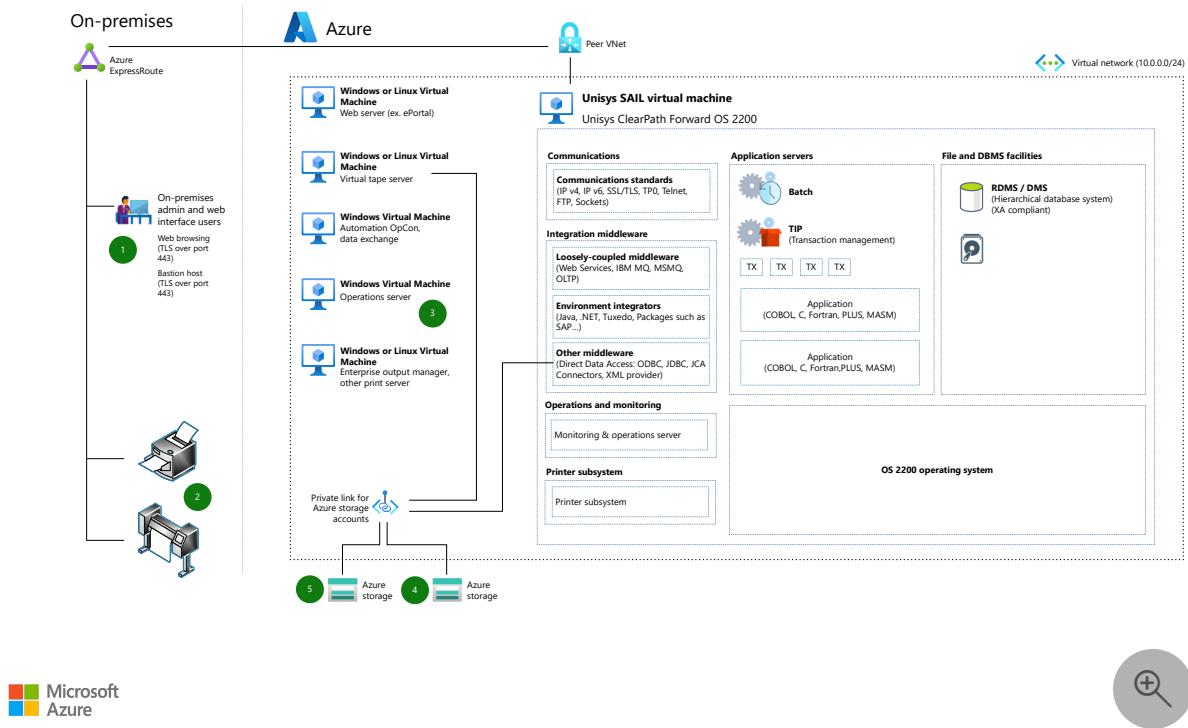
**Example source (premigration) architecture:** The following architecture illustrates a typical, on-premises Unisys CPF Dorado (Sperry 1100/2200) enterprise server.



Download a [Visio file](#) of this architecture.

**Example Azure (postmigration) architecture:** The following architecture illustrates an example utilizing virtualization technologies from Unisys related to the Unisys CPF

## Dorado enterprise server.



Download a [Visio file](#) of this architecture.

## Workflow

Numeric callouts 1, 2, and 3 are used in both diagrams to highlight the similarities between the before and after states of the system.

1. User access is provided over TLS port 443 for accessing web-based applications. The web-based applications presentation layer can be kept unchanged to minimize customer retraining. On the other hand, the web application presentation layer can be updated with modern UX frameworks if desired. Further, for admin access to the virtual machines (VMs), [Azure Bastion hosts](#) can be used to maximize security by minimizing open ports.
2. Printers and other system output devices are supported as long as they're IP attached to the Azure network. Print functions on Dorado are retained so that application changes aren't needed.
3. The Operations function is moved out of the Dorado enterprise server to an Azure VM. You can implement more automation by using an OpCon VM in the ecosystem to monitor and control the entire environment.
4. If physical tapes are in use, they're converted to virtual tapes. Tape formatting and read and write functionality are retained. The tapes are written to Azure or offline storage. Tape functionality is maintained, eliminating the need to rewrite source code. Benefits include [Azure Blob Storage](#) accounts for backup of virtual tape.

files and faster access times because IO operations are conducted directly against disk media.

5. The Dorado storage construct is mapped onto Azure storage, maintaining the Dorado disk drive nomenclature. No application or operations changes are needed.
6. [Azure Site Recovery](#) provides disaster recovery (DR) capabilities by mirroring the Azure VMs to a secondary Azure region. These capabilities ensure a quick failover in the rare case of an Azure datacenter failure.

## Components

- [Azure Virtual Machines](#) is one of several types of on-demand, scalable computing resources that Azure offers. A Virtual Machine gives you the flexibility of virtualization without you having to buy and maintain physical hardware.
- [Azure Virtual Network](#) is the fundamental building block for your private network in Azure. Virtual Network enables many types of Azure resources, such as Virtual Machines, to securely communicate with each other, the internet, and on-premises networks. Virtual Network is similar to a traditional network that operates in your own datacenter but with the added benefits of Azure's infrastructure, such as scale, availability, and isolation. [Network interface cards \(NICs\)](#) enable a VM to communicate with the internet, Azure, and on-premises resources. For example, you can add more NICs to the same VM, which allows the Solaris child VMs to have their own dedicated network interface device and IP address.
- [Azure ExpressRoute](#) lets you extend your on-premises networks into the Microsoft cloud over a private connection facilitated by a connectivity provider. With ExpressRoute, you can establish connections to Microsoft cloud services, such as Azure and Microsoft 365.
- [Azure Site Recovery](#) enables Azure region-to-region failover for DR if a primary region outage occurs.

## Alternatives

The Unisys virtualization of the OS2200 environment provides a *lift and shift* approach to transitioning to Azure. Data, processes, and application code are all maintained and transferred to Azure. Testing is minimal because all applications are carried over from the mainframe.

Other ways to transfer data and processes to Azure include:

- Refactoring the application code to C# or Java by using automated tools. This solution moves the functionality but provides for a code base in an Azure-native

form. This solution takes longer to implement and requires thorough testing to ensure maintained functionality.

- Rewriting the application code to the language of your choice. This solution is usually the longest and most expensive solution. Code is rewritten to account for the application needs. New functionality can be added. This solution requires thorough testing to ensure that the new code performs as expected.

## Scenario details

The Unisys enterprise servers trace their heritage to the first commercially available enterprise servers. The Unisys CPF Dorado (Sperry 1100/2200) and Libra (Burroughs A Series/Master Control Program) systems are full-featured enterprise server operating environments. They can scale vertically to handle mission-critical workloads. You can emulate, convert, or modernize these systems into Azure. Azure offers similar or even improved performance characteristics and service-level agreement (SLA) metrics.

A Unisys transition moves the entire Dorado system from today's hardware to Azure via a VM. The 2200 Exec OS and all processors, libraries, and data appear as they did on the physical environment. The OS requires a license from Unisys. The architecture includes support VMs, which handle functions such as virtual tapes operations, automation and workload management (OpCon), web services, and other support functions. The architecture also uses Azure storage features, including:

- [Azure SSD managed disks](#) are block-level storage volumes managed by Azure and used with Virtual Machines. The available types of disks are ultra disks, premium SSDs, standard SSDs, and standard HDDs. For this architecture, you should use either premium SSDs or ultra disk SSDs.
- [Azure Files](#) is a service that you can use to fully manage file shares in the cloud that are accessible by using the industry-standard Server Message Block (SMB) protocol. Cloud or on-premises deployments of Windows, Linux, and macOS can mount Azure file shares concurrently.

The benefit of this approach is a rapid move to Azure compared to other methodologies. Because hardware maintenance and facility costs are decreased, there's a quick return on investment (ROI). Because the Dorado environment is unchanged, there's no cost associated with retraining users and programmers.

Depending upon your end goal, a transition can be the end state or a first step toward modernizing applications within the Dorado environment or within Azure. This approach provides a measured, planned path for updating applications. It retains the investment in the existing application code. After conversion, you can use other Unisys CloudForte and Azure data analytic services.

## Potential use cases

- Move existing Unisys CPF Dorado workloads to Azure rapidly, with low risk.
- Use [Azure Arc](#) to create a DR plan for an existing on-premises workload.
- Add Unisys CloudForte or Azure data services to existing client capabilities.
- Use Azure-based CPF to serve as a DR, test, or development environment without the need for more hardware or facility resources.

## Considerations

These considerations implement the pillars of the Azure Well-Architected Framework, which is a set of guiding tenets that you can use to improve the quality of a workload. For more information, see [Microsoft Azure Well-Architected Framework](#).

### Reliability

Reliability ensures your application can meet the commitments you make to your customers. For more information, see [Overview of the reliability pillar](#).

Unisys CPF in Azure uses Site Recovery to ensure system availability and consistency.

### Security

Security provides assurances against deliberate attacks and the abuse of your valuable data and systems. For more information, see [Overview of the security pillar](#).

Unisys CPF is a secure system on its own. Azure adds a layer of encryption for data at rest and in motion.

Unisys Stealth technology hides endpoints. Azure offers other security controls.

### Cost optimization

Cost optimization is about reducing unnecessary expenses and improving operational efficiencies. For more information, see [Overview of the cost optimization pillar](#).

Unisys CPF in Azure eliminates hardware maintenance and facility costs up front. Further savings derive from not having to retrain staff on how to operate or use the system. The virtualized computer runs as it did on the datacenter floor.

You can also optimize your costs by following the process to right-size the capacity of your VMs from the beginning, along with simplified resizing as needed. For more

information, see the Well-Architected Framework's [Principles of cost optimization](#).

To estimate the cost of Azure products and configurations, visit the [Azure pricing calculator](#).

To learn more about Unisys CPF offerings and pricing, visit [Unisys CPF products](#).

## Operational excellence

Operational excellence covers the operations processes that deploy an application and keep it running in production. For more information, see [Overview of the operational excellence pillar](#).

Unisys demonstrates operational excellence by presenting a known environment to the staff, while including new services like Site Recovery to provide DR failover.

You can optimize your operational efficiency by deploying your solution with Azure Resource Manager templates and by using Azure Monitor to measure and improve your performance. See the Well-Architected Framework's [Operational excellence principles](#) and [Monitoring for DevOps](#).

## Performance efficiency

Performance efficiency is the ability of your workload to scale to meet the demands placed on it by users in an efficient manner. For more information, see [Performance efficiency pillar overview](#).

Unisys matches the operational performance in Azure with Developer Studio. You can use the gold or platinum tier depending on your workload and operational needs. Use Developer Studio to increase the speed of tasks including new code development, queries, report generation, and other tasks.

## Contributors

*This article is maintained by Microsoft. It was originally written by the following contributors.*

Principal authors:

- [Philip Brooks](#) | Senior Program Manager
- [Adam Gallagher](#) | Senior Solution Manager

*To see non-public LinkedIn profiles, sign in to LinkedIn.*

# Next steps

For more information, contact [legacy2azure@microsoft.com](mailto:legacy2azure@microsoft.com), or see the following resources:

- [Azure ExpressRoute documentation](#)
- [Azure mainframe and midrange migration ↗](#)
- [Azure Virtual Network documentation](#)
- [Create, change, or delete a network interface](#)
- [Introduction to Azure managed disks](#)
- [Mainframe rehosting on Azure Virtual Machines](#)
- [SMA OpCon in Azure](#)
- [Unisys CloudForte for Azure in Azure Marketplace ↗](#)
- [Unisys cloud management ↗](#)
- [Unisys CPF MCP mainframe rehost to Azure using Unisys virtualization](#)
- [Unisys cybersecurity ↗](#)
- [What is Azure Files?](#)

## Related resources

- [Azure database migration guides](#)
- [Mainframe file replication and sync on Azure](#)
- [Modernize mainframe and midrange data](#)

# Unisys ClearPath MCP virtualization on Azure

Azure ExpressRoute

Azure Storage

Azure Virtual Machines

Azure Virtual Network

The Unisys mainframe systems trace their heritage to the first commercially available mainframes. The Unisys ClearPath Forward (CPF) Dorado (legacy Sperry 1100/2200) and Libra (legacy Burroughs A Series/Master Control Program) systems are full-featured mainframe operating environments. They can scale vertically to handle mission-critical workloads. These systems can be emulated, converted, or modernized into Azure. Azure offers similar or even improved performance characteristics and service-level agreement (SLA) metrics.

This article shows how to use virtualization technologies from Microsoft partner Unisys with a legacy Unisys CPF Libra mainframe. This approach allows an accelerated move into Azure. It eliminates the need to rewrite the application code or redesign the database architecture. Legacy code is maintained in its original form—eliminating the need to recompile application code. The application screens, user interactions, and data structures behind the scenes are unchanged, which eliminates the need to retrain your users.

Unisys replatforming lifts the entire Libra system from today's proprietary hardware to Azure as a virtual machine (VM). The Master Control Program (MCP) OS and all processors, libraries, and data appear just as they did on the proprietary environment. The OS requires a license from Unisys. The architecture includes support VMs, which handle functions such as virtual tapes operations, automation and workload management (OpCon), web services, and other support functions.

The benefit of this approach is a rapid move to Azure compared to other methodologies. Because hardware maintenance and facility costs are dropped, there's a quick return on investment (ROI). Because the MCP environment is unchanged, there's no cost associated with retraining users and programmers.

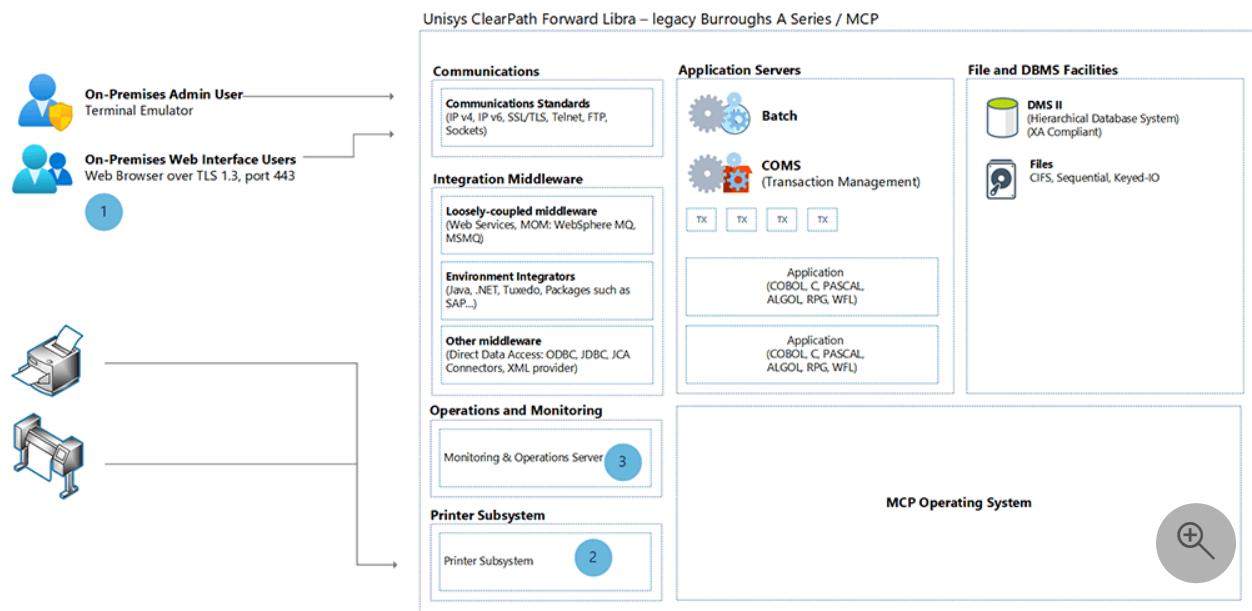
Depending upon the client's end goal, the transitioned Azure MCP could be the end state or a first step toward modernizing applications within the MCP environment or within Azure. This approach to landing in Azure permits a measured, planned path to updating applications. It retains the investment made in existing application code. After conversion, other Unisys Cloud Forte and Azure data analytic services can be employed as well.

# Potential use cases

- Move existing Unisys ClearPath Forward Libra workloads to Azure rapidly, with low risk.
- Use [Azure Arc](#) so Azure can become the disaster recovery (DR) plan for an existing on-premises workload.
- Add Unisys Cloud Forte or Azure data services to existing client capabilities.

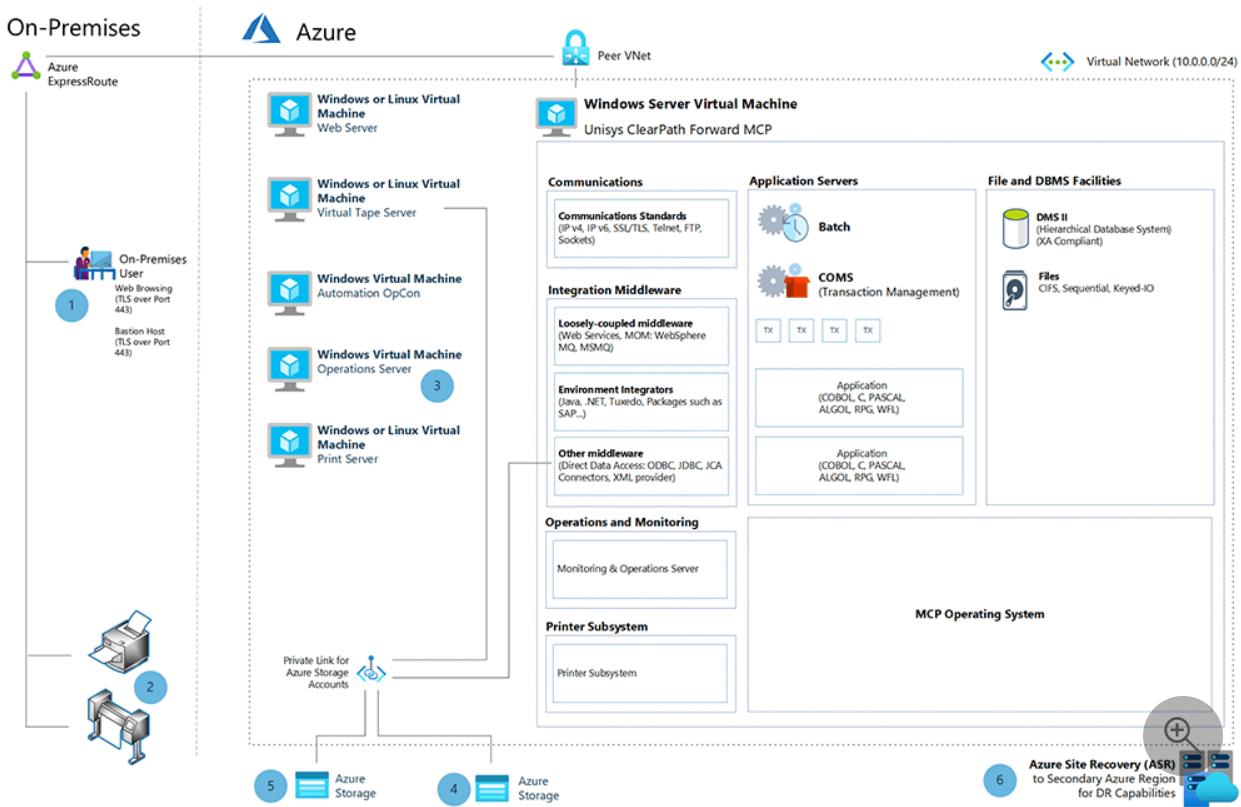
# Architecture

**Example source (premigration) architecture.** The architecture below illustrates a typical, on-premises Unisys ClearPath Forward Libra (legacy Burroughs A Series/MCP) mainframe.



[Download an \*SVG\* of this premigration diagram.](#)

**Example Azure (postmigration) architecture.** The architecture below illustrates an example utilizing virtualization technologies from Microsoft partner Unisys with respect to the legacy Unisys CPF Libra mainframe.



Download an [SVG of this postmigration diagram](#).

## Workflow

The legend matches both diagrams to highlight the similarities between the original and migrated state of the system.

1. Legacy Burroughs terminal emulation for demand and online users is replaced by a web browser to access system resources in Azure. User access provided over TLS port 443 for accessing web-based applications. Web-based applications presentation layer can be kept virtually unchanged to minimize customer retraining. On the other hand, the web-application presentation layer can be updated with modern UX frameworks if desired. Further, for admin access to the VMs, [Azure Bastion hosts](#) can be used to maximize security by minimizing open ports.
2. Printers and other legacy system output devices are supported as long as they're IP attached to the Azure network. Print functions on MCP are retained so that no application changes are needed.
3. **Operations** is moved out of the MCP to an external VM. More automation can be achieved by use of an OpCon VM in the ecosystem to monitor and control the entire environment.
4. If physical tapes are in use, they're converted to virtual tape. Tape formatting and read/write functionality are retained. The tapes are written to Azure or offline storage. Tape functionality is maintained, eliminating the need to rewrite source code. Benefits include [Azure Blob Storage](#) accounts for backup of virtual tape

files and faster access times, as IO operations are conducted directly against disk media.

5. The MCP storage construct can be mapped onto Azure storage, maintaining the MCP drive mapping nomenclature. No application or operations changes are needed.
6. [Azure Site Recovery](#) provides disaster recovery capabilities by mirroring the Azure VMs to a secondary Azure region for quick failover in the rare case of an Azure datacenter failure.

## Components

- [Azure Virtual Machines](#) is one of several types of on-demand, scalable computing resources that Azure offers. An Azure virtual machine gives you the flexibility of virtualization without having to buy and maintain the physical hardware that runs it.
- [Azure Virtual Network](#) is the fundamental building block for your private network in Azure. Virtual Network enables many types of Azure resources, such as Azure Virtual Machines, to securely communicate with each other, the internet, and on-premises networks. Virtual Network is similar to a traditional network that you'd operate in your own datacenter, but with the added benefits of Azure's infrastructure, such as scale, availability, and isolation.
- [Azure Virtual Network interface cards](#) enable an Azure VM to communicate with internet, Azure, and on-premises resources. As shown in this architecture, you can add more network interface cards to the same Azure VM, which allows the Solaris child-VMs to have their own dedicated network interface device and IP address.
- [Azure SSD managed disks](#) are block-level storage volumes managed by Azure and used with Azure Virtual Machines. The available types of disks are ultra disks, premium solid-state drives (SSDs), standard SSDs, and standard hard disk drives (HDDs). For this architecture, we recommend either premium SSDs or ultra disk SSDs.
- [Azure Files](#) offers fully managed file shares in the cloud that are accessible by using the industry-standard Server Message Block (SMB) protocol. Azure file shares can be mounted concurrently by cloud or on-premises deployments of Windows, Linux, and macOS.
- [Azure ExpressRoute](#) lets you extend your on-premises networks into the Microsoft cloud over a private connection facilitated by a connectivity provider. With ExpressRoute, you can establish connections to Microsoft cloud services, such as Microsoft Azure and Office 365.

## Considerations

The following considerations apply to this solution.

## Availability

Unisys CPF in Azure uses Site Recovery to ensure system availability and consistency.

## Operations

Unisys demonstrates operational excellence by presenting a known environment to the staff, while including new capabilities like Azure Site Recovery to provide disaster recovery failover.

You can optimize your operational efficiency by deploying your solution with Azure Resource Manager templates, and by using Azure Monitor to measure and improve your performance. See the Azure Well-Architected Framework's [Operational excellence principles](#) and [Monitoring for DevOps](#).

## Performance

Unisys matches operational performance in Azure with Bronze, Silver, Gold, Platinum, and Titanium offerings to match client workload to operational needs.

## Security

Unisys CPF is inherently a very secure system on its own.

Unisys Stealth technology effectively hides endpoints. Azure offers other security controls.

## Pricing

Unisys CPF in Azure eliminates hardware maintenance and facility costs upfront. Further savings derive from not having to retrain staff how to operate or use the system. The virtualized computer runs just as it did on the datacenter floor.

You can also optimize your costs by following the process to right-size the capacity of your VMs, from the beginning, along with simplified resizing, as needed. For more information, see the Azure Well-Architected Framework's [Principles of cost optimization](#).

To estimate the cost of Azure products and configurations, visit the [Azure pricing calculator](#).

To learn more about Unisys CPF offerings and pricing, visit the [Unisys ClearPath Forward Products webpage](#).

## Contributors

*This article is maintained by Microsoft. It was originally written by the following contributors.*

Principal author:

- [Philip Brooks](#) | Senior TPM

*To see non-public LinkedIn profiles, sign in to LinkedIn.*

## Next steps

For more information, please contact [legacy2azure@microsoft.com](mailto:legacy2azure@microsoft.com), or check out the following resources:

- [Azure Mainframe and midrange migration](#)
- [Mainframe rehosting on Azure virtual machines](#)
- [Unisys CloudForte for Azure in the Azure Marketplace](#)
- [Unisys Cloud Migration Services](#)
- [Unisys Stealth](#)
- [Unisys Documentation Libraries](#)
- [Azure Virtual Network documentation](#)
- [Manage Azure Virtual Network interface cards](#)
- [Introduction to Azure managed disks](#)
- [What is Azure Files?](#)
- [Azure ExpressRoute documentation](#)

## Related resources

- [Unisys ClearPath Forward OS 2200 enterprise server virtualization on Azure](#)
- [Mainframe file replication and sync on Azure](#)
- [Azure Database Migration Guides](#)
- [Unisys mainframe migration to Azure using Avanade AMT](#)
- [Micro Focus Enterprise Server on Azure VMs](#)
- [Modernize mainframe & midrange data](#)
- [Migrate IBM mainframe applications to Azure with TmaxSoft OpenFrame](#)
- [SMA OpCon in Azure](#)

# Unisys Dorado mainframe migration to Azure with Astadia and Micro Focus

Azure Data Factory

Azure SQL Database

Azure Storage

Azure Virtual Machines

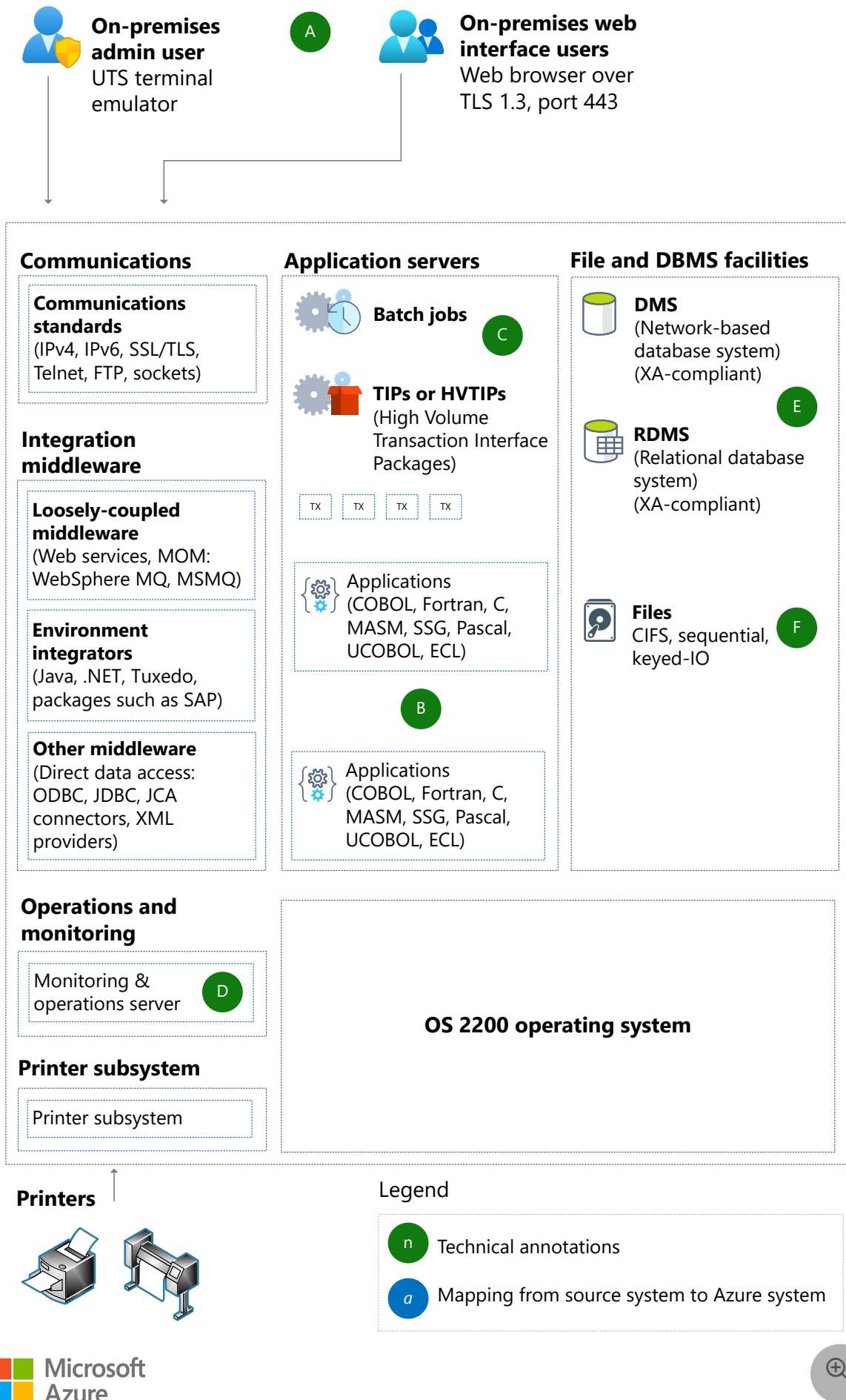
This solution migrates Unisys Dorado mainframe systems to Azure with Astadia and Micro Focus products, without rewriting code, switching data models, or updating screens.

## Architecture

### Legacy architecture

This diagram shows the components that Unisys Sperry OS 1100/2200 mainframe systems typically contain:

# Unisys Dorado – Legacy Sperry 1100/2200



Download a [Visio file](#) of this architecture.

## Workflow

- On-premises users interact with the mainframe (A):
  - Admin users interact through a Universal Terminal System (UTS) terminal emulator.
  - Web interface users interact via a web browser over TLS 1.3 port 443.

Mainframes use communication standards such as:

- Internet Protocol version 4 (IPv4)
- Internet Protocol version 6 (IPv6)
- Secure Sockets Layer (SSL)/TLS
- Telnet
- File Transfer Protocol (FTP)
- Sockets

In Azure, web browsers replace legacy terminal emulation. On-demand and online users can use these web browsers to access system resources.

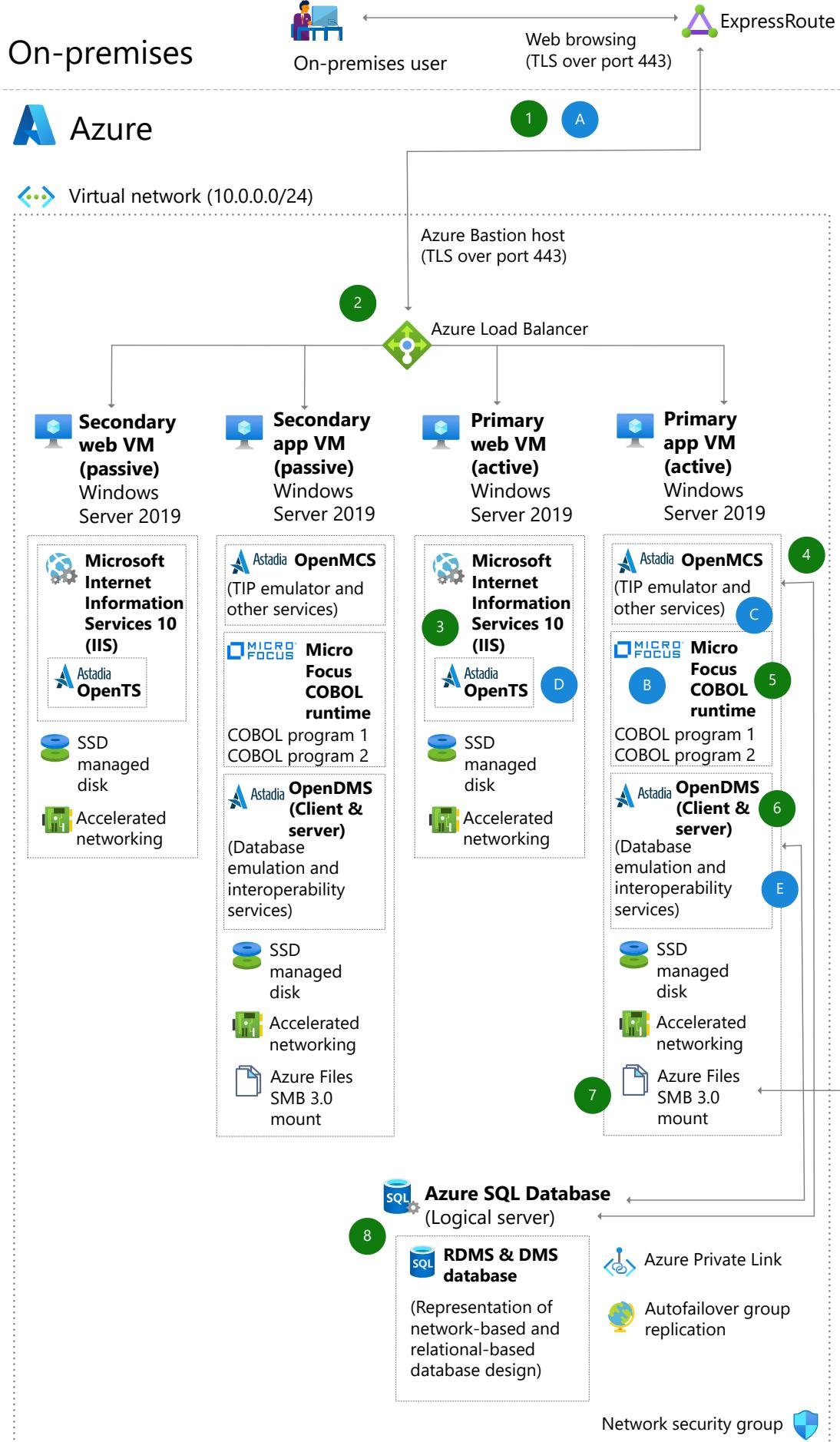
- Mainframe applications are in COBOL, Fortran, C, MASM, SSG, Pascal, UCOBOL, and ECL (B). In Azure, Micro Focus COBOL recompiles COBOL and other legacy application code to .NET. Micro Focus can also maintain and reprocess original base code whenever that code changes. This architecture doesn't require any changes in the original source code.
- Mainframe batch and transaction loads run on application servers (C). For transactions, these servers use TIPs or High Volume TIPs (HVTIPs). In the new architecture:
  - Server topologies handle batch and transaction workloads.
  - An Azure load balancer routes traffic to the server sets.
  - Site Recovery provides high availability (HA) and disaster recovery (DR) capabilities.
- A dedicated server handles workload automation, scheduling, reporting, and system monitoring (D). These functions use the same platforms in Azure.
- A printer subsystem manages on-premises printers.
- Database management systems (E) follow the eXtended Architecture (XA) specification. Mainframes use relational database systems like RDMS and network-based database systems like DMS II and DMS. The new architecture migrates

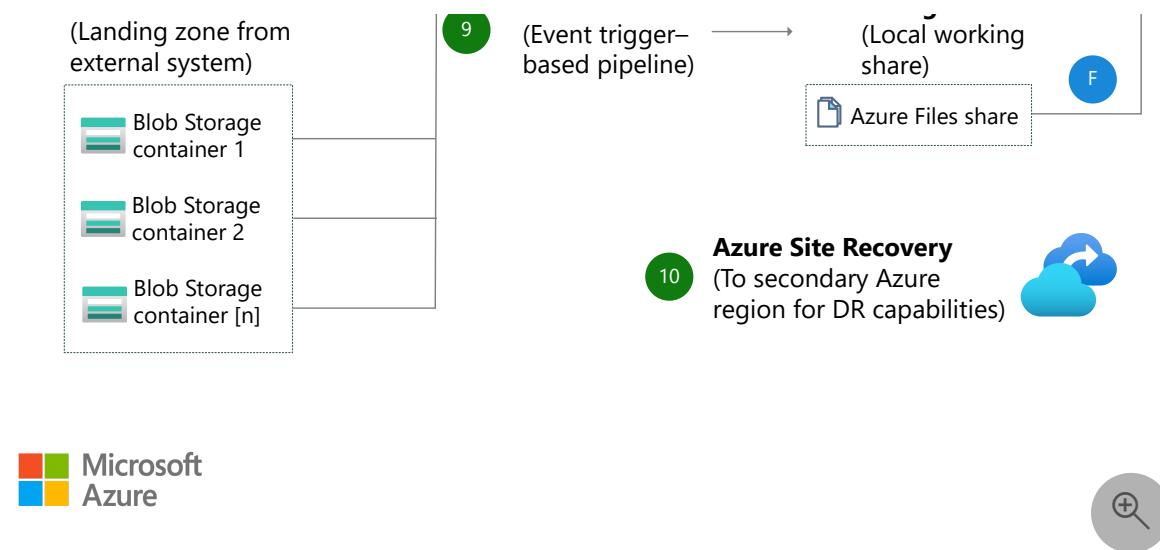
legacy database structures to SQL Database, which provides DR and HA capabilities.

- Mainframe file structures include Common Internet File System (CIFS), flat files, and virtual tape. These file structures map easily to Azure data constructs within structured files or Blob Storage (F). Data Factory provides a modern PaaS data transformation service that fully integrates with this architecture pattern.

## Azure architecture

This architecture demonstrates the solution, after it was migrated to Azure:





Download a [Visio file](#) of this architecture.

## Workflow

1. Transport Layer Security (TLS) connections that use port 443 provide access to web-based applications:
  - To minimize the need for retraining, you can avoid modifying the web application presentation layer during migration. But you can also update the presentation layer to align with UX requirements.
  - Azure Bastion hosts help to maximize security. When you give administrators access to VMs, these hosts minimize the number of open ports.
  - Azure ExpressRoute securely connects on-premises and Azure components.
2. The solution uses two sets of two Azure Virtual Machines (VMs):
  - Within each set, one VM runs the web layer, and one runs the application emulation layer.
  - One set of VMs is the primary, active set. The other set is the secondary, passive set.
  - Azure Load Balancer distributes approaching traffic. When the active VM set fails, the standby set comes online. The load balancer then routes traffic to that newly activated set.
3. Astadia OpenTS simulates Unisys mainframe screens. This component runs presentation layer code in Internet Information Services (IIS) and uses ASP.NET. OpenTS can either run on its own VM or on the same VM as other Astadia emulation products.
4. OpenMCS is a program from Astadia that emulates these components:

- The Unisys Dorado Mainframe Transactional Interface Package (TIP).
  - Other services that Unisys mainframe COBOL programs use.
5. Micro Focus COBOL runs COBOL programs on the Windows server. There's no need to rewrite COBOL code. Micro Focus COBOL can invoke Unisys mainframe facilities through the Astadia emulation components.
6. Astadia OpenDMS emulates the Unisys Dorado mainframe DMS database access technology. With this component, you can migrate tables and data into SQL Database from these systems:
- Relational-based relational database management systems (RDBMSs).
  - Network-based data management software (DMS) databases.
7. An Azure Files share is mounted on the Windows server VM. COBOL programs then have easy access to the Azure Files repository for file processing.
8. With either the Hyperscale or Business Critical service tier, SQL Database provides these capabilities:
- High input/output operations per second (IOPS).
  - High uptime SLA.
- Azure Private Link provides a private, direct connection from VMs to SQL Database through the Azure network backbone. An auto-failover group manages database replication.
9. Data Factory version 2 (V2) provides data movement pipelines that events can trigger. After data from external sources lands in Azure Blob Storage, these pipelines move that data into Azure Files storage. Emulated COBOL programs then process the files.
10. Azure Site Recovery provides disaster recovery capabilities. This service mirrors the VMs to a secondary Azure region. In the rare case of an Azure datacenter failure, the system then provides quick failover.

## Components

This architecture uses the following components:

- [VMs](#) are on-demand, scalable computing resources. An [Azure VM](#) provides the flexibility of virtualization but eliminates the maintenance demands of physical hardware.

- [Azure solid-state drive \(SSD\) managed disks](#) are block-level storage volumes that Azure manages. VMs use these disks. Available types include:
  - Ultra Disks
  - Premium SSD Managed Disks
  - Standard SSD Managed Disks
  - Standard hard disk drives (HDD) Managed Disks

Premium SSDs or Ultra Disks work best with this architecture.

- [Azure Virtual Network](#) is the fundamental building block for private networks in Azure. Through Virtual Network, Azure resources like VMs can securely communicate with each other, the internet, and on-premises networks. An Azure virtual network is like a traditional network operating in a datacenter. But an Azure virtual network also provides scalability, availability, isolation, and other benefits of Azure's infrastructure.

[Virtual network interface cards](#) provide a way for VMs to communicate with internet, Azure, and on-premises resources. You can add network interface cards to a VM to give Solaris child VMs their own dedicated network interface devices and IP addresses.

- [Azure Files](#) is a service that's part of [Azure Storage](#). Azure Files offers fully managed file shares in the cloud. Azure file shares are accessible via the industry standard Server Message Block (SMB) protocol. You can mount these file shares concurrently by cloud or on-premises deployments. Windows, Linux, and macOS clients can access these file shares.
- [Azure Blob Storage](#) is a service that's part of Storage. Blob Storage provides optimized cloud object storage that manages massive amounts of unstructured data.
- [Azure SQL Database](#) is a fully managed PaaS database engine. With AI-powered, automated features, SQL Database handles database management functions like upgrading, patching, backups, and monitoring. SQL Database offers 99.99 percent availability and runs on the latest stable version of the SQL Server database engine and patched operating system. Because SQL Database offers built-in PaaS capabilities, you can focus on domain-specific database administration and optimization activities that are critical for your business.
- [Azure Data Factory](#) is a hybrid data integration service. You can use this fully managed, serverless solution to create, schedule, and orchestrate extract-transform-load (ETL) and extract-load-transform (ELT) workflows.

- [IIS](#) is an extensible web server. Its modular architecture provides a flexible web hosting environment.
- [Azure Load Balancer](#) distributes inbound traffic to back-end pool instances. Load Balancer directs traffic according to configured load-balancing rules and health probes. The back-end pool instances can be Azure VMs or instances in an Azure Virtual Machine Scale Set.
- [Azure ExpressRoute](#) extends on-premises networks into the Microsoft cloud. By using a connectivity provider, ExpressRoute establishes private connections to cloud components like Azure services and Microsoft 365.
- [Azure Bastion](#) provides secure and seamless Remote Desktop Protocol (RDP) and Secure Shell (SSH) access to VMs. This service uses SSL without exposing public IP addresses.
- [Azure Private Link](#) provides a private endpoint in a virtual network. You can use the private endpoint to connect to Azure PaaS services or to customer or partner services.
- [Azure network security groups](#) filter traffic in an Azure virtual network. Security rules determine the type of traffic that can flow to and from Azure resources in the network.
- [Azure Site Recovery](#) keeps applications and workloads running during outages. This service works by replicating VMs from a primary site to a secondary location.
- An [auto-failover group](#) manages the replication and failover of databases to another region. With this feature, you can start failover manually. You can also set up a user-defined policy to delegate failover to Azure.

## Scenario details

Unisys Dorado mainframe systems are full-featured operating environments. You can scale them up vertically to handle mission-critical workloads. But emulating or modernizing these systems into Azure can provide similar or better performance and SLA guarantees. Azure systems also offer added flexibility, reliability, and the benefit of future capabilities.

This architecture uses emulation technology from two Microsoft partners, [Astadia](#) and [Micro Focus](#). The solution provides an accelerated way to move to Azure. There's no need for these steps:

- Rewriting application code.

- Redesigning data architecture or switching from a network-based to a relational-based model.
- Changing application screens.

## Potential use cases

Many cases can benefit from the Astadia and Micro Focus pattern:

- Businesses with Unisys Dorado mainframe systems that can't modify original source code, such as COBOL. Reasons include compliance factors, prohibitive costs, complexity, or other considerations.
- Organizations looking for approaches to modernizing workloads that offer these capabilities:
  - A way to migrate application layer source code.
  - Modern platform as a service (PaaS) services, including:
    - Azure SQL Database with its built-in high availability.
    - Azure Data Factory with its automated and serverless file routing and transformation.

## Considerations

The following considerations, based on the [Microsoft Azure Well-Architected Framework](#), apply to this solution.

## Availability

- Availability sets for VMs ensure enough VMs are available to meet mission-critical batch process needs.
- Load Balancer improves reliability by rerouting traffic to a spare VM set if the active set fails.
- Various Azure components provide reliability across geographic regions through HA and DR:
  - Site Recovery
  - The Business Critical service tier of SQL Database
  - Azure Storage redundancy
  - Azure Files redundancy

# Operational

- Besides scalability and availability, these Azure PaaS components also provide updates to services:
  - SQL Database
  - Data Factory
  - Azure Storage
  - Azure Files
- Consider using [Azure Resource Manager templates \(ARM templates\)](#) to automate deployment of Azure components such as Storage accounts, VMs, and Data Factory.
- Consider using [Azure Monitor](#) to increase monitoring in these areas:
  - Tracking the state of infrastructure.
  - Monitoring external dependencies.
  - App troubleshooting and telemetry through [Application Insights](#).
  - Network component management through [Azure Network Watcher](#).

# Performance efficiency

- SQL Database, Storage accounts, and other Azure PaaS components provide high performance in these areas:
  - Data reads and writes.
  - Hot storage access.
  - Long-term data storage.
- The use of VMs in this architecture aligns with the framework's [performance efficiency pillar](#), since you can optimize the VM configuration to boost performance.

# Scalability

Various Azure PaaS components provide scalability:

- SQL Database
- Data Factory
- Azure Storage
- Azure Files

# Security

Security provides assurances against deliberate attacks and the abuse of your valuable data and systems. For more information, see [Overview of the security pillar](#).

All the components in this architecture work with Azure security components as needed. Examples include network security groups, virtual networks, and TLS encryption.

## Cost optimization

Cost optimization is about looking at ways to reduce unnecessary expenses and improve operational efficiencies. For more information, see [Overview of the cost optimization pillar](#).

To estimate the cost of implementing this solution, use the [Azure pricing calculator](#).

- [VM pricing](#) depends on your compute capacity. This solution helps you [optimize VM costs](#) in these ways:
  - Turning off VMs that aren't in use.
  - Scripting a schedule for known usage patterns.
- For SQL Database:
  - Use the Hyperscale or Business Critical service tier for high input/output operations per second (IOPS) and high uptime SLA.
  - You pay for [computing power and a SQL license](#). But if you have [Azure Hybrid Benefit](#), you can [use your on-premises SQL Server license](#).
- With ExpressRoute, you pay a [monthly port fee](#) and [outbound data transfer charges](#).
- Azure Storage costs depend on [data redundancy options and volume](#).
- Azure Files pricing depends on many factors: [data volume](#), [data redundancy](#), [transaction volume](#), and [the number of file sync servers](#) that you use.
- For SSD managed disk pricing, see [Managed disks pricing](#).
- With Site Recovery, you pay for each [protected instance](#).
- For IIS software plan charges, see [Internet Information Services pricing](#).
- Other services are free with your Azure subscription, but you pay for usage and traffic:
  - With Data Factory, your [activity run volume determines the cost](#).
  - For Virtual Network, [IP addresses carry a nominal charge](#).
  - Private Link costs depend on [endpoints and data volume](#).
  - Load Balancer [rules and traffic incur charges](#).

- With Azure Bastion, the [outbound data transfer volume determines the price](#).
- [Contact Astadia](#) for pricing information on OpenTS, OpenMCS, and OpenDMS.
- [Contact Micro Focus](#) for pricing on Micro Focus COBOL.

## Contributors

*This article is maintained by Microsoft. It was originally written by the following contributors.*

Principal author:

- [Philip Brooks](#) | Senior Technical Program Manager

*To see non-public LinkedIn profiles, sign in to LinkedIn.*

## Next steps

- Contact [legacy2azure@microsoft.com](mailto:legacy2azure@microsoft.com) for more information.
- See the [Azure Friday](#) tech talk with Astadia on [mainframe modernization](#).

## Related resources

- [Unisys ClearPath MCP virtualization on Azure](#)
- [Unisys ClearPath Forward OS 2200 enterprise server virtualization on Azure](#)
- [SMA OpCon in Azure](#)
- [Mainframe rehosting on Azure virtual machines](#)
- Reference architectures:
  - [Unisys mainframe migration to Azure using Avanade AMT](#)
  - [Micro Focus Enterprise Server on Azure VMs](#)
  - [Modernize mainframe & midrange data](#)
  - [Migrate IBM mainframe applications to Azure with TmaxSoft OpenFrame](#)

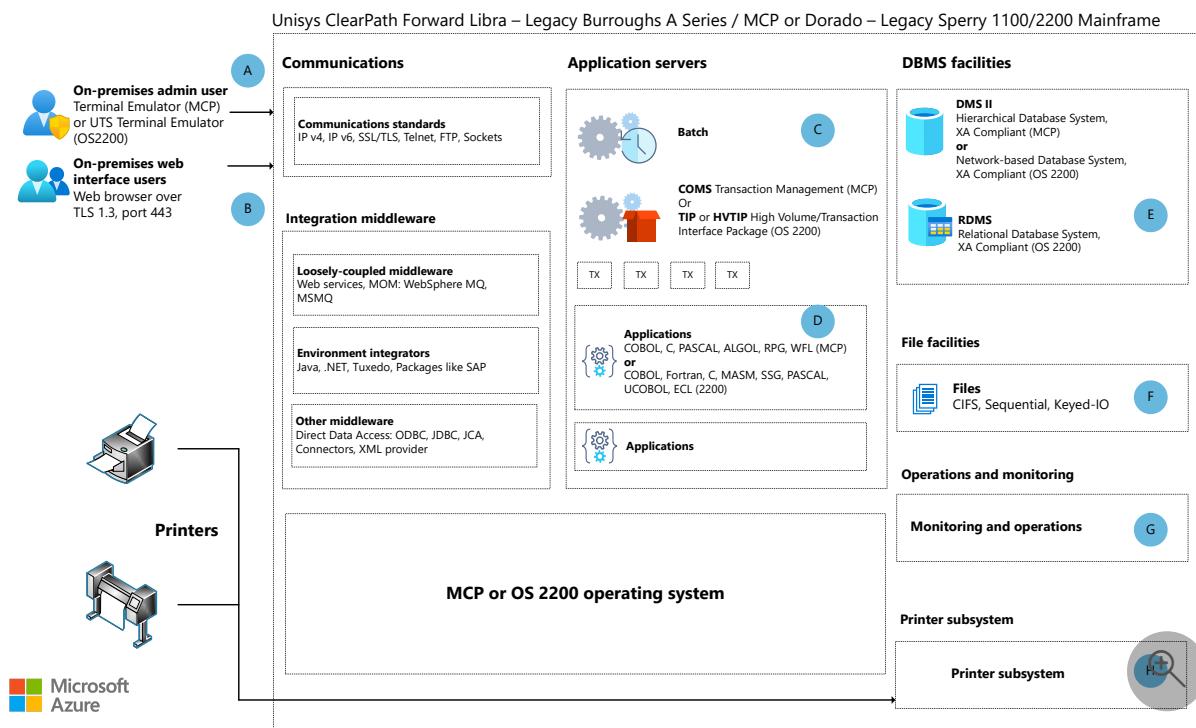
# Unisys mainframe migration with Avanade AMT

Azure Bastion   Azure ExpressRoute   Azure SQL Database   Azure Virtual Machines   Azure Virtual Network

This article describes the conversion technologies that Microsoft partner [Avanade](#) uses to migrate Unisys mainframe workflows to Azure.

## Legacy architecture

The following diagram shows the typical components of Unisys Burroughs MCP or Unisys Sperry OS 1100/2200 mainframe systems.



[Download a Visio file](#) of this architecture.

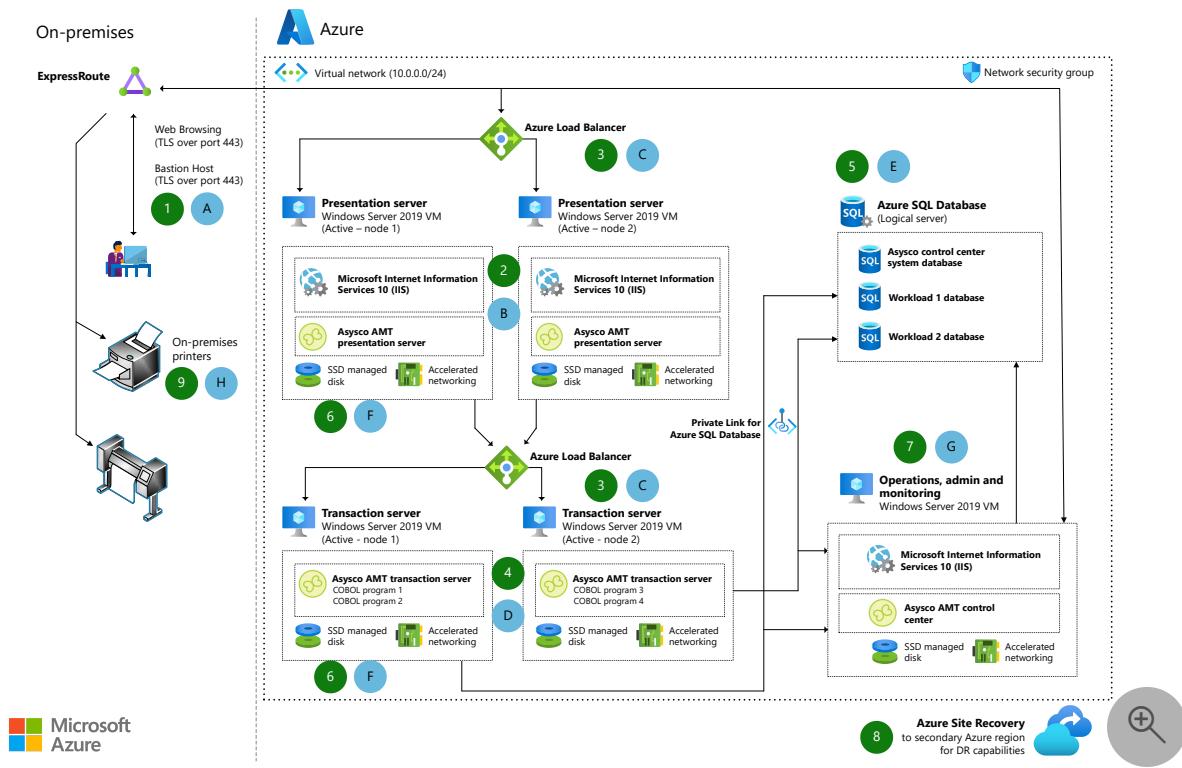
## Workflow

- On-premises admin users interact with the mainframe through Terminal Emulator (MCP systems) or UTS Terminal Emulator (OS 1100/2200 systems) (A). On-premises web interface users can interact via a web browser over TLS 1.3 port 443 (B). Mainframes use communication standards like IPv4, IPv6, SSL/TLS, Telnet, FTP, and Sockets.

- Loosely coupled integrated middleware includes web services, MOM, WebSphere MQ, and MSMQ. Environment integrators include Java, .NET, Tuxedo, and packages like SAP. Other middleware includes direct data access via ODBC, JDBC, and JCA connectors, and XML providers.
- Application servers (**C**) do batch processing, and handle transactions through COMS Transaction Management server for MCP, or High Volume/Transaction Interface Packages (TIP/HVTIP) for OS 2200.
- Applications (**D**) for MCP are written in COBOL, C, PASCAL, ALGOL, RPG, or WFL. For OS 2200, applications are in COBOL, Fortran, C, MASM, SSG, PASCAL, UCOBOL, or ECL.
- Database management systems (**E**) are XA-compliant. MCP uses hierarchical DMS II database systems, and OS 2200 uses network-based DMS II or relational database systems.
- File facilities (**F**) include CIFS, sequential, flat files, keyed IO, and virtual tape files.
- A dedicated server handles operations and monitoring (**G**).
- A printer subsystem (**H**) manages on-premises printers.

## Azure Architecture

The second diagram shows how the Unisys mainframe components can map and migrate to Azure capabilities.



[Download a Visio file](#) of this architecture.

## Workflow

1. A web browser to access Azure system resources replaces terminal emulation for demand and online users (A). Users access web-based applications over TLS port 443. For admin access to the Azure Virtual Machines (VMs), [Azure Bastion](#) hosts maximize security by minimizing open ports.
2. Presentation layer code runs in IIS and uses ASP.NET to maintain the Unisys mainframe user-interface screens (B). The applications' presentation layers can remain virtually unchanged, to minimize end user retraining. Or you can update the web application presentation layer with modern user experience frameworks.
3. The AMT Framework converts mainframe presentation, batch, and transaction loads (C) to sufficient server farms to handle the work. The solution uses two sets of two VMs running the web and application layers, fronted by Azure Load Balancers in *active-active* arrangements to spread query and transaction traffic. Batch-only workflows may have other transaction server sets behind load balancers, rather than presentation servers.
4. The AMT Framework converts legacy application code (D) to C#/.NET. If code needs changing or editing, AMT can maintain and reprocess the original code. Or

you can edit the converted C# code directly to advance the code base to new standards.

5. Legacy database structures (E) can migrate to [Azure SQL Database](#), with the high availability (HA) and disaster recovery (DR) capabilities that Azure provides. Avanade data migration tools can convert DMS and RDMS schemas to SQL. [Azure Private Link](#) provides a private, direct connection from the Azure VMs to Azure SQL Database.
6. File structures (F) map easily to Azure structured file or blob storage data constructs. Features like Azure autofailover group replication can provide data protection.
7. Workload automation, scheduling, reporting, and system monitoring systems (G) that are Azure-capable can keep their current platforms. These platforms include Unisys Operations Sentinel and SMA OpCon. Avanade AMT Control Center can also serve these functions.
8. Azure Site Recovery HA/DR capabilities mirror the Azure VMs to a secondary Azure region for quick failover if there's Azure datacenter failure.
9. The system can support printers (H) and other legacy system output devices if they have IP addresses connected to the Azure network.

## Components

- [Azure Virtual Machines](#) provides on-demand, scalable computing resources. Azure Virtual Machines gives you the flexibility of virtualization without requiring you to buy and maintain physical hardware.
- [Azure Virtual Networks](#) are the fundamental building blocks for Azure private networks. Virtual networks let Azure resources like VMs securely communicate with each other, the internet, and on-premises networks. Although an Azure Virtual Network is similar to a traditional network on premises, it offers the extra benefits of Azure's infrastructure, such as scalability, availability, and isolation.

Virtual network interfaces let Azure VMs communicate with internet, Azure, and on-premises resources. As in this architecture, you can add several network interface cards to one Azure VM. Then child VMs can have their own dedicated network interface devices and IP addresses.
- [Azure managed disks](#) are block-level storage volumes that Azure manages on Azure VMs. The available types of disks are ultra disks, premium solid-state drives

(SSDs), standard SSDs, and standard hard disk drives (HDDs). This architecture works best with Premium SSDs or Ultra Disk SSDs.

- [Azure Files](#) offers fully managed file shares in your Azure Storage Account that are accessible from the cloud or on-premises. Windows, Linux, and macOS deployments can mount Azure file shares concurrently, and access files via the industry standard Server Message Block (SMB) protocol.
- [Azure ExpressRoute](#) lets you extend your on-premises networks into the Microsoft cloud over a private connection facilitated by a connectivity provider. With ExpressRoute, you can establish connections to cloud services like Azure and Office 365.
- [Azure Bastion](#) is a fully managed platform as a service (PaaS) that you provision inside your virtual network. Bastion provides secure and seamless RDP and SSH connectivity to the VMs in your virtual network directly from the Azure portal over TLS.
- [Azure SQL Database](#) is a fully managed PaaS database engine that is always running on the latest stable version of SQL Server and patched OS, with 99.99% availability. SQL Database handles most database management functions like upgrading, patching, backups, and monitoring without user involvement. These PaaS capabilities let you focus on business critical, domain-specific database administration and optimization.
- [Azure Private Link](#) for Azure SQL Database provides a private, direct connection that's isolated to the Azure networking backbone from the Azure VMs to Azure SQL Database.

## Scenario details

Unisys ClearPath mainframe systems are full-featured operating environments that can scale up vertically to handle mission critical workloads. ClearPath mainframe models include Dorado, running Legacy Sperry 1100/2200, and Libra, running Legacy Burroughs A Series/MCP. Emulating, converting, or modernizing these systems into Azure can provide similar or better performance and SLA guarantees, while taking advantage of Azure flexibility, reliability, and future capabilities.

The Automated Migration Technology (AMT) Framework allows an accelerated move into Azure without rewriting application code or redesigning data architecture. The framework converts legacy code to C#, while maintaining the source code in its original

form. Application user interfaces and interactions can be virtually unchanged, minimizing the need for end user retraining.

Avanade AMT Transform automates the migration of the complete mainframe ecosystem to Azure, by converting:

- Transaction application code to AMT COBOL or directly to C# and .NET. AMT maintains the original code structure to use as a baseline or to enable future edits.
- All databases, whether hierarchical, network, or relational, to Azure SQL Database.
- WFL/ECL scripts to Windows PowerShell or to open-source Visual Basic scripts.
- All binary or indexed flat files.

## Potential use cases

The AMT Framework supports several options to move client workloads to Azure:

- One method is to convert and move the entire mainframe system to Azure at once, saving interim mainframe maintenance and facility support costs. This approach carries some risk: All processes, like application conversion, data migration, and testing, must align to allow a smooth transition.
- A second methodology is to move applications from the mainframe to Azure gradually, with complete transition as the ultimate goal. This tactic provides savings per application, and lessons learned to convert each application can help with subsequent conversions. Modernizing each application on its own schedule can be more relaxed than converting everything at once.

This stepped method can also provide more processing cycles on the mainframe as applications convert to Azure. Eventually, starvation of the mainframe as applications convert to Azure can highlight the need to retire the mainframe.

## Considerations

These considerations implement the pillars of the Azure Well-Architected Framework, which is a set of guiding tenets that can be used to improve the quality of a workload. For more information, see [Microsoft Azure Well-Architected Framework](#).

## Availability

- This architecture uses [Azure Site Recovery](#) to mirror the Azure VMs to a secondary Azure region for quick failover and DR if there's Azure datacenter failure.

- [Azure autofailover group replication](#) provides data protection by managing database replication and failover to another region.

## Resiliency

Resiliency is built into this solution because of the Load Balancers. If one presentation or transaction server fails, the other server behind the Load Balancer shoulders the workload.

## Scalability

You can scale out the server sets to provide more throughput. For more information, see [Virtual machine scale sets](#).

## Security

Security provides assurances against deliberate attacks and the abuse of your valuable data and systems. For more information, see [Overview of the security pillar](#).

- This solution uses an Azure network security group (NSG) to manage traffic between Azure resources. For more information, see [Network security groups](#).
- [Private Link for Azure SQL Database](#) provides a private, direct connection isolated to the Azure networking backbone from the Azure VMs to Azure SQL Database.

[Azure Bastion](#) maximizes admin access security by minimizing open ports. Bastion provides secure and seamless secure RDP and SSH connectivity over TLS from the Azure portal to VMs in the virtual network.

## Cost optimization

Cost optimization is about looking at ways to reduce unnecessary expenses and improve operational efficiencies. For more information, see [Overview of the cost optimization pillar](#).

- Azure SQL Database should use [Hyperscale or Business Critical](#) SQL Database tiers for high input/output operations per second (IOPS) and high uptime SLA.
- This architecture works best with Premium SSDs or Ultra Disk SSDs. For more information, see [Managed Disks pricing](#).

# Contributors

*This article is maintained by Microsoft. It was originally written by the following contributors.*

Principal author:

- [Philip Brooks](#) | Senior Technical Program Manager

*To see non-public LinkedIn profiles, sign in to LinkedIn.*

## Next Steps

- [Cloud Adoption Framework](#)
- [Create, change, or delete a network interface](#)
- [Azure setup guide](#)
- [Migration best practices](#)
- [Avanade](#)

For more information, please contact [legacy2azure@microsoft.com](mailto:legacy2azure@microsoft.com).

## Related resources

Explore related resources:

- [Unisys ClearPath Forward MCP mainframe rehost to Azure using Unisys virtualization](#)
- [Unisys ClearPath Forward OS 2200 enterprise server virtualization on Azure](#)
- [SMA OpCon in Azure](#)
- [High-volume batch transaction processing](#)
- [Mainframe file replication and sync on Azure](#)
- [Mainframe access to Azure databases](#)
- [Replicate and sync mainframe data in Azure](#)
- [Unlock legacy data with Azure Stack](#)
- [Modernize mainframe & midrange data](#)

# Use LzLabs Software Defined Mainframe (SDM) in an Azure VM deployment

Azure Virtual Machines

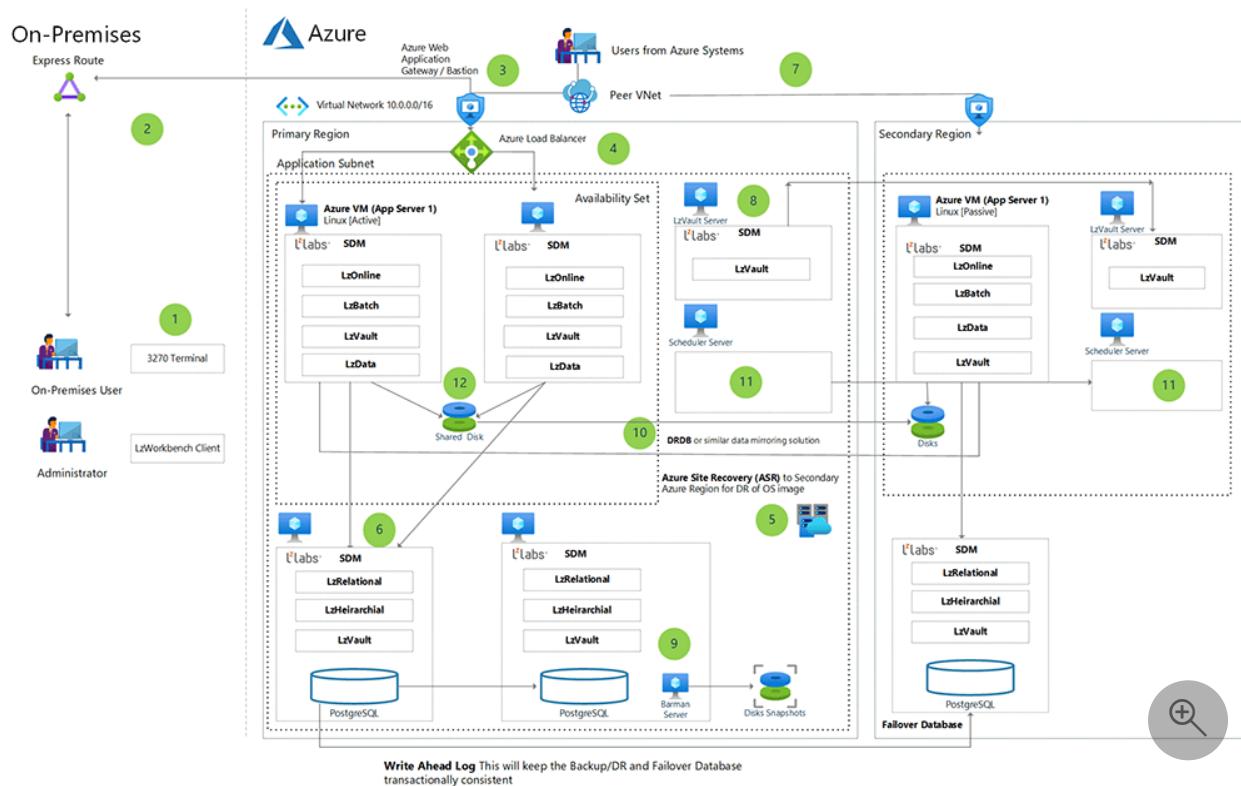
Azure Database for PostgreSQL

Azure Virtual Network

Azure Resource Manager

LzLabs Software Defined Mainframe (SDM) significantly reduces the risk and complexity of legacy workload rehosting by eliminating the need to find, modify, and recompile source code of legacy applications. This approach enables z/Architecture binary executable programs to operate at native speeds on x86\_64 architecture computers running an open systems software stack, opening the path to legacy modernization.

## Architecture



Download an [SVG file](#) of this architecture.

## Workflow

1. LzLabs SDM applications are accessed just like ordinary mainframe applications via a 3270 terminal. You can use any terminal emulator you like. For management, administration, and other activities, the [LzWorkbench client](#) is used. The server component runs on the SDM VM.

2. The port access is typically configured to adapt to the security requirements of the customer.
3. For a secure implementation of SDM, a web services front end should be implemented that consists of:
  - [Azure Application Gateway](#) for any CICS APIs accessing via the web,
  - [Azure Bastion](#) for secure access to manage the VMs,
  - and an [Azure Load Balancer](#).
4. SDM can be configured for failover with virtual network (VNet) peering to the backup, disaster recovery (DR), and Secondary Azure regions. This improves the availability of SDM for production workloads because Azure keeps a consistent replica in case the initial VM goes offline.
5. The SDM virtual machine in the production environment is replicated and kept in sync in the Failover Region by [Azure Site Recovery](#). This service keeps the main OS disk for production and the attached disk images in sync with the Secondary Region SDM. It does this for all attached disks except the disk responsible for index file processing (see item 10). Site Recovery is also used to keep all of the other VM images in sync with the Secondary Region.
6. The database in this architecture is [PostgreSQL IaaS](#). Currently, Azure PostgreSQL Service cannot be used, and PostgreSQL IaaS must be used with SDM in deployments. This is because of a limitation in Azure PostgreSQL for processing user-defined data types (UDT). The database instance in the Secondary Region is kept up to date with the Write Ahead Transaction Log. This allows for cross-region failover. When failover occurs, the mode will be set to active. The same process is used to keep the production failover database transactionally consistent within the Production Region. By using the Write Ahead Transaction Log to keep these two replicas in sync, the database tier is highly available. **Note:** For best performance, the database VM and the SDM VM must be placed in an [Azure proximity placement group](#).
7. A web services front end needs to be deployed for the DR Region in order to maintain secure access to the system. Many mainframe workloads have an API web services layer to access CICS transactions and DB2 data.
8. For RACF and Top Secret identity integration using Active Directory extensions, LzVault provides authentication and authorization in Azure for the security rules migrated from the mainframe.
9. The Barman Server is configured in the Data Tier. This provides snapshot replicas of the PostgreSQL database for point-in-time recovery within both the Production Region and the Secondary Region.
10. As mentioned in item 5, the disk that maintains the indexed file processing for SDM needs to be synchronized across regions using a database mirroring solution.

This is because Azure Site Recovery cannot guarantee the transaction consistency needed for a database. Since the indexed file processing is not within PostgreSQL, a solution must be used that can provide this.

11. To accommodate the scheduling of batch job processing, a scheduler such as [Azure Logic Apps](#) or SMA must be used.
12. To provide availability, there are two SDM VMs deployed into an Azure availability set. [Azure Load Balancer](#) provides load balancing services to the two VMs. State is shared between the two VMs using an [Azure shared disk](#). This is replicated via DRDB to the DR instance.

## Components

- [Azure Virtual Machines](#) (VMs) is one of several types of on-demand, scalable computing resources that Azure offers. An Azure VM gives you the flexibility of virtualization without having to buy and maintain the physical hardware that runs it.
- [Virtual networks](#) (VNets) are the fundamental building block for your private network in Azure Virtual Network. Virtual Network enables many types of Azure resources, including VMs, to securely communicate with each other, the internet, and on-premises networks. Virtual Network is similar to a traditional network that you'd operate in your own datacenter but with the added benefits of Azure's infrastructure, such as scaling, availability, and isolation.
- [Azure Virtual Network Interface](#) (NIC) is a network interface that enables an Azure Virtual Machine to communicate with the internet, other resources in Azure, and on-premises resources. As shown in this architecture, you can add more NICs to the same VM, allowing the Solaris child-VMs to have their own dedicated network interface device and IP address.
- [Azure SSD managed disks](#) are block-level storage volumes that are managed by Azure and used with Azure Virtual Machines. The available types of disks are Ultra Disk, Premium SSDs, Standard SSDs, and Standard Hard Disk Drives (HDDs). For this architecture, we recommend either Premium SSDs or Ultra Disk SSDs.
- [Azure Storage](#) and [Azure Files](#) offer fully managed file shares in the cloud that are accessible via the industry- standard Server Message Block (SMB) protocol. Azure file shares can be mounted concurrently by cloud or on-premises deployments of Windows, Linux, and macOS.
- [Azure ExpressRoute](#) lets you extend your on-premises networks into the Microsoft cloud over a private connection facilitated by a connectivity provider. With ExpressRoute, you can establish connections to Microsoft cloud services, such as Microsoft Azure and Office 365.

- [Azure SQL Database](#) is a fully managed platform as a service (PaaS) database engine that handles most of the database management functions without user involvement, including upgrading, patching, backups, and monitoring. Azure SQL Database is always running on the latest stable version of the SQL Server database engine and patched OS with 99.99-percent availability. PaaS capabilities that are built into Azure SQL Database enable you to focus on the domain-specific database administration and optimization activities that are critical for your business.

## Scenario details

LzLabs Software Defined Mainframe (SDM) is a workload rehosting and mainframe application modernization platform. SDM enables mainframe legacy applications to execute on open systems with no requirement for source-code changes, recompilation, or conversion of data types. SDM also has features that allow legacy applications to be gracefully modernized to contemporary languages and implementations, without compromising the integrity or operation of the system as a whole.

SDM significantly reduces the risk and complexity of legacy workload rehosting by eliminating the need to find, modify, and recompile source code of legacy applications. This approach enables z/Architecture binary executable programs to operate at native speeds on x86\_64 architecture computers running an open systems software stack, opening the path to legacy modernization.

## Potential use cases

- **No source code.** LzLabs is a solution for customers who have mainframe workloads but do not have the source code for the running applications. This can happen if the solution was a customizable off-the-shelf solution (COTS) purchased from an independent software vendor that did not have the source code to the IP. Also, since many of these COBOL-based applications were written long ago, the source code could have been lost or misplaced. LzLabs solves this problem because all that is needed are the load modules (binaries) for execution in SDM.
- **Customer has source code and wants to rehost.** The customer might still have the source code and simply want to rehost their mainframe workloads to reduce cost and enjoy the benefits of a cloud platform like Azure. The COBOL code can be maintained in the SDM in a modern DevOps environment.
- **Failover.** To increase uptime and avoid potential disruptions in business continuity, customers can use LzLabs SDM for a failover environment. In this case, the load

modules are loaded into the SDM and used as a secondary environment if the production environment becomes unavailable.

## Considerations

The following considerations, based on the [Azure Well-Architected Framework](#), apply to this solution.

### Availability

Availability for the application tier is provided with Site Recovery as shown in the diagram. Since LzLabs SDM leverages PostgreSQL for the database tier, availability is provided with a write-ahead transaction log. This ensures that the secondary database is transactionally consistent with the production database.

### Operations

The Azure environment in the diagram is managed either with the Azure portal or [Azure Resource Manager \(ARM\) templates and scripts](#). This allows for the administration of assets (like resizing) and managing security and access. Management of the actual SDM environment is provided via the LzWorkbench administration tool. This allows for the creation and management of execution environments in the SDM.

### Performance efficiency

When migrating mainframe workloads to Azure, keep in mind that the MIPS per vCPU ratio ranges from 50 to 150 MIPS per vCPU. This can vary depending on the type of workload. You will need to profile the mainframe workload for online and batch environments, and then size resources accordingly.

### Scalability

Currently, the solution for scaling SDM is to scale up the virtual machines by adding more vCPUs and memory.

### Security

Access to Azure assets is managed via the Azure portal and/or Azure Resource Manager. Security for the SDM is managed using the Vault component of SDM. This migrates the

security and permissions from RACF or Top Secret into an LDAP-based environment for management in Azure.

## Cost optimization

To estimate the cost of Azure products and configurations, visit the [Azure pricing calculator](#).

To learn more about pricing for LzLabs Software Defined Mainframe products and their related services, visit the [LzLabs website](#).

## Contributors

*This article is maintained by Microsoft. It was originally written by the following contributors.*

Principal author:

- [Jonathon Frost](#) | Principal Software Engineer

*To see non-public LinkedIn profiles, sign in to LinkedIn.*

## Next steps

- For more information, contact legacy2azure@microsoft.com

See the following resources from LzLabs:

- [LzLabs website](#)
- [LzLabs Software Defined Mainframe product overview](#)
- [LzWorkbench](#)
- [LzLabs video library](#)

See the following documentation from Microsoft:

- [Virtual machines in Azure](#)
- [Azure Virtual Network documentation](#)
- [Azure Resource Manager template documentation](#)
- [Azure ExpressRoute documentation](#)
- [Demystifying mainframe to Azure migration white paper](#)
- [Azure Mainframe Migration center](#)
- [Mainframe migration overview](#)
- [Mainframe workloads supported on Azure](#)

- Mainframe rehosting on Azure virtual machines

## Related resources

See the following related articles on Azure Architecture Center:

- [Azure mainframe and midrange architecture concepts and patterns](#)
- [Migrate AIX workloads to Azure with Skytap](#)
- [Refactor mainframe applications with Astadia](#)
- [Refactor mainframe applications with Advanced](#)

# Migrate IBM mainframe applications to Azure with TmaxSoft OpenFrame

Azure Bastion

Azure ExpressRoute

Azure Files

Azure SQL Database

Azure Virtual Machines

## 💡 Solution ideas

This article is a solution idea. If you'd like us to expand the content with more information, such as potential use cases, alternative services, implementation considerations, or pricing guidance, let us know by providing [GitHub feedback](#).

*Lift and shift*, also known as *rehosting*, is the process of mainframe migration to produce an exact copy of an application, workload, and all associated data from one environment to another. Mainframe applications can be migrated from on-premises to public or private cloud.

TmaxSoft OpenFrame is a rehosting solution that makes it easy to lift-and-shift existing IBM zSeries mainframe applications to Microsoft Azure, using a no-code approach. TmaxSoft quickly migrates an existing application, as is, to a zSeries mainframe emulation environment on Azure.

This article illustrates how the TmaxSoft OpenFrame solution runs on Azure. The approach consists of two virtual machines (VMs) running Linux in an [active-active](#) configuration. An Azure Load Balancer distributes incoming traffic between the VMs. OpenFrame emulation software runs on the VMs and provides a zSeries runtime and facilities. Working with the OpenFrame software is an Azure SQL Database. This modernized database layer includes built-in business continuity features.

## Potential use cases

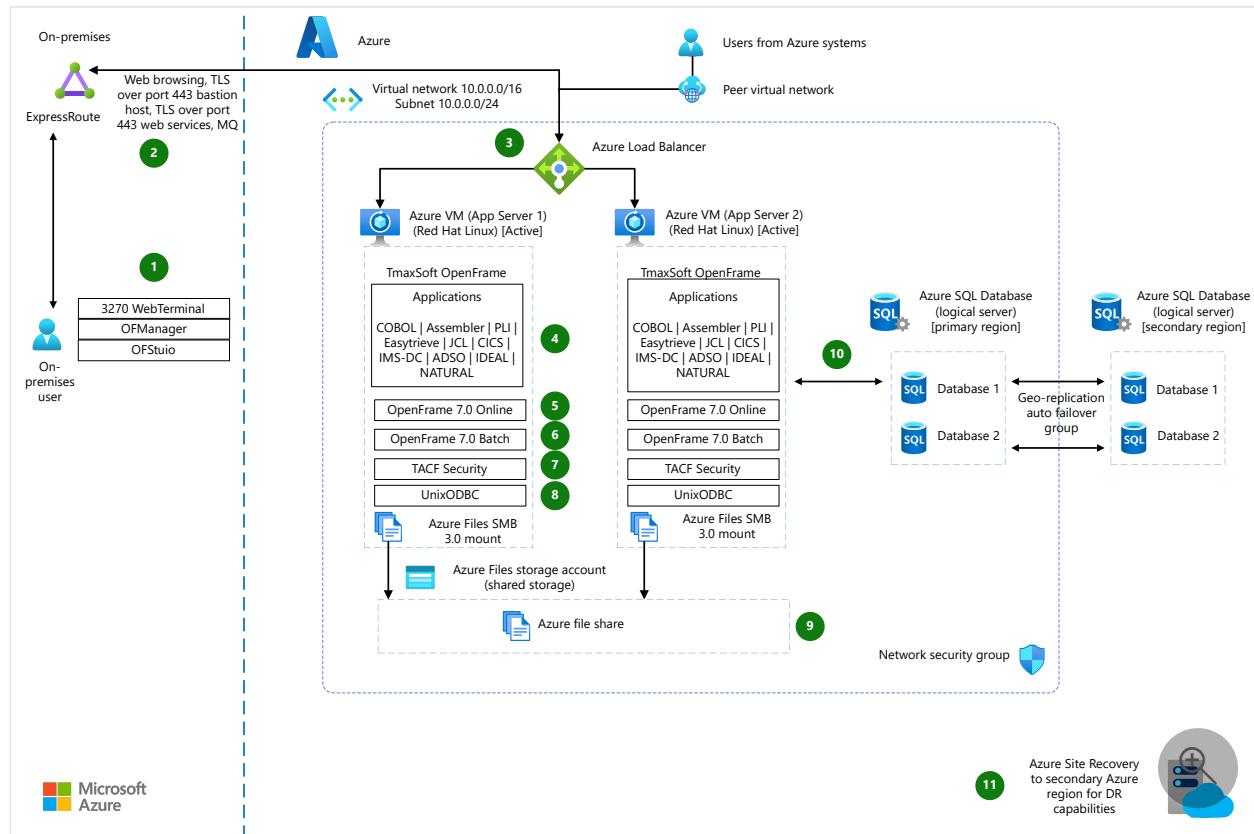
Many scenarios can benefit from TmaxSoft OpenFrame lift and shift. Possibilities include the following cases:

- Businesses seeking to modernize infrastructure and escape the high costs, limitations, and rigidity associated with mainframes.
- Organizations opting to move IBM zSeries mainframe workloads to the cloud without the side effects of a complete redevelopment.

- IBM zSeries mainframe customers who need to migrate mission-critical applications while maintaining continuity with other on-premises applications.
- Teams looking for the horizontal and vertical scalability that Azure offers.
- Businesses that favor solutions offering disaster recovery options.

# Architecture

The following diagram shows the patient record creation request flow:



Download a [Visio file](#) of this architecture.

At the center of the diagram are two virtual machines. Labeled boxes indicate that TmaxSoft OpenFrame software runs on the machines, and each box represents a different type of software. These programs migrate applications to Azure and handle transaction processes. They also manage batch programs and provide security. A load balancer is pictured above the virtual machines. Arrows show that it distributes incoming traffic between the machines. Below the virtual machines, a file sharing system is pictured, and to the right is a database. From arrows, it's clear that the virtual machines communicate with the file share and the database. A dotted line surrounds all these components. Outside that line are on-premises users, Azure users, and disaster recovery services. Arrows show the users interacting with the system. :::image-end:::

1. On-premises users interact with [OpenFrame](#) applications by using 3270 WebTerminal, OFManager, and OFStudio:

- The web application 3270 WebTerminal runs in browsers. This app connects users with [Customer Information Control System \(CICS\)](#) and [Information Management System - Data Communications \(IMS-DC\)](#) applications. By providing access to these 3270 terminal screens, the 3270 WebTerminal app eliminates the need for TN3270 terminal emulation software.
- [OFManager](#) provides tools for executing, monitoring, and managing batch workloads. This web application also monitors and manages datasets and security systems.
- [OFStudio](#) provides an IDE for programming, debugging, and maintaining applications.

2. Azure ExpressRoute creates private connections between the on-premises infrastructure and Azure. Transport Layer Security (TLS) connections that use port 443 provide access to web-based applications:

- After migration, the web application presentation layer remains virtually unchanged. As a result, end users require minimal retraining. Alternatively, the web application presentation layer can be updated to align with UX goals.
- [Azure Bastion hosts](#) work to maximize security. While giving administrators access to VMs, these hosts minimize the number of open ports.
- OpenFrame provides middleware integration. For instance, this functionality works with web services and [message queues \(MQs\)](#).

3. The TmaxSoft solution uses two VMs. Each one runs an application server, and an Azure Load Balancer manages approaching traffic. OpenFrame supports both [active-active](#) and [active-passive](#) configurations.

4. [OpenFrame language compilers](#) migrate COBOL, Assembler, PL/I, Easytrieve, and other mainframe applications to Azure by recompiling the source.

5. [OpenFrame Online](#) provides tools and commands that replace CICS, IMS-DC, Application Development and Maintenance (ADM), and Application Infrastructure and Middleware (AIM) technologies.

6. [OpenFrame Batch](#) provides tools for managing batch programs that replace the job entry subsystem (JES). OpenFrame Batch minimizes code updates by supporting native Job Control Language (JCL) syntax and batch utilities.

7. Tmax Access Control Facility (TACF) Security provides authentication and authorization features in OpenFrame by extracting and migrating mainframe security rules.

8. [UnixODBC \(Open Database Connectivity\)](#) connection drivers communicate with relational database management systems (RDBMSs). Examples include Azure SQL Database, Microsoft SQL Server, Oracle, Db2 LUW, Tibero, Postgres, and MySQL.
9. Azure File Share is mounted on the Linux server VMs. As a result, COBOL programs have easy access to the Azure Files repository for file processing. Load modules and various log files also use Azure File Share.
10. OpenFrame can integrate with any RDBMS. Examples include Azure SQL Database, SQL Server, Oracle, Db2 LUW, Tibero, Postgres, and MySQL. OpenFrame uses ODBC connection drivers to communicate with installed databases.
11. Azure Site Recovery provides disaster recovery (DR) for the virtual machine components.

## Components

- [Azure ExpressRoute](#) extends on-premises networks into the Microsoft cloud by using a connectivity provider. ExpressRoute establishes private connections to Microsoft cloud services like [Microsoft Azure](#) and [Microsoft 365](#).
- [Azure Bastion](#) provides secure and seamless [Remote Desktop Protocol \(RDP\)](#) and [Secure Shell \(SSH\)](#) connectivity to VMs in a network. Instead of using a public IP address, users connect to the VMs directly from the Azure portal.
- [Azure Load Balancer](#) operates at layer four of the [Open Systems Interconnection \(OSI\)](#) model. As the single point of contact for clients, Load Balancer distributes inbound traffic to back-end pool instances. It directs traffic according to configured load-balancing rules and health probes. The back-end pool instances can be Azure VMs or instances in a virtual machine scale set.
- [Azure VMs](#) are one of several types of on-demand, scalable computing resources that are available with Azure. An Azure VM provides the flexibility of virtualization. But it eliminates the maintenance demands of physical hardware. Azure VMs offer a choice of operating systems, including Windows and Linux.
- [Azure Virtual Networks](#) are the fundamental building blocks for private networks in Azure. These networks provide a way for many types of Azure resources, such as Azure VMs, to securely communicate with each other, the internet, and on-premises networks. An Azure virtual network is like a traditional network operating in a data center. But an Azure virtual network also provides scalability, availability, isolation, and other benefits of Azure's infrastructure.

- [Azure Files Storage Accounts](#) and [Azure File Shares](#) are fully managed file shares in the cloud. Azure file shares are accessible via the industry standard [Server Message Block \(SMB\)](#) protocol. They can be mounted concurrently by cloud or on-premises deployments. Windows, Linux, and macOS clients can access these file shares.
- [Azure SQL Database](#) is an intelligent, scalable relational database service built for the cloud. With AI-powered, automated features, Azure SQL Database handles database management functions like upgrading, patching, backups, and monitoring.
- [Azure Site Recovery](#) provides replication, failover, and recovery processes to help keep applications running during outages.

## Next steps

- Contact [legacy2azure@microsoft.com](mailto:legacy2azure@microsoft.com) for more information.
- See [TmaxSoft OpenFrame](#) on Azure Marketplace.
- Read how to [install TmaxSoft OpenFrame on Azure](#).

## Related resources

- [Mainframe rehosting on Azure virtual machines](#)
- [Lift-and-Shift Me Up: The Benefits of Mainframe Rehosting](#)
- [Lift, shift, and modernize: proven mainframe modernization strategies that enable digital transformation](#)

# Stromasys Charon-SSP Solaris emulator on Azure VMs

Azure Storage

Azure Virtual Machines

## 💡 Solution ideas

This article is a solution idea. If you'd like us to expand the content with more information, such as potential use cases, alternative services, implementation considerations, or pricing guidance, let us know by providing [GitHub feedback](#).

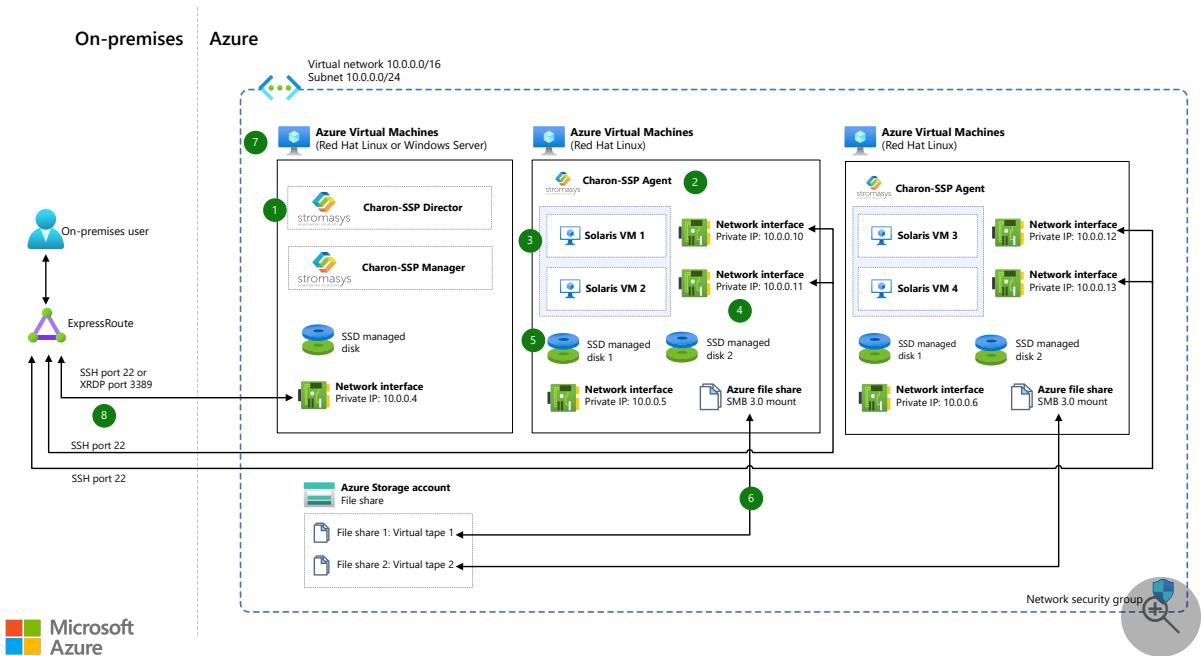
This article shows how an emulator called Charon-SSP from the Microsoft partner, Stromasys, can run SPARC processor-based Solaris virtual machines (VMs) in Azure. Charon-SSP is a member of the Charon cross-platform hardware virtualization product family. The emulator can create virtual replicas of Sun-4m, Sun-4u, or Sun-4v SPARC family members on standard x86-64 Linux physical computers or hypervisors.

Running applications in an emulator on Azure has several benefits, such as reduced operational costs and energy consumption. You can also run multiple application instances on a single x86-64 standard host or existing virtualization infrastructure, giving you the advantages of consolidation while easing legacy system management and maintenance.

## Potential use cases

- Enable low-friction "lift-and-shift" from on-premises workloads running on SPARC Solaris machines into Azure.
- Continue to use applications that run on end-of-life SPARCstation or SPARCserver, without changes.
- Manage multiple server hosts and child Solaris VMs from a single interface.
- Allow mapping to low-cost Azure storage to archive tapes for regulatory and compliance purposes.

## Architecture



Download a [Visio file](#) of this architecture.

## Dataflow

1. The Charon-SSP Director allows managing multiple server hosts, each potentially running one or more child Solaris VMs. This setup provides a single place of management as you scale out your farm of host VMs and their Solaris child VMs. Charon-SSP Manager provides an easy-to-use and intuitive graphical management interface.
2. The Charon-SSP Agent runs on Linux distributions on Azure VMs. This component runs the child Solaris VMs and emulates the SPARC processor architecture.
3. The child Solaris VMs are based on the SPARC processor architecture.
4. The child Solaris VMs each get their own Azure network interface, and therefore have their own dedicated private IP addresses. Optionally, you can easily set up Azure public IP addresses on the same network interfaces.
5. The Solaris VM images can reside on the solid-state drive (SSD) managed disk of the host Azure VM. Azure Ultra SSD managed disks are also a potential option for even higher input/output operations per second (IOPS).
6. Azure Storage Account file shares mounted on the Linux VM allow mapping of the Charon-SSP Virtual Tape Manager to a locally mounted device, which is backed by an Azure Files storage account in the cloud. This mapping allows for low-cost storage of archived tapes for regulatory and compliance purposes.
7. The management VM that runs Charon-SSP Director and Manager can be either Windows-based, or Linux-based with a graphic user interface like [GNOME](#).
8. End users can secure-shell (SSH) connect directly to the Solaris VMs, which have their own dedicated network interface cards and IP addresses.

XDMCP [🔗](#) is available for desktop access to the Solaris VMs. XDMCP isn't an encrypted protocol, so the recommended topology for accessing a Solaris VM via XDMCP is to create a Windows Server VM in Azure as a "hop" server, in which an XDMCP client, such as [MobaXterm](#) [🔗](#), can be installed. With this configuration, all network traffic occurs over the private Azure virtual network.

## Components

- [Azure VMs](#) [🔗](#) are on-demand, scalable computing resources in Azure. An Azure VM gives you the flexibility of virtualization without having to buy and maintain physical hardware. Azure VMs give you a choice of operating systems including Windows and Linux.
- [Azure Virtual Network](#) is the fundamental building block for private networks in Azure. Virtual networks let Azure resources like VMs securely communicate with each other, the internet, and on-premises networks. Azure Virtual Network is similar to a traditional network in your own datacenter, but provides the additional scale, availability, and isolation benefits of the Azure infrastructure.
- [Azure Virtual Network interface cards](#) enable an Azure VM to communicate with internet, Azure, and on-premises resources. As shown in this architecture, you can add additional network interface cards to the same Azure VM, which allows the Solaris child VMs to have their own dedicated network interface devices and IP addresses.
- [Azure SSD managed disks](#) are block-level storage volumes managed by Azure that are used with Azure VMs. The available types of disks are ultra disks, premium SSDs, standard SSDs, and standard hard disk drives (HDDs). For this architecture, we recommend either Premium SSDs or Ultra Disk SSDs.
- [Azure Files](#) storage accounts offer fully managed file shares in the cloud that are accessible via the industry standard Server Message Block (SMB) protocol. Azure file shares can be mounted concurrently by cloud and on-premises deployments of Windows, Linux, and macOS.
- [Azure ExpressRoute](#) lets you extend your on-premises networks into the Microsoft cloud over a private connection facilitated by a connectivity provider. With ExpressRoute, you can establish connections to Microsoft cloud services like Microsoft Azure and Microsoft 365.
- [Stromasys Charon-SSP](#) [🔗](#) emulator recreates the SPARC virtual hardware layer on industry standard x86-64 computer systems and VMs. The virtual SPARC virtual hardware layer is compatible with any Sun software running on it, so there's no

need for code conversion or source code. Charon-SSP is fully compatible with SPARC storage, Ethernet, and serial I/O hardware.

## Next steps

- For more information, please contact [legacy2azure@microsoft.com](mailto:legacy2azure@microsoft.com).
- See [Charon-SSP](#) on the Stromasys website.
- Read the [Charon-SSP Azure Setup Guide](#).

# Replicate mainframe data by using Precisely Connect

Azure SQL Database

Azure SQL Managed Instance

Azure Synapse Analytics

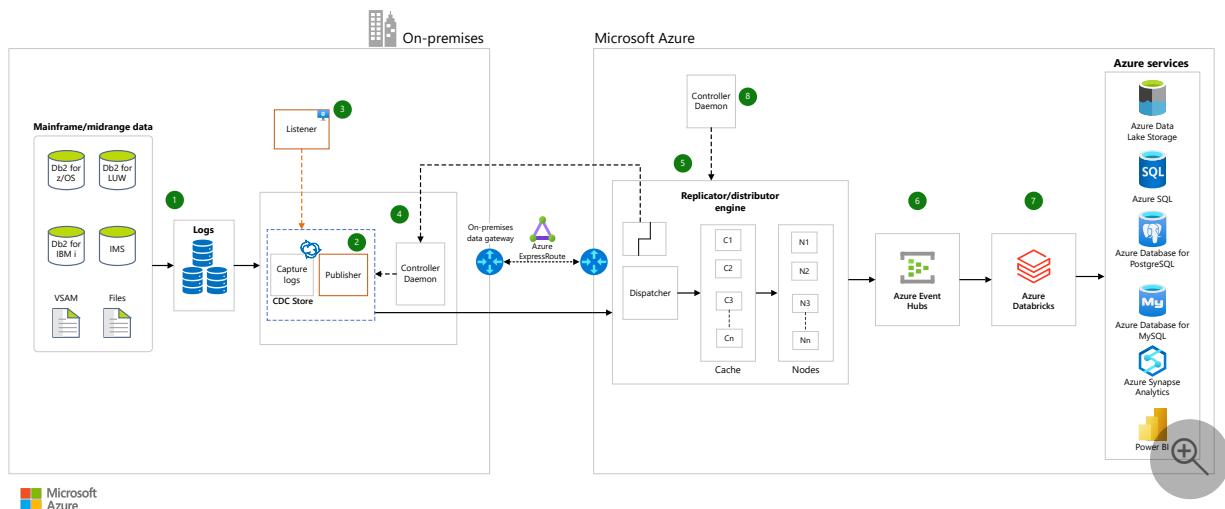
Azure Databricks

Azure Event Hubs

This article describes how to use Precisely Connect to migrate mainframe and midrange systems to Azure.

*Apache®, [Spark](#), and the flame logo are either registered trademarks or trademarks of the Apache Software Foundation in the United States and/or other countries. No endorsement by The Apache Software Foundation is implied by the use of these marks.*

## Architecture



Download a [Visio file](#) of this architecture.

## Workflow

1. A Connect agent component captures change logs by using mainframe or midrange native utilities and caches the logs in temporary storage.
2. For mainframe systems, a publisher component on the mainframe manages data migration.
3. For midrange systems, in place of the publisher, a listener component manages data migration. It's located on either a Windows or Linux machine.
4. The publisher or listener moves the data from on-premises to Azure via an enhanced-security connection. The publisher or listener handles the commit and

rollback of transactions for each unit of work, maintaining the integrity of data.

5. The Connect Replicator Engine captures the data from the publisher or listener and applies it to the target. It distributes data for parallel processing.
6. The target is a database that receives the changes via ODBC or ingests the changes via Azure Event Hubs.
7. The changed data is consumed by Azure Databricks and applied to Azure data platform services.
8. The Connect Controller Daemon authenticates the request and establishes the socket connection between the publisher or listener and the Replicator Engine.

## Components

### Networking and identity

- [Azure ExpressRoute](#) extends your on-premises networks to the Azure cloud platform over a private connection from a connectivity provider.
- [Azure VPN Gateway](#) enables you to create virtual network gateways that send encrypted traffic between an Azure virtual network and an on-premises location over the public internet.
- [Microsoft Entra ID](#) is an identity and access management service that synchronizes with on-premises Active Directory.

### Storage

- [Azure SQL Database](#) is part of the Azure SQL family. It's built for the cloud and provides all the benefits of a fully managed and evergreen platform as a service (PaaS). SQL Database also provides AI-powered automated features that optimize performance and durability. Serverless compute and Hyperscale storage options automatically scale resources on demand.
- [Azure Database for PostgreSQL](#) is a fully managed relational database service that's based on the community edition of the open-source PostgreSQL database engine.
- [Azure Database for MySQL](#) is a fully managed relational database service that's based on the community edition of the open-source MySQL database engine.
- [Azure SQL Managed Instance](#) is an intelligent, scalable cloud database service that offers all the benefits of a fully managed and evergreen PaaS. SQL Managed Instance has nearly 100 percent compatibility with the latest SQL Server Enterprise edition database engine. It also provides a native virtual network implementation that addresses common security concerns.

- [Azure Synapse Analytics](#) is a fast and flexible cloud data warehouse that helps you scale, compute, and store elastically and independently, with a massively parallel processing architecture.
- [Azure Storage](#) is a cloud storage solution that includes object, file, disk, queue, and table storage. Services include hybrid storage solutions and tools for transferring, sharing, and backing up data.

## Analysis and reporting

- [Power BI](#) is a suite of business analytics tools that can deliver insights throughout your organization. By using Power BI, you can connect to hundreds of data sources, simplify data preparation, and drive ad hoc analysis.

## Monitoring

- [Azure Monitor](#) provides a comprehensive solution for collecting, analyzing, and acting on telemetry from cloud and on-premises environments. Features include Application Insights, Azure Monitor Logs, and Log Analytics.

## Data integrators

- [Precisely Connect](#) can integrate data from multiple sources and provide real-time replication to Azure. You can use it to replicate data without making changes to your application. Connect can also improve the performance of extract, transform, load (ETL) jobs.
- [Azure Databricks](#) is based on Apache Spark and integrates with open-source libraries. It provides a unified platform for running analytics workloads. You can use Python, Scala, R, and SQL languages to frame ETL pipelines and orchestrate jobs.
- [Azure Event Hubs](#) is a real-time ingestion service that can process millions of records per second. You can ingest data from multiple sources and use it for real-time analytics. You can easily scale Event Hubs based on the volume of data.

## Scenario details

You can use various strategies to migrate mainframe and midrange systems to Azure. Data migration plays a key role in this process. In a hybrid cloud architecture, data needs to be replicated between mainframe or midrange systems and the Azure data platform. To maintain the integrity of the data, you need real-time replication for business-critical applications. Precisely Connect can help you replicate data from mainframe and

midrange data sources to the Azure data platform in real time by using change data capture (CDC) or by using batch ingestion.

Precisely Connect supports various mainframe and midrange data sources, including Db2 z/OS, Db2 LUW, Db2 for i, IMS, VSAM, files, and copybooks. It migrates them to Azure targets, like SQL Database, Azure Database for PostgreSQL, Azure Database for MySQL, Azure Data Lake Storage, and Azure Synapse Analytics, without affecting applications. It also supports scalability based on data volume and customer requirements. It replicates data without affecting performance or straining the network.

## Potential use cases

This solution applies to the following scenarios:

- Data replication from mainframe and midrange data sources to the Azure data platform.
- In a hybrid cloud architecture, data sync between mainframe or midrange systems and the Azure data platform.
- Near real-time analytics on Azure, based on operational data from mainframe or midrange systems.
- Migration of data from mainframe or midrange systems to Azure without affecting applications.

## Considerations

These considerations implement the pillars of the Azure Well-Architected Framework, a set of guiding tenets that you can use to improve the quality of a workload. For more information, see [Microsoft Azure Well-Architected Framework](#).

## Reliability

Reliability ensures that your application can meet the commitments you make to your customers. For more information, see [Overview of the reliability pillar](#).

Use [Azure Monitor](#) and [Application Insights](#) to monitor your data migration. Set up alerts for proactive management. For more information about reliability in Azure, see [Designing reliable Azure applications](#).

## Cost optimization

Cost optimization is about reducing unnecessary expenses and improving operational efficiencies. For more information, see [Overview of the cost optimization pillar](#).

- Replicating data to Azure and processing it in Azure services can be more cost effective than maintaining it in a mainframe system.
- The Cost Management tool in the Azure portal provides a cost analysis view that can help you analyze your spending.
- You can use Azure Databricks to resize your cluster with autoscaling to optimize costs. Doing so can be less expensive than using a fixed configuration.
- Azure Advisor provides recommendations to optimize performance and cost management.

Use the [Azure pricing calculator](#) to estimate the cost of implementing this solution.

## Performance efficiency

Performance efficiency is the ability of your workload to scale to meet the demands placed on it by users in an efficient manner. For more information, see [Performance efficiency pillar overview](#).

- Precisely Connect can scale based on the volume of data and optimize data replication.
- The Connect Replicator Engine can distribute data for parallel processing. You can balance distribution based on the ingestion of workloads.
- SQL Database serverless can scale automatically based on the volume of workloads.
- Event Hubs can scale based on throughput units and the number of partitions.

For more information, see [Autoscaling best practices in Azure](#).

## Contributors

*This article is maintained by Microsoft. It was originally written by the following contributors.*

Principal author:

- [Seetharaman Sankaran](#) | Senior Engineering Architect

Other contributor:

- [Mick Alberts](#) | Technical Writer

*To see non-public LinkedIn profiles, sign in to LinkedIn.*

## Next steps

- [Change Data Capture with Connect ↗](#)
- [What is Azure ExpressRoute?](#)
- [What is VPN Gateway?](#)
- [What is Azure SQL Database?](#)
- [Contact Mainframe Data Modernization Engineering at Microsoft](#)

## Related resources

- [Modernize mainframe and midrange data](#)
- [Re-engineer mainframe batch applications on Azure](#)
- [Replicate and sync mainframe data on Azure](#)
- [Mainframe access to Azure databases](#)
- [Mainframe file replication and sync on Azure](#)

# Mainframe and midrange data replication to Azure using Qlik

Azure Event Hubs

Azure Data Lake

Azure Databricks

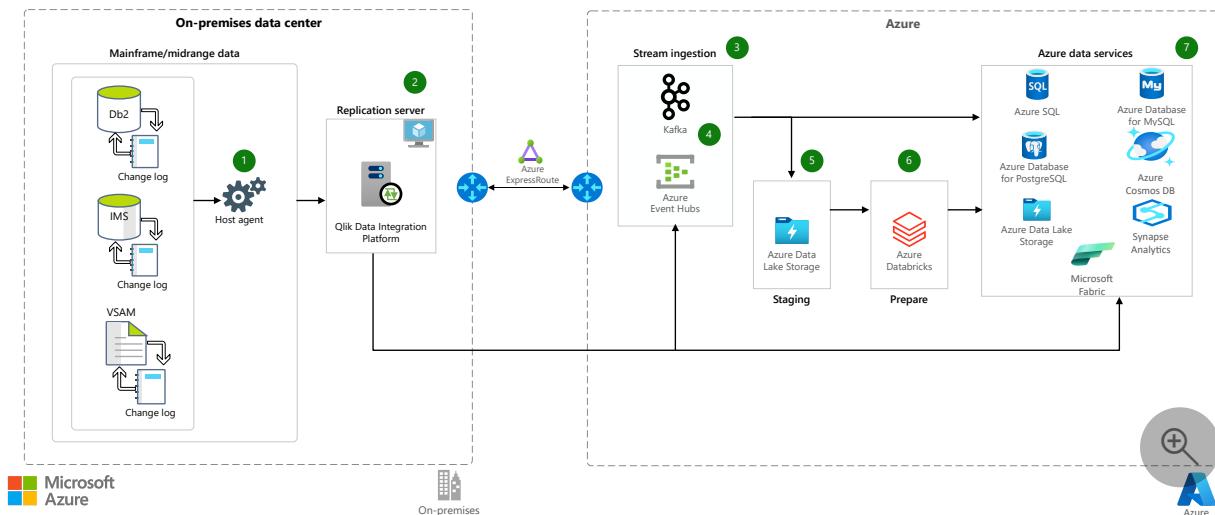
This solution uses an on-premises instance of Qlik to replicate on-premises data sources to Azure in real time.

## ⓘ Note

Pronounce "Qlik" like "click".

*Apache® and Apache Kafka® are either registered trademarks or trademarks of the Apache Software Foundation in the United States and/or other countries. No endorsement by The Apache Software Foundation is implied by the use of these marks.*

## Architecture



Download a [Visio file](#) of this architecture.

## Workflow

1. **Host agent:** The Host agent on the on-premises system captures change log information from Db2, IMS, and VSAM data stores, and passes it to the Qlik Replication server.

2. **Replication server:** The Qlik Replication server software passes the change log information to Kafka and Azure Event Hubs. Qlik in this example is on-premises, but it could instead be deployed on a virtual machine in Azure.
3. **Stream ingestion:** Kafka and Event Hubs provide message brokers to receive and store change log information.
4. **Kafka Connect:** The Kafka Connect API is used to get data from Kafka for updating Azure data stores such as Azure Data Lake Storage, Azure Databricks, and Azure Synapse Analytics.
5. **Data Lake Storage:** Data Lake Storage is a staging area for the change log data.
6. **Databricks:** Databricks processes the change log data and updates the corresponding files on Azure.
7. **Azure data services:** Azure provides a variety of efficient data storage services.

Prominent among these are:

- Relational databases services:
  - SQL Server on Azure Virtual Machines
  - Azure SQL Database
  - Azure SQL Managed Instance
  - Azure Database for PostgreSQL
  - Azure Database for MySQL
  - Azure Cosmos DB

There are many factors to consider when choosing a data storage service: type of workload, cross-database queries, two-phase commit requirements, ability to access the file system, amount of data, required throughput, latency, and so on.

- **Azure non-relational database services:** Azure Cosmos DB, a NoSQL database, provides quick response, automatic scalability, and guaranteed speed at any scale.
- **Azure Synapse Analytics:** Synapse Analytics is an analytics service that brings together data integration, enterprise data warehousing, and big data analytics. With it, you can query data by using either serverless or dedicated resources at scale.
- **Microsoft Fabric:** Microsoft Fabric is an all-in-one analytics solution for enterprises. It covers everything from data movement to data science, Real-Time Analytics, and business intelligence. It offers a comprehensive suite of services, including data lake, data engineering, and data integration.

## Components

This architecture consists of several Azure cloud services and is divided into four categories of resources: networking and identity, application, storage, and monitoring. The services for each and their roles are described in the following sections.

## Networking and identity

- [Azure ExpressRoute](#) extends your on-premises networks into cloud services offered by Microsoft over a private connection from a connectivity provider. With ExpressRoute, you can establish connections to cloud services such as Microsoft Azure and Office 365.
- [Azure VPN Gateway](#) is a specific type of virtual network gateway that sends encrypted traffic between an Azure virtual network and an on-premises location over the public internet.
- [Microsoft Entra ID](#) is an identity and access management service that can synchronize with an on-premises active directory.

## Application

- [Azure Event Hubs](#) is a big data streaming platform and event ingestion service that can store Db2, IMS, and VSAM change data messages. It can receive and process millions of messages per second. Data sent to an event hub can be transformed and stored by using a real-time analytics provider or a custom adapter.
- [Apache Kafka](#) is an open-source distributed event streaming platform that's used for high-performance data pipelines, streaming analytics, data integration, and mission-critical applications. It can be easily integrated with Qlik data integration to store Db2 change data.
- [Azure Data Lake Storage](#) Azure Data Lake Storage provides a data lake for storing the processed on-premises change log data.
- [Azure Databricks](#) is a cloud-based data engineering tool that's based on Apache Spark. It can process and transform massive quantities of data. You can explore the data by using machine learning models. Jobs can be written in R, Python, Java, Scala, and Spark SQL.

## Storage

- [Azure Storage](#) is a set of massively scalable and secure cloud services for data, apps, and workloads. It includes [Azure Files](#), [Azure Table Storage](#), and [Azure Queue Storage](#). Azure Files is often an effective tool for migrating mainframe workloads.

- [Azure Cosmos DB](#) is a fully managed NoSQL database service with open-source APIs for MongoDB and Cassandra. A possible application is to migrate mainframe non-tabular data to Azure.

## Monitoring

- [Azure Monitor](#) delivers a comprehensive solution for collecting, analyzing, and acting on telemetry from cloud and on-premises environments. It includes:
  - Application Insights, for analyzing and presenting telemetry.
  - Monitor Logs, which collects and organizes log and performance data from monitored resources. Data from different sources such as platform logs from Azure services, log and performance data from virtual machines agents, and usage and performance data from applications can be consolidated into a single workspace to be analyzed together. Analysis uses a sophisticated query language that's capable of quickly analyzing millions of records.
  - Log Analytics, which can query Monitor logs. A powerful query language lets you join data from multiple tables, aggregate large sets of data, and perform complex operations with minimal code.

## Alternatives

- The diagram shows Qlik installed on-premises, a recommended best practice to keep it close to the on-premises data sources. An alternative is to install Qlik in the cloud on an Azure virtual machine.
- Qlik Data Integration can deliver directly to Databricks without going through Kafka or an event hub.
- Qlik Data integration can't replicate directly to Azure Cosmos DB, but you can integrate Azure Cosmos DB with an event hub by using event-sourcing architecture.

## Scenario details

Many organizations use mainframe and midrange systems to run demanding and critical workloads. Most applications use one or more databases, and most databases are shared by many applications, often on multiple systems. In such an environment, modernizing to the cloud means that on-premises data must be provided to cloud-based applications. Therefore, data replication becomes an important modernization tactic.

The [Qlik](#) Data Integration platform includes Qlik Replication, which does data replication. It uses change data capture (CDC) to replicate on-premises data stores in real time to Azure. The change data can come from Db2, IMS, and VSAM change logs. This replication technique eliminates inconvenient batch bulk loads. This solution uses an on-premises instance of Qlik to replicate on-premises data sources to Azure in real time.

## Potential use cases

This solution might be appropriate for:

- Hybrid environments that require replication of data changes from a mainframe or midrange system to Azure databases.
- Online database migration from Db2 to an Azure SQL database with little downtime.
- Data replication from various on-premises data stores to Azure for consolidation and analysis.

## Considerations

These considerations implement the pillars of the Azure Well-Architected Framework, which is a set of guiding tenets that can be used to improve the quality of a workload. For more information, see [Microsoft Azure Well-Architected Framework](#).

## Reliability

Reliability ensures your application can meet the commitments you make to your customers. For more information, see [Overview of the reliability pillar](#).

- Qlik Data Integration can be configured in a high-availability cluster.
- The Azure database services support zone redundancy and can be designed to fail over to a secondary node in case of an outage or during a maintenance window.

## Security

Security provides assurances against deliberate attacks and the abuse of your valuable data and systems. For more information, see [Overview of the security pillar](#).

- ExpressRoute provides a private and efficient connection to Azure from on-premises, but you could instead use [site-to-site VPN](#).

- Azure resources can be authenticated by using Microsoft Entra ID. Permissions can be managed by role-based access control.
- Database services in Azure support various security options, such as:
  - Data Encryption at rest.
  - Dynamic data masking.
  - Always-encrypted database.
- For general guidance on designing secure solutions, see the [Azure Security Documentation](#).

## Cost optimization

Cost optimization is about looking at ways to reduce unnecessary expenses and improve operational efficiencies. For more information, see [Overview of the cost optimization pillar](#).

Use the [Azure Pricing Calculator](#) to estimate costs for your implementation.

## Operational excellence

Operational excellence covers the operations processes that deploy an application and keep it running in production. For more information, see [Overview of the operational excellence pillar](#).

- You can combine Monitor's Application Insights and Log Analytics features to monitor the health of Azure resources. You can set alerts so that you can manage proactively.
- For guidance on resiliency in Azure, see [Designing reliable Azure applications](#).

## Performance efficiency

Performance efficiency is the ability of your workload to scale to meet the demands placed on it by users in an efficient manner. For more information, see [Performance efficiency pillar overview](#).

Databricks, Data Lake Storage, and other Azure databases have auto-scaling capabilities. For more information, see [Autoscaling](#).

## Contributors

*This article is maintained by Microsoft. It was originally written by the following contributors.*

Principal author:

- [Nithish Aruldoss](#) | Engineering Architect
- [Ashish Khandelwal](#) | Principal Engineering Architecture Manager

*To see non-public LinkedIn profiles, sign in to LinkedIn.*

## Next steps

- [Qlik Data Integration platform](#)
- [Unleash New Azure Analytics Initiatives \(PDF data sheet\)](#)
- [What is Azure ExpressRoute?](#)
- [What is VPN Gateway?](#)
- [What is Microsoft Entra ID?](#)
- [Azure Event Hubs — A big data streaming platform and event ingestion service](#)
- [Introduction to Azure Data Lake Storage Gen2](#)
- [Introduction to the core Azure Storage services](#)
- [What is Azure SQL Database](#)
- [Welcome to Azure Cosmos DB](#)
- [Azure Monitor overview](#)
- [What is Application Insights?](#)
- [Azure Monitor Logs overview](#)
- [Log queries in Azure Monitor](#)
- [Contact us \(select to create email\)](#)

## Related resources

- [Modernize mainframe and midrange data](#)
- [Re-engineer mainframe batch applications on Azure](#)
- [Replicate and sync mainframe data in Azure](#)
- [Mainframe access to Azure databases](#)
- [Mainframe file replication and sync on Azure](#)

# Mainframe and midrange data replication to Azure using tcVISION

Azure Database Migration service

Azure Functions

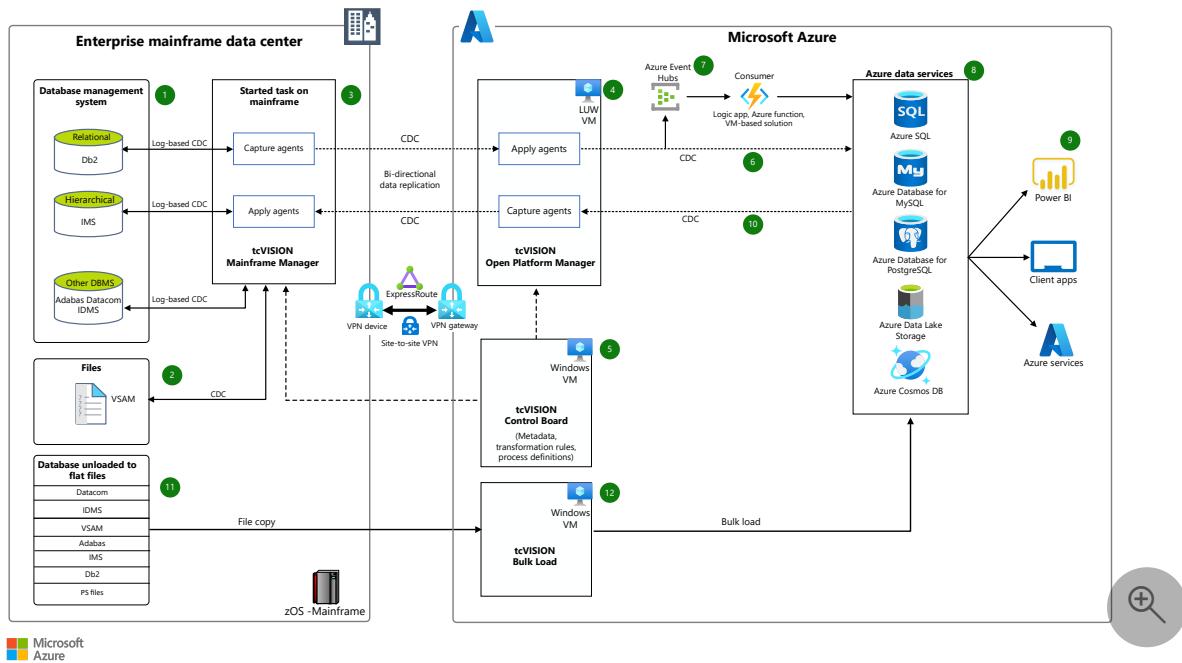
Azure Logic Apps

Azure SQL Database

Azure Storage

tcVISION is a data replication solution from BOS Software. It provides an IBM mainframe integration solution for mainframe data replication, data synchronization, data migration, and change data capture (CDC) to multiple Azure data platform services.

## Architecture



Download a [Visio file](#) of this architecture.

The tcVISION logo is a trademark of its respective company. No endorsement is implied by the use of this mark.

## Workflow

1. The tcVISION data replication solution supports CDC from many mainframe-based databases. Examples include Db2, IMS/DB, Software AG ADABAS, CA Datacom, CA IDMS, and so on. tcVISION provides log-based CDC agents to capture the change

data on the record level. This log-based CDC puts negligible overhead on production source databases.

2. tcVISION also supports CDC from virtual storage access method (VSAM) files.
  3. A task starts on the mainframe. Started tasks (STCs) are created on the mainframe as part of tcVISION software installation. Two vital STCs are:
    - Capture agent, which captures changed data from the source.
    - Apply agent, which uses DBMS-specific APIs for efficiently writing changed data to the target.
- (!) Note**

For Db2 z/OS, tcVISION also offers an Agentless CDC solution by way of a Db2 user-defined type (UDT) that doesn't need a started task.
4. The Open Platform Manager acts as a replication server. This server contains utilities for automatic data mapping to generate metadata for sources and targets. It contains the rule set for extracting the data from the source. And the server transforms and processes the data for the target systems and writes the data into the targets. You can install this component in Linux, Unix, and Windows Operating System.
  5. The tcVISION Control Board or dashboard provides administration, review, operation, control, and monitoring of the data exchange processes. The tcVISION command line utilities help automate data exchange processes and manage the unattended operations of the data synchronization process.
  6. The tcVISION Apply Agent uses database management system-specific APIs. These APIs efficiently implement real-time data changes in combination with CDC technology at the source to the target Azure Data Services (as in, database and files).
  7. tcVISION supports direct streaming of the changed data into Azure Event Hubs or Kafka. These events are then processed by Azure Logic Apps, a function, or a custom solution in the virtual machine.
  8. The Azure data platform targets supported by tcVISION include Azure SQL Database, Azure Database for PostgreSQL, and MySQL, Azure Cosmos DB, Azure Data Lake Storage, and so on.

9. After data lands in Azure data platform, it's consumed by Azure services or others that are permitted to see it. Examples include Power BI, Synapse Analytics, or even a custom application.
10. The tcVISION product can even reverse-sync capture changes from an Azure database platform (Azure SQL Database, MySQL, PostgreSQL, Azure Data Lake Storage, and so on) and write them back to the mainframe data tier.
11. The mainframe database backup and unload files are copied to an Azure VM with tcVISION for bulk load processing.
12. The tcVISION bulk load performs an initial target database load using mainframe source data. Source data can be read directly from the mainframe data store or from a mainframe backup or unload. The bulk load provides an automatic translation of mainframe data types, such as extended binary coded decimal interchange code (EBCDIC)-packed fields. Typically, for the best performance, use the backup or unload data versus a direct read of the mainframe database. The reason for not reading directly is because moving unload or backup data to the requisite tcVISION Azure VM and using native database loaders minimizes network IO and reduces load time.

## Components

The solution uses the following components.

### Networking and identity components

- [Azure ExpressRoute](#) - ExpressRoute lets you extend your on-premises networks into the Microsoft Cloud over a private connection handled by a connectivity provider. With ExpressRoute, you can establish connections to cloud services, such as Microsoft Azure and Microsoft 365.
- [Azure VPN Gateway](#) - A VPN gateway is a specific type of virtual network gateway that sends encrypted traffic between an Azure virtual network and an on-premises location over the public internet.
- [Microsoft Entra ID](#) - Microsoft Entra ID is an identity and access management service that you can sync with an on-premises directory.

### Application components

- [Azure Logic Apps](#) - Logic Apps helps create and run automated recurring tasks and processes on a schedule. You can call services inside and outside Azure, such as

HTTP or HTTPS endpoints, post messages to Azure services such as Azure Storage and Azure Service Bus, or upload files to a file share.

- [Azure Functions](#) - Functions lets you run small pieces of code, or functions, without worrying about application infrastructure. With Functions, the cloud infrastructure provides the up-to-date servers you need to keep your application running at scale.
- [Azure Virtual Machines](#) - Azure VMs are on-demand, scalable computing resources that are available with Azure. An Azure VM provides the flexibility of virtualization. But it eliminates the maintenance demands of physical hardware. Azure VMs offer a choice of operating systems, including Windows and Linux.

## Storage components

- [Azure Storage](#) - This offers un-managed storage solutions like Azure Blob, Azure Tables, Azure Files, and Azure Queues. Files can particularly come in handy for reengineered mainframe solutions. This provides an effective add-on with the managed SQL storage.
- [Azure SQL](#) - This is a fully managed PaaS service for SQL Server from Azure. The relational data can be migrated and used efficiently with other Azure services (Azure SQL Managed Instance, Azure SQL VM, Azure Database for PostgreSQL, Azure Database for MariaDB, MySQL, and so on.)
- [Azure Cosmos DB](#) - Azure Cosmos DB is no-SQL offering that you can use to migrate non-tabular data off the mainframe.

## Monitoring components

- [Azure Monitor](#) - Azure Monitor delivers a comprehensive solution for collecting, analyzing, and acting on telemetry from cloud and on-premises environments.
- [Application Insights](#) - Application telemetry is sent to Application Insights for analysis and presentation.
- [Azure Monitor Logs](#) - Azure Monitor Logs is a feature of Azure Monitor that collects and organizes log and performance data from monitored resources. Data from different sources such as platform logs from Azure services, log and performance data from VM agents, and usage and performance data from applications can be consolidated into a single workspace so they can be analyzed together using a sophisticated query language that's capable of quickly analyzing millions of records.
- [Log Analytics](#) - Log queries help you gain the value of the data collected in Azure Monitor Logs. A powerful query language lets you join data from multiple tables, aggregate large sets of data, and perform complex operations with minimal code.

# Considerations

These considerations implement the pillars of the Azure Well-Architected Framework, which is a set of guiding tenets that can be used to improve the quality of a workload. For more information, see [Microsoft Azure Well-Architected Framework](#).

## Reliability

- Set up tcVISION Open Platform Manager (OPM) on Azure virtual machines deployed in separate availability zones to provide high availability. In case of failures, a secondary tcVISION OPM is activated and it communicates its IP address to tcVISION Mainframe Manager. The mainframe starts communicating with the new tcVISION OPM that continues processing at its next logical restart point, using a combination of logical unit of work (LUW) and restart files.
- Design database services in the Azure support zone redundancy to fail over to a secondary node in case of an outage or during a maintenance window.
- Use Azure Monitor and Application Insights on top of Log Analytics to monitor the health of the Azure resource. You can set alerts for proactive management.
- For more information about resiliency in Azure, see [Designing reliable Azure applications](#).

## Scalability

- CDC: Set up tcVISION scaling for CDC processing by running multiple parallel replication streams. First, analyze the files included in logical transactions. These files must be processed together in sequence. The tcVISION CDC process ensures the integrity of each logical transaction, and these files must be processed together. For instance, sets of tables that don't participate in common transactions might be divided into parallel tasks by creating multiple processing scripts.
- Bulk Load: tcVISION can run parallel concurrent bulk load processing simultaneously on a single Azure VM or on multiple Azure VMs giving horizontal scalability. Bulk load large tables faster by splitting the process into multiple tasks, either by arbitrary intervals, or by way of row filtering. Row filtering can use a key, partition key, date, and so on.
- Auto scaling: The SQL Database serverless compute tier provides an auto scaling option based on workload. Other Azure databases can be scaled up and down by using automation to meet the workload demands.
- For more information, see [Autoscaling best practices in Azure](#).

## Security

Security provides assurances against deliberate attacks and the abuse of your valuable data and systems. For more information, see [Overview of the security pillar](#).

- Control authentication and access for tcVISION with Microsoft Entra ID.
- Encrypt data transfer between tcVISION products (mainframe to Azure) with transport layer security (TLS).
- Use Express Route or site-to-site VPN for private and efficient connection to Azure from an on-premises environment.
- Authenticate Azure resources with Microsoft Entra ID and manage permissions with role-based access control.
- Use the database services in Azure to support various security options like data encryption at rest (TDE), data encryption in transit (TLS), and data encryption while processing, as in, always encrypted.
- For guidelines about designing secure solutions, see the [Azure security documentation](#).
- To find out your security baseline, see [Azure security baseline assessment](#).

## Cost optimization

Cost optimization is about looking at ways to reduce unnecessary expenses and improve operational efficiencies. For more information, see [Overview of the cost optimization pillar](#). Use the [Azure pricing calculator](#) to estimate the cost of implementing this solution.

## Scenario details

Mainframes are servers that process a large number of transactions. Mainframe applications generate and consume large amounts of data daily. With the introduction of public clouds that offer elasticity, cost optimization, ease of use, and easy integration, many x86 and mainframe applications are moving to the cloud. It's now important for organizations to have a well-designed mainframe-to-cloud data integration and migration strategy.

This scenario shows how to integrate an IBM Z (mainframe) data tier with Azure cloud data platform. To integrate mainframe with Azure cloud data platform, use [tcVISION](#) software provided by [B.O.S. Software](#) and [Treehouse Software](#).

## Potential use cases

This solution is ideal for large-scale data migrations to Azure data platform. Consider this scenario for the following use cases:

- Full migration of a mainframe data tier: In this use case, a customer wants to move all their Db2, IMS, IDMS, files, and so on from a mainframe to the Azure data platform.
- Co-existence of mainframe and Azure based applications: In this use case, a customer often has a requirement to support a bidirectional sync between mainframe and Azure data platform.
- Archival: In this use case, a customer wants to store data for audit and compliance purposes but doesn't want to access this data frequently. Azure storage provides a low-cost solution to store archive data.

## Contributors

*This article is maintained by Microsoft. It was originally written by the following contributors.*

Principal authors:

- [Sandip Khandelwal](#) | Senior Engineering Architect

Other contributors:

- [Liz Casey](#) | Senior Content Developer

*To see non-public LinkedIn profiles, sign in to LinkedIn.*

## Next steps

- [Application Insights](#)
- [Architect a data platform in Azure](#)
- [Microsoft Entra ID](#)
- [Azure Artifacts](#)
- [Azure Boards](#)
- [Azure Cosmos DB](#)
- [Azure Data Engineering for mainframe and midrange modernization.](#)
- [Azure ExpressRoute](#)
- [Azure Functions](#)
- [Azure Logic Apps](#)
- [Azure Monitor](#)
- [Azure Monitor Logs](#)
- [Azure Pipelines](#)
- [Azure Repos](#)
- [Azure SQL](#)

- [Azure Storage ↗](#)
- [Azure Test Plans ↗](#)
- [Azure Virtual Machines ↗](#)
- [Azure VPN Gateway ↗](#)
- [Data migration guide ↗](#)
- [Data platform modernization](#)
- [Design your data migration to Azure](#)
- [Log Analytics](#)

## Related resources

- [Modernize mainframe and midrange data](#)
- [Replicate and synch mainframe data in Azure](#)

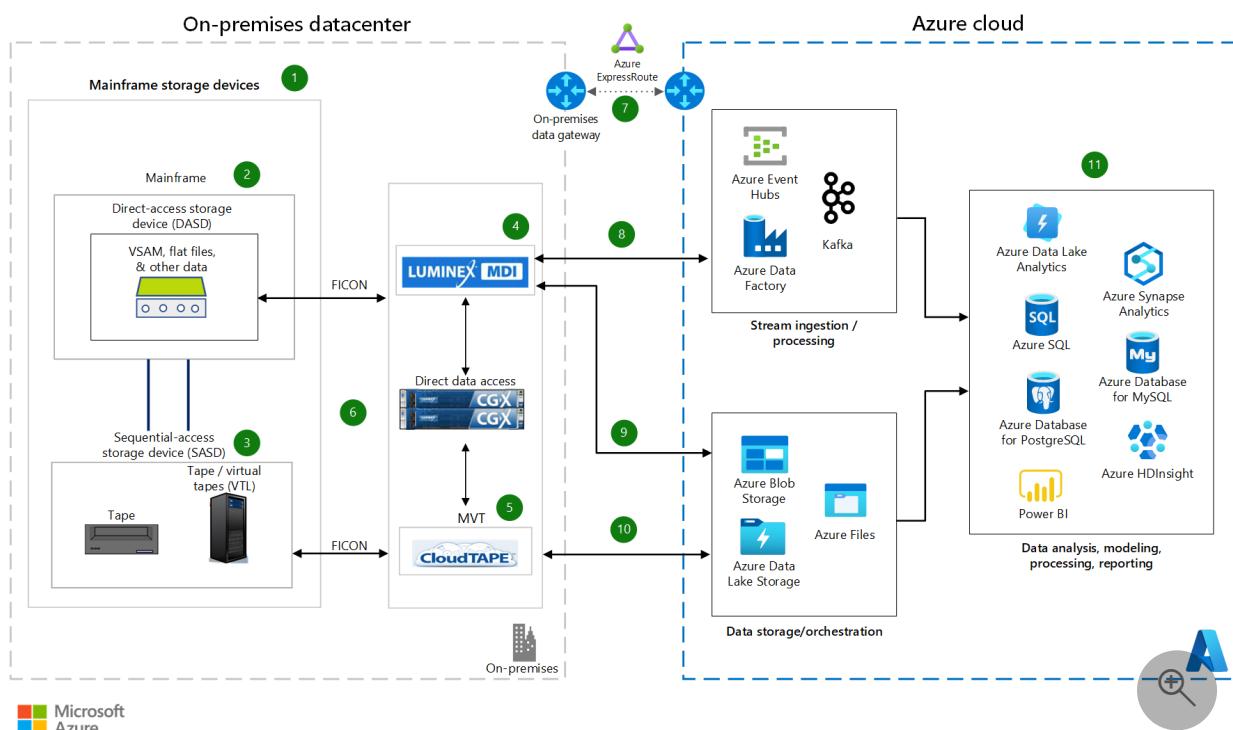
# Mainframe file and tape backup to Azure using Luminex

Azure Event Hubs   Azure ExpressRoute   Azure SQL Database   Azure Storage   Power BI

This article presents a solution for using Luminex products to transfer mainframe data to and from Azure to meet backup, archival, and other business needs. Key components in the solution include the Luminex mainframe data integration (MDI) platform's Cloud Data Sharing and the Luminex mainframe virtual tape (MVT) platform's CloudTAPE.

*Apache® and [Apache Kafka](#) are either registered trademarks or trademarks of the Apache Software Foundation in the United States and/or other countries. No endorsement by the Apache Software Foundation is implied by the use of these marks.*

## Architecture



Download a [Visio file](#) of this architecture.

## Dataflow

1. On a mainframe, secondary storage devices include direct-access storage devices (DASDs) and sequential-access storage devices (SASDs).

2. DASDs are mounted on the mainframe.
3. A tape is a type of SASD that's attached to the mainframe as external storage.
4. The MDI platform sends information that can be stored on files to Azure. Examples include system management facilities (SMF) data, Virtual Storage Access Method (VSAM) files, sequential files, and generation data groups (GDGs). MDI hardware that's installed in the datacenter includes Luminex Channel Gateway X (CGX) controllers and Luminex MDI servers.
5. MVT CloudTAPE provides tape archival and backup. MVT hardware that's installed in the datacenter includes Luminex CGX controllers and CloudTAPE servers.
6. MDI and MVT use CGX controller devices that are based on the Fibre Connection (FICON) protocol. These devices connect directly to the mainframe. No System z Integrated Information Processor (zIIP) specialty engines are needed for data transfer. There's no Luminex agent on the mainframe, and no TCP/IP ports need to be open for communication between the mainframe and Luminex devices.
7. The mainframe data is transferred to Azure through a private, secure Azure ExpressRoute connection.
8. Luminex MDI zKonnect and other services stream the file data for big data analysis on Azure. For instance, system data like mainframe logs and SMF data is streamed to Azure Event Hubs. Azure services ingest the data and then process, transform, and project it.
9. MDI uses Luminex CGX devices to process, transfer, and cache file data. Two options are available:
  - Job control language (JCL) statements are submitted. Luminex provides statements that specify information about input files, the Azure destination, keys and security information, data transformation, and cloud file formats. Organizations that use the Luminex procedure for data transfer can use their own JCL statements. When the job finishes, a return code of zero indicates a successful transfer.
  - The job is monitored from the MDI UI. An operations team can use a combination of the scheduler, the mainframe, and the MDI UI to monitor and troubleshoot jobs. The MDI UI provides information like the job name, the job ID, the user or group, the start time, and the elapsed time. MDI retry mechanisms engage if the file transfer doesn't initially succeed.

The job can be configured to cache the files in local storage before the transfer. After the transfer finishes, that local storage is removed.

10. MVT CloudTAPE sends mainframe tape data to Azure data stores like Azure Blob Storage, Azure Files, and Azure Data Lake Storage. The data can be structured and unstructured. The transfer doesn't use JCL statements. Instead, MVT CloudTAPE moves or replicates mainframe tapes in IBM 3490 or 3590 format that CGX controllers emulate.
11. Azure services provide data processing, storage, analytics, and visualization capabilities.

## Components

- [ExpressRoute](#) extends on-premises networks to the Microsoft cloud. ExpressRoute uses a connectivity provider to establish private connections between on-premises data and Microsoft cloud services.
- [Azure Files](#) is a service that's part of [Azure Storage](#). Azure Files offers fully managed file shares in the cloud. Azure file shares are accessible via the industry-standard Server Message Block (SMB) protocol. This solution uses Luminex MDI and MVT to transfer mainframe files to Azure Files.
- [Blob Storage](#) is a service that's part of Storage. Blob Storage provides optimized cloud object storage for massive amounts of unstructured data. In this solution, Blob Storage provides a way to archive hot and cold mainstream data.
- In this solution, Luminex products can transfer mainframe data to several Azure databases:
  - [Azure SQL](#) is a family of Azure databases that are powered by the SQL Server engine.
  - [Azure SQL Database](#) is a fully managed platform as a service (PaaS) database engine that's part of the Azure SQL family. With AI-powered, automated features, SQL Database handles database management functions like upgrading, patching, backups, and monitoring.
  - [Azure Database for PostgreSQL](#) is a fully managed relational database service that's based on the community edition of the open-source PostgreSQL database engine.
  - [Azure Database for MySQL](#) is a fully managed relational database service that's based on the community edition of the open-source MySQL database engine.
- [Event Hubs](#) is a fully managed big data streaming platform. In this solution, [Luminex zKonnect](#) streams mainframe data to Event Hubs in near real time. Event Hubs provides an endpoint that's compatible with Apache Kafka producer

and consumer APIs. Most existing Apache Kafka client applications use these APIs as an alternative to running their own Apache Kafka clusters.

- [Power BI](#) is a collection of software services and apps that display analytics information. This solution uses mainframe data that comes from various sources and has varying structures. Power BI is used to turn the data into coherent, visually immersive, and interactive insights.
- [Data Lake Storage](#) provides a way to perform big data analytics with low-cost, tiered storage and high throughput.

## Alternatives

- Instead of using third-party solutions for data transfer, you can use a Microsoft solution. For information about transferring data from mainframe and midrange systems to Azure, see [Move archive data from mainframe systems to Azure](#). For information about specific Microsoft solutions, see the following resources:
  - [Copy files from a mainframe to Azure by using the Azure Data Factory FTP connector](#)
  - [Transfer mainframe files to Azure by using SFTP](#)
  - [Transfer a mainframe dataset to Azure Blob Storage by using a mainframe JCL](#)
- To address any latency, connectivity, technological, and regulatory considerations, you can transfer data to Azure Stack instead of to Azure. Azure Stack Hub offers a set of cloud storage services. For more information, see [Azure Stack Hub storage: Differences and considerations](#).
- You can also use Luminex MVT and CGX devices for IBM z/VM and z/VSE mainframes.
- When you transfer tapes to Azure, you can compress and encrypt them to help transmit data safely at all stages. You can easily configure this functionality.
- You can also use this solution for bidirectional data interchange. You can recall the tape data to the mainframe and transform it into its original form.
  - With MDI, the process is similar to the transfer to Azure. You submit JCL statements that provide the specifics of the reverse transfer. The data can be transferred as tapes or as sequential files. The JCL configuration specifies the format.
  - With MVT CloudTAPE, the data is automatically recalled if you request it from the mainframe.

- Luminex CGX devices also support [Enterprise Systems Connection \(ESCON\)](#) ↗ channel connectivity. The existing mainframe backup software sees the channel gateway as a recognized mainframe tape device. As a result, no software change is needed.
- This solution uses ExpressRoute to transfer data from the datacenter to Azure. We recommend this approach, but you can also use the internet for data transfer.

## Scenario details

Mainframe physical storage can be located on the mainframe processor, or it can be external to the mainframe. Processor storage, which is like memory for the mainframe, is located on the processor. Disk drives and tape drives are examples of external storage. Datasets in storage are organized into various logical record and block structures. Parameters like the dataset organization (DSORG) and record format (RECFM) define these data structures. Records in the dataset can be fixed or variable in length, and they can be stored in binary or text format.

Secondary storage devices like [DASDs](#) ↗ and [SASDs](#) ↗ store data that's either frequently or infrequently accessed.

- DASDs are used for immediate data location and retrieval. With direct access, you can read or write data by going directly to a specific physical location on the device. As a result, DASDs are fast and efficient.
- SASDs, such as tapes, are inherently slower than DASDs. To access tape data, you start at one location and then go through successive locations until you find the data that you need. Mainframes use physical tapes and [virtual tape libraries \(VTLs\)](#) ↗, which are also called *virtual tapes*. Currently, virtual tapes are preferred over physical tapes.

The type of storage that you use depends on your needs. Many organizations need cold storage for compliance, regulatory, reporting, audit, or other purposes. Some organizations have data retention policies that require you to store data for nearly 100 years. Examples of this type of data include copies of prescriptions, patient records, customer reward history, and other information. Data that you store for the long term is mostly high in volume and accessed infrequently. Long-term storage generally costs less than active storage, which you typically access multiple times a day and which is frequently updated. Security considerations also affect your choice of storage. Cyberattacks are a constant threat.

Azure offers various storage solutions and is a proven landing place for your storage, backup, and long-term archival needs. You can use cold storage for infrequently

accessed data and hot storage for frequently accessed data. Mainframe file structures, such as VSAM datasets, flat files, and tape data, map to Azure data constructs within databases, structured files, and blob storage. Azure storage can store volume-intense data with cost efficiency, scalability, replication, and self-sustainability. Azure services can also help you retrieve your data, visualize your data, and gain insights from your data.

The solution in this article uses the [Luminex MDI](#) and MVT platforms to transfer mainframe data to and from Azure to meet backup, archival, and other business needs.

- [Luminex MDI](#) is a data transfer and coprocessing platform. MDI uses Luminex CGX devices to process, transfer, and cache mainframe files. MDI provides secure and efficient exchange of data and workload sharing between z/OS mainframes and distributed systems. By using MDI products like Cloud Data Sharing, Big Data Transfer, and zKonnect, you can move files to Azure for backup, archival, data normalization, merging, and analysis. You can configure the transferred data to arrive in ASCII or EBCDIC format in Azure. [MDI Cloud Data Sharing](#) provides a way to migrate mainframe files like VSAM files, sequential files, and GDGs to Azure. MDI also supports integration with Azure messaging services. Applications that are hosted on Azure can use the mainframe files that are stored on Azure for modernization, reduced latency, and improved performance.
- [Luminex MVT](#) is a tape archival and backup platform. MVT uses Luminex CGX control unit software that emulates mainframe 3490 and 3590 tape drives, so you can use existing tape applications without change. The CGX environment provides a suite of products for tape encryption, vaulting, migration, replication, retrieval, disaster recovery, and high availability. Specifically, the [CloudTAPE](#) product provides a way to migrate tape data to Azure.

MDI and MVT both use high-speed CGX controller devices to connect directly to the mainframe. These controllers are based on [FICON](#), a transport protocol that mainframe servers and attached enterprise-class storage controllers support. FICON uses Fibre Channel as the underlying transport protocol. The CGX controllers also take advantage of network attached storage (NAS) and internal storage systems to supply the high levels of performance, scalability, reliability, security, and availability that enterprises demand. With FICON transport, I/O can be shared across multiple systems. FICON delivers optimal protocol efficiency. It also helps provide data integrity and security, even with increased distances between server and storage devices.

With MDI and MVT, no zIIP specialty engines are needed for data transfer, and no TCP/IP ports need to be open to enable communication between the mainframe and Luminex devices. You plug the Luminex CGX devices directly into the mainframe just like

any other mainframe storage device. If necessary, your existing legacy backup and tape management software can run in parallel. For MVT CloudTAPE and MDI Cloud Data Sharing, the millions of instructions per second (MIPS) consumption is minimal because the transfer uses lightweight processes.

## Potential use cases

Many scenarios can benefit from this solution. Possibilities include organizations with the following goals:

- Minimizing tape management and maintenance efforts.
- Modernizing legacy workloads.
- Finding backup and archival solutions.
- Extending their mainframe modernization by moving mainframe tapes to the cloud. Organizations might have this goal if they want to downsize their datacenter but not abandon it. If an organization doesn't use mainframe tapes heavily, the tapes might be a suitable candidate for migration.
- Transforming migrated data into a different format for cloud storage, such as converting EBCDIC data to ASCII, VSAM files to JSON, and sequential data to CSV format.
- Transferring tape metadata to Azure storage metadata.
- Providing new and refactored applications that are hosted on Azure with easy access to data.
- Expanding their cloud footprint.
- Easily monitoring, displaying, and reporting on mainframe files and tape data, and integrating this data with Azure services.
- Monetizing current and historical unlocked mainframe data and using it in cloud business intelligence and analytics tools.

If you're implementing a similar solution and want to share your experiences or feedback, contact the [Microsoft Legacy Modernization Azure Core Engineering \(ACE\) team](#).

## Considerations

These considerations implement the pillars of the Azure Well-Architected Framework, which is a set of guiding tenets that can be used to improve the quality of a workload. For more information, see [Microsoft Azure Well-Architected Framework](#).

## Reliability

Reliability ensures your application can meet the commitments you make to your customers. For more information, see [Overview of the reliability pillar](#).

- You can deploy this solution in multiple regions, and you can implement geo-replication in the data layer. Azure auto-failover groups also help provide data protection.
- Clustered CGX controllers can provide an active-active recovery solution during a failure.
- [MVT Synchronous Tape Matrix](#) provides reliability across multiple datacenters. Its infrastructure adjusts to failures without interruption.
- [Luminex Replication](#) can replicate data to one or many targets. A target can be one or more disaster recovery sites that each have a mainframe and CGX controller installed on the property. You can also preconfigure a target through Azure geo-replication. If you use Azure and other private or public clouds, you can also use a hybrid strategy for disaster recovery. Essentially, you can use the replication strategy that best meets your requirements. Examples include one-to-one, one-to-many, many-to-many, and cascading strategies.

## Security

Security provides assurances against deliberate attacks and the abuse of your valuable data and systems. For more information, see [Overview of the security pillar](#).

- The fully managed storage in this solution eliminates issues that are related to physical media safety. Examples are damage or unauthorized access that might occur when you ship physical tapes in vehicles.
- [Luminex CGSafe](#) provides tape compression and encryption. This product is part of the MVT family and is included with CloudTAPE. CGSafe encrypts and compresses tapes during ingestion, at rest, and in transit.
- When you use MDI Cloud Data Sharing, files are sent over HTTPS by using SSL. In Azure, you can encrypt the files at rest.
- Because the solution uses FICON and ESCON connectivity, you don't need to open any ports for data transfer.

## Cost optimization

Cost optimization is about looking at ways to reduce unnecessary expenses and improve operational efficiencies. For more information, see [Overview of the cost optimization pillar](#).

- Pay-as-you-go pricing and multi-tiered models in Azure provide options to suit various cost and performance needs. For instance, if you access data infrequently, the Azure cool access tier is a good option for low-cost storage.
- The pricing of this solution depends on your volume of tape data, your datacenter location, and your bandwidth. The cost also depends on which Azure services you use. These factors determine the hardware that you use, such as the number of Luminex CGX controllers. The factors also affect your software, service, licensing, and support costs.
- The data interchange doesn't require zIIP processors. As a result, you save on costs when you run the software.
- After the Luminex infrastructure is in place, you can use the Luminex hardware for other purposes. For instance, you might already use MDI Cloud Data Sharing for file transfer. If you augment your environment with MDI zKonnect for streaming, you can save on costs because you can purchase additional Luminex software and infrastructure at a significantly reduced price.
- If you already have an ExpressRoute infrastructure in place, you can use it for this solution.
- Using Azure and Luminex for backup and recovery helps you eliminate some costs that are associated with physical tape infrastructure. Examples include media and shipping expenses and off-site storage for vaulting.

## Operational excellence

Operational excellence covers the operations processes that deploy an application and keep it running in production. For more information, see [Overview of the operational excellence pillar](#).

- The data transfer to Azure in this solution gives you flexibility when you develop a backup strategy. You can enable automated, regular migration or phased-data migration. After you've installed a Luminex device in your datacenter, you can configure unidirectional or bidirectional communication, staged migration, or one-time migration. This flexibility provides support for implementing DevOps and Agile working principles and for immediate cloud adoption.
- You can take advantage of Azure capabilities for mainframe backup, archive, and disaster recovery.
- You can deploy continuous integration/continuous delivery (CI/CD) pipelines on Azure to manage data movement, transformation, and control activities.

## Performance efficiency

Performance efficiency is the ability of your workload to scale to meet the demands placed on it by users in an efficient manner. For more information, see [Performance efficiency pillar overview](#).

- If you have a high volume of data, you can cluster CGX controllers. Typically, one CGX device offers a data-transfer speed of up to 800 megabytes per second (MB/s). CGX controllers are available with up to four Fibre Channel ports or 1 Gigabit Ethernet (GbE), 10 GbE, or 25 GbE. These controllers also offer up to four ports for connectivity to attached storage systems.
- In Azure services, various performance options and tiers are available. For instance, block blob storage accounts offer standard and premium performance tiers. You can choose the tier that best meets your needs.
- Predefined access and life cycle management in Azure make it easy to optimize the performance of specific use cases.
- The tape emulation software in this solution uses the FICON I/O system. By using this system, you can reduce CPU time, increase data transmission speed, and reduce elapsed time.

## Contributors

*This article is maintained by Microsoft. It was originally written by the following contributors.*

Principal authors:

- [Daniel Saunders](#) | Sales Engineer
- [Bhuvi Vatsey](#) | Senior Technical Program Manager

*To see non-public LinkedIn profiles, sign in to LinkedIn.*

## Next steps

- For more information, contact the [Microsoft Legacy Modernization Azure Core Engineering \(ACE\) team](#).
- For information about third-party data transfer solutions, see [Third-party archive solutions](#).

## Related resources

- [Modernize mainframe and midrange data](#)
- [Move archive data from mainframe systems to Azure](#)

- Replicate mainframe data by using Precisely Connect
- Mainframe and midrange data replication to Azure using Qlik
- Mainframe and midrange data replication to Azure using tcVISION
- Migrate a mainframe data tier to Azure by using mLogica LIBER\*IRIS
- Mainframe modernization using Model9

# Migrate mainframe data tier to Azure with mLogica LIBER\*IRIS

Azure Database for MySQL   Azure Database for PostgreSQL   Azure Cosmos DB   Azure SQL Database

Azure Storage

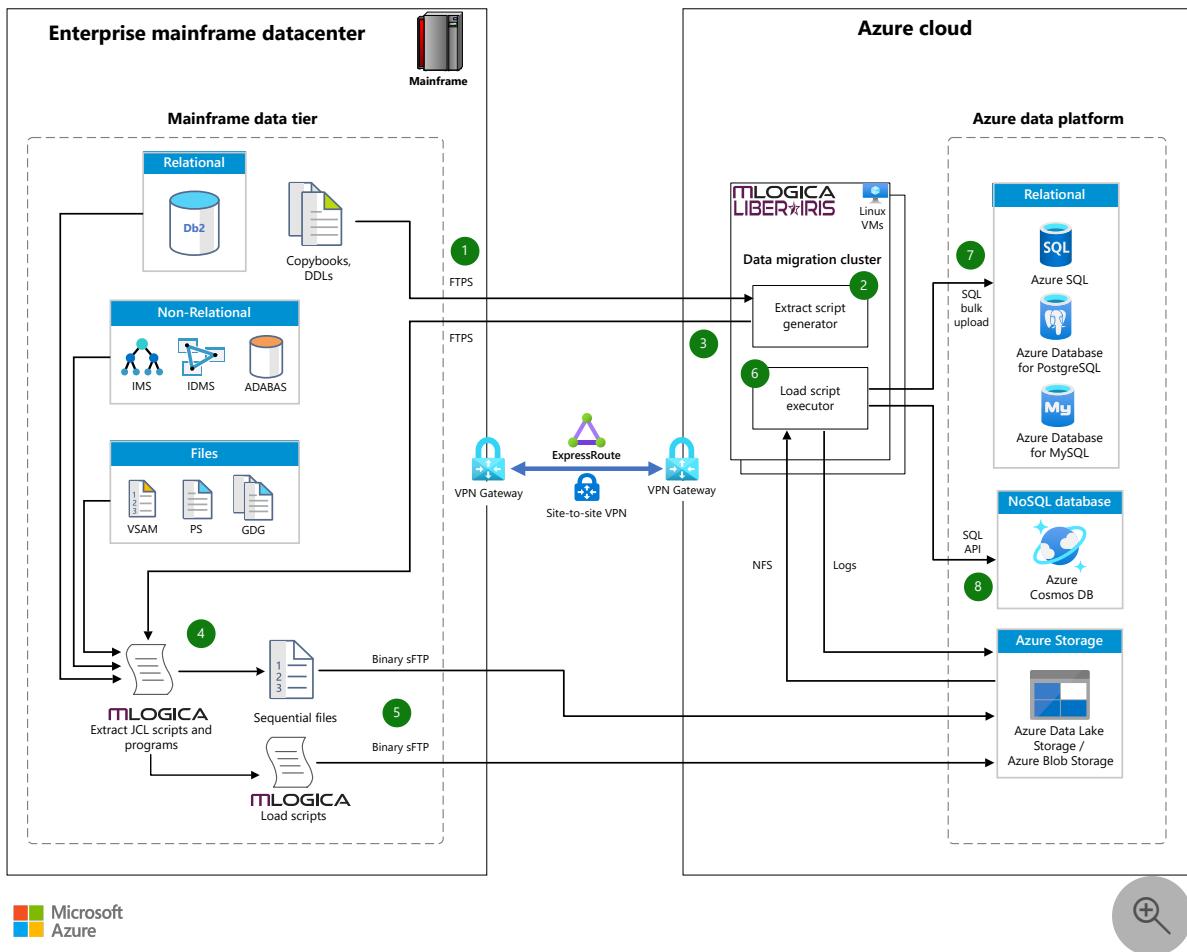
The high volume of transactions for mainframe applications creates a large volume of data. Azure offers a compelling target for mainframe modernization and data migration. Azure relational and NoSQL databases provide scalability, high availability, and ease of maintenance that meets or exceeds that of mainframe environments. If you want to retire a mainframe workload and retain the data in a low-cost storage, Azure provides options.

Migrating workloads from mainframe to Azure as a part of application replatforming or refactoring typically requires data migration at scale. [mLogica's LIBER\\*IRIS](#) provides a proven solution for bulk data migration from a mainframe to Azure. The solution operates at scale for migrating enterprise workloads. This article shows how to migrate IBM z/OS mainframe data with high fidelity to Azure.

*mLogica LIBER\*IRIS and its logos are trademarks of its company. No endorsement is implied by the use of these marks.*

## Architecture

The following diagram shows how mLogica LIBER\*IRIS integrates with Azure components to migrate mainframe data to Azure at scale.



Download a [Visio file](#) of this architecture.

## Workflow

The steps to migrate mainframe data to Azure are as follows:

1. Copy data definition language (DDL) files, database description (DBD) files, copybooks, data layouts, and other data description artifacts to an Azure Linux virtual machine configured with the mLogica data migration service tools using FTPS over a secure Azure site-to-site virtual private network (VPN) or Azure ExpressRoute.
2. The mLogica Liber\*IRIS data migration cluster generates data extraction scripts to run on the mainframe.
3. Use FTPS over the VPN to transfer the data extraction scripts to the mainframe. The FTPS connection converts ASCII to the mainframe EBCDIC format.
4. The extracted scripts run on the mainframe. They export data from multiple sources into *sequential files*, where all packed decimal data is unpacked. They generate the SQL *load scripts* used to load the data into the target database.
5. The sequential files and load scripts are transferred by using binary SFTP to Azure Blob Storage. Mainframe data is still in EBCDIC format at this point.

6. The mLogica data migration service runs the load scripts to convert EBCDIC to ASCII. The scripts write errors during load to Azure Storage. To reduce costs, you can use two storage accounts: store data files on a hot access tier and log files on a cold access tier.
7. The scripts load the ASCII converted data from sequential files into the target Azure relational database. The load scripts include DDL commands to create tables and other objects and SQL queries to load the data into those objects. Scale the load process horizontally across a cluster to maximize throughput, as needed. Execution logs and detailed exception logs are stored in Azure Blob Storage for further analysis.
8. The mLogica Liber\*IRIS data migration service runs the load scripts to transform data from relational file format to NoSQL database format. You can load this NoSQL data to Azure Cosmos DB by using the Azure Cosmos DB SQL API.

## Components

- Networking and Identity
  - [Azure ExpressRoute](#) lets you extend your on-premises networks into Azure over a private connection by using a connectivity provider.
  - [Azure VPN Gateway](#) is a virtual network gateway used to send encrypted traffic between an Azure virtual network and an on-premises location over the internet.
  - [Microsoft Entra ID](#) is an identity and access management service that can be synchronized with an on-premises directory.
- Application
  - [Azure Virtual Machines](#) provides on-demand, scalable computing resources. The mLogica data migration cluster runs on Azure Linux virtual machines optimized for network performance.
- Storage
  - [Azure Blob Storage](#) offers a highly available, encrypted-at-rest, cost-efficient, high-capacity storage facility. It enables direct binary SFTP traffic from the mainframe. Blob Storage can mount containers on Linux virtual machines by using NFS.
  - [Azure SQL](#), [Azure Database for PostgreSQL](#), and [Azure Database for MySQL](#) are fully managed platform as a service (PaaS) services for SQL Server, PostgreSQL, and MySQL. They provide high-performance, highly available options for mainframe relational data, emulated non-relational data, and emulated Virtual Storage Access Method (VSAM) data.

- [Azure Cosmos DB](#) is an Azure NoSQL database. Use it to migrate non-relational mainframe sources like Information Management System (IMS), Integrated Database Management System (IDMS), and adaptable database system (ADABAS).
- Monitoring
  - [Azure Monitor](#) delivers a comprehensive solution for collecting, analyzing, and acting on telemetry from cloud and on-premises environments.
  - [Application Insight](#) receives application telemetry to analyze and present.
  - [Azure Monitor Logs](#) is a feature of Azure Monitor that collects and organizes log and performance data from monitored resources. This feature can consolidate data from multiple sources into a single workspace. These sources include platform logs from Azure services, log and performance data from virtual machine agents, and usage and performance data from applications. Analyze these sources together by using a sophisticated query language, which is capable of quickly analyzing millions of records.
  - [Log Analytics](#) is a feature of Azure Monitor. Log queries help you use the data collected in Azure Monitor Logs and mLogica load script execution logs, which are stored in Blob Storage. A powerful query language allows you to join data from multiple tables, aggregate large sets of data, and perform complex operations.

## Potential use cases

There are two key use cases for this example workload:

- Workload replatforming or refactoring

Move all mainframe data related to the workload from a mainframe to Azure. This data includes databases, like Db2, IMS, and IDMS, and files.

- Archival

Retire mainframe workload and retain the data in a low-cost Azure storage solution.

## Recommendations

Follow these general recommendations unless you have a specific requirement that overrides them:

- To reduce network latency, create all Azure resources mentioned in this scenario in one region.

- Instead of sending a single large file from the mainframe to Azure, split data into multiple files and send them in parallel.

# Considerations

These considerations implement the pillars of the Azure Well-Architected Framework, which is a set of guiding tenets that can be used to improve the quality of a workload. For more information, see [Microsoft Azure Well-Architected Framework](#).

## Reliability

Reliability ensures your application can meet the commitments you make to your customers. For more information, see [Overview of the reliability pillar](#).

## Resiliency

Use Azure Monitor and [Application Insights](#) to monitor the mLogica data migration cluster. Set up alerts for proactive management.

For more information about resiliency in Azure, see [Designing reliable Azure applications](#).

## Availability

This example workflow describes mainframe-to-Azure data migration to replatform, refactor, or archive a workload. This task is discrete, performed a few times during a month-long project. Although high availability isn't required in this scenario, you can design the mLogica data migration cluster to provide high availability.

Azure database services support zone redundancy. You can configure them to fail over if there's an outage or during a maintenance window.

## Security

Security provides assurances against deliberate attacks and the abuse of your valuable data and systems. For more information, see [Overview of the security pillar](#). For general guidance on designing secure solutions, see the [Azure security documentation](#).

Database services in Azure support various security options:

- Data encryption at rest using [transparent data encryption](#)

- Data encryption in transit using [TLS](#)
- Data encryption while processing using [Always Encrypted with secure enclaves](#)

You can control authentication and access control on the mLogica data migration cluster by using Microsoft Entra ID. You can configure Azure resources for authentication and authorization using Microsoft Entra ID and role-based access control.

Data transferred between the mLogica data migration cluster and the mainframe is encrypted in transit by using TLS. TLS certificates can be stored in [Azure Key Vault](#) for enhanced security. Data transferred from the mainframe to Azure Blob Storage is encrypted in transit using SSH.

The mainframe data and load scripts are temporarily stored in Azure Blob Storage. They're encrypted at rest. Data is deleted from Azure Blob Storage after the migration is complete.

This example workflow uses Azure ExpressRoute or [site-to-site VPN](#) for a private and efficient connection to Azure from your on-premises environment.

## Cost optimization

Cost optimization is about looking at ways to reduce unnecessary expenses and improve operational efficiencies. For more information, see [Overview of the cost optimization pillar](#).

Here are some cost optimization possibilities:

- [Azure SQL Database serverless](#) automatically scales, pauses, and resumes compute resources based upon your workload activity, so that you only pay for the resources you consume.
- Use lifecycle policy to move data between access tiers in Azure storage.
- In Azure storage, if there's no access for a period of time, move your data from a hotter access tier to a cooler one. You can also move data from a cooler access tier to an archive access tier.
- Use [Azure Advisor](#) to find underused resources. Get recommendations on how to reconfigure or consolidate resources to reduce your spending.

Use the [Azure pricing calculator](#) to estimate the cost of using Azure components for this solution.

## Operational excellence

Operational excellence covers the operations processes that deploy an application and keep it running in production. For more information, see [Overview of the operational excellence pillar](#).

Azure DevOps can be used for re-engineering mainframe applications on Azure during every phase of software development and team collaboration. Azure DevOps provides these services:

- [Azure Boards](#). Agile planning, work item tracking, visualization, and reporting.
- [Azure Pipelines](#). A language, platform, and cloud independent continuous integration/continuous delivery (CI/CD) platform with support for containers or Kubernetes.
- [Azure Repos](#). Cloud-hosted private Git repositories.
- [Azure Artifacts](#). Integrated package management with support for Maven, npm, Python, and NuGet package feeds from public or private sources.
- [Azure Test Plans](#). An integrated planned and exploratory testing solution.

## Performance efficiency

Performance efficiency is the ability of your workload to scale to meet the demands placed on it by users in an efficient manner. For more information, see [Performance efficiency pillar overview](#).

If you're migrating multiple large independent datasets, deploy the mLogica data migration cluster on multiple virtual machines to maximize data loading speed.

You can upload multiple data sets in parallel from the mainframe to Blob Storage.

[Azure SQL DB serverless](#) provides an option for autoscaling based on workload. Other Azure databases can be scaled up and down using automation to meet the workload demands. For more information, see [Autoscaling](#).

## Contributors

*This article is maintained by Microsoft. It was originally written by the following contributor.*

Principal author:

[Sandip Khandelwal](#) | Senior Engineering Architect

*To see non-public LinkedIn profiles, sign in to LinkedIn.*

# Next steps

Review the [Azure Database Migration Guides](#).

For more information, contact [Azure Data Engineering - Mainframe & Midrange Modernization](#).

- [Azure Monitor overview](#)
- [Introduction to Azure Blob Storage](#)
- [mLogica LIBER\\*IRIS ↗](#)
- [Quickstart: Create a Linux virtual machine in the Azure portal](#)
- [Virtual machines in Azure](#)
- [Welcome to Azure Cosmos DB](#)

## Related resources

- [Azure data platform end-to-end](#)
- [Modernize mainframe and midrange data](#)
- [Rehost Adabas & Natural applications in Azure](#)
- [Rehost a general mainframe on Azure](#)

# Modernize mainframe workloads by using Model9

Azure Blob Storage   Azure ExpressRoute   Azure VPN Gateway   Azure Synapse Analytics   Azure Monitor

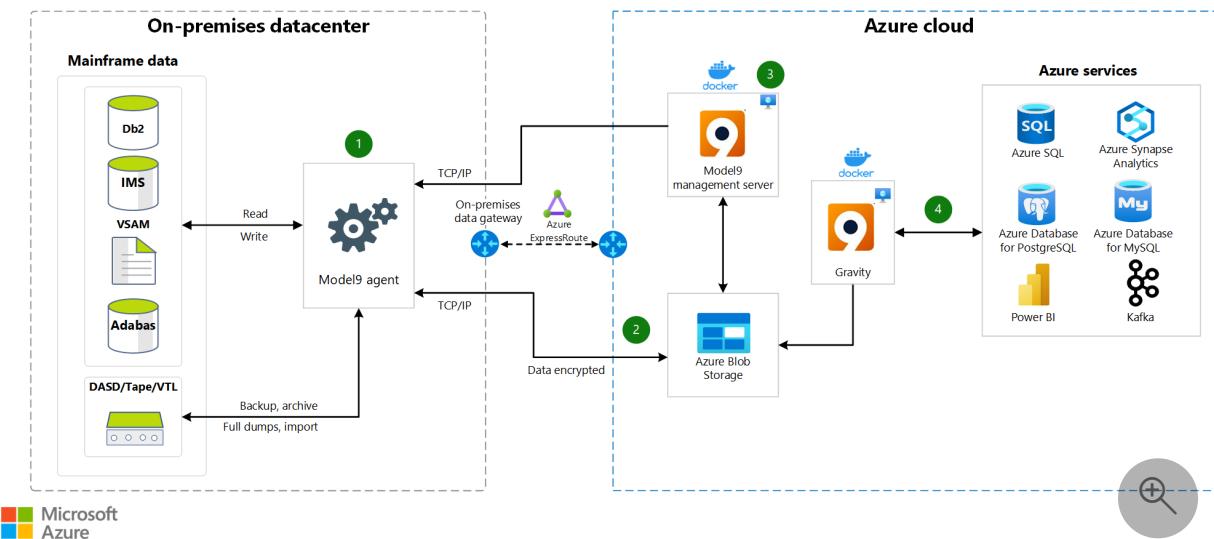
This article describes how to use Model9 Manager to send mainframe data directly to Azure Blob Storage as part of a mainframe modernization migration.

You can use Model9 Shield together with Azure Blob Storage as an alternative to a virtual tape library (VTL) to back up data in a faster and more cost-effective way.

Model9 Gravity transforms mainframe data that's transferred to Azure Blob Storage into open formats that can be used by other Azure services.

*Apache®, [Kafka](#), and the flame logo are either registered trademarks or trademarks of the Apache Software Foundation in the United States and/or other countries. No endorsement by The Apache Software Foundation is implied by the use of these marks.*

## Architecture



Download a [Visio file](#) of this architecture.

## Workflow

1. The Model9 agent is a z/OS started task that sends mainframe data directly to Azure Blob Storage.

2. The Model9 agent sends the data, which is encrypted, to Azure Blob Storage over TCP/IP.
3. Model9 management server manages Model9 policies, activities, and storage.
4. Model9 Gravity transforms mainframe data in Azure Blob Storage into open formats that can be used by Azure services.

## Components

This solution uses the following components.

### Model9 cloud data platform components

The main components of Model9 cloud data platform are:

- **Model9 agent.** A Java-based application that runs as a started task on one or more z/OS logical partitions (LPARs). It reads and writes data directly from and to Azure Blob Storage over TCP/IP. The Model9 agent can run on zIIP engines, which dramatically reduces general CPU consumption.
- **Model9 management server.** A web application that runs in a Docker container. It manages the web UI and the communication with z/OS agents. It provides a way for you to define several types of policies for data protection, migration, and data archival.
- **Lifecycle management engine.** A Java-based application that runs on-premises on a z/OS LPAR and deletes expired data both from the object storage and from z/OS.
- **Data management command-line interface (CLI).** A CLI that runs on z/OS LPAR. You can use it to perform backup, restore, archive, recall, and delete resource-based actions to and from Azure Blob Storage.
- **Gravity.** A Docker-based application that supports the transformation of Model9 managed objects into an open format that gets processed by AI, business intelligence, and machine learning applications. The data can be transformed either to a CSV or JSON file, or directly to Azure Database for SQL.

### Networking and identity

- [Azure ExpressRoute](#) extends your on-premises networks into cloud services that are offered by Microsoft over a private connection from a connectivity provider. With ExpressRoute, you can establish connections to cloud components such as Azure services and Microsoft 365.

- [Azure VPN Gateway](#) is a specific type of virtual network gateway that sends encrypted traffic between Azure Virtual Network and an on-premises location over the public internet.
- [Microsoft Entra ID](#) is an identity and access management service that synchronizes with an on-premises active directory.

## Application

- [Apache Kafka](#) is an open-source distributed event-streaming platform that's used for high-performance data pipelines, streaming analytics, data integration, and business-critical applications.

## Storage

- [Azure SQL Database](#) is part of the Azure SQL family and is built on the cloud. This service offers all the benefits of a fully managed and evergreen platform as a service (PaaS). Azure SQL Database also provides AI-powered, automated features that optimize performance and durability. Serverless compute and Hyperscale storage options automatically scale resources on demand.
- [Azure Database for PostgreSQL](#) is a fully managed relational database service that's based on the community edition of the open-source PostgreSQL database engine. With this service, you can focus on application innovation instead of database management. You can also scale your workload quickly and easily.
- [Azure Database for MySQL](#) is a fully managed relational database service that's based on the community edition of the open-source MySQL database engine.
- [Azure SQL Managed Instance](#) is an intelligent, scalable cloud database service that offers all the benefits of a fully managed and evergreen PaaS. SQL Managed Instance has nearly 100 percent compatibility with the latest SQL Server (Enterprise Edition) database engine. This service also provides a native virtual network implementation that addresses common security concerns.
- [Azure Synapse Analytics](#) is a fast and flexible cloud data warehouse that helps you scale, compute, and store elastically and independently, with a massively parallel processing architecture.
- [Azure Storage](#) is a cloud storage solution that includes object, file, disk, queue, and table storage. Services include hybrid storage solutions and tools for transferring, sharing, and backing up data.

## Analysis and reporting

- [Power BI](#) is a suite of business analytics tools that deliver insights throughout your organization. By using Power BI, you can connect to hundreds of data sources, simplify data preparation, and drive ad hoc analysis. You can produce beautiful reports, then publish them for your organization to consume on the web, and across mobile devices.

## Monitoring

- [Azure Monitor](#) delivers a comprehensive solution for collecting, analyzing, and acting on telemetry from cloud and on-premises environments. It contains the Application Insights, Azure Monitor Logs, and Azure Log Analytics features.
- Model9 supports monitoring the status and results of all activities via the **Activities** page in the Model9 UI. For more information, see [Monitoring Activities](#).

## Alternatives

- Instead of installing the Model9 management server in the cloud on Azure Virtual Network, you can install it on-premises. On a Linux or z/Linux OS, you can also install the management server on z/OS Container Extensions (zCX).
- Model9 data transformation service runs externally to the mainframe in an on-premises environment. This setup saves expensive mainframe resources. You can also deploy on the cloud by using either a server instance or container services.
- ExpressRoute provides a private and efficient connection to Azure from on-premises, but you can also use [site-to-site VPN](#).

## Scenario details

Mainframe data that's stored in physical or virtual tape libraries is critical for customers. As this data grows, its volume can require a significant amount of storage. The data can then become more demanding to maintain on-premises. You can easily migrate this data to Azure storage and use it for AI, business intelligence, machine learning, and analytics applications. Azure storage carries several unique benefits over traditional on-premises storage approaches, and includes data management services, scalability, performance, reliability, and security.

Model9 provides an end-to-end suite of cloud data management solutions for mainframes that solves the migration problem — elegantly, cost effectively, and with no

application changes. Based on a unique, proven technology, Model9 solutions migrate mainframe data by using secure and expedited technology to access the Azure cloud.

Model9 solutions are designed to save expensive mainframe CPU resources by using the mainframe zIIP engines and to connect the mainframe data to Azure cloud storage. Cloud applications can use the migrated data in Azure storage services. This article outlines a solution for migrating mainframe data to the cloud. Besides Model9, the solution's core components include Azure storage and database services.

## Potential use cases

Model9 offers a suite of services that are based on the Model9 cloud data platform. These services are suitable for the following use cases:

- Make mainframe data available to Azure data services, AI, machine learning, analytics, and business intelligence tools.
- Protect mainframe data with backup and archive to Azure Blob Storage.
- Have mainframe applications write and read data directly to and from Blob Storage.
- Provide protection for mainframe data against cyberattacks by creating an immutable third copy in Azure.

## Considerations

These considerations implement the pillars of the Azure Well-Architected Framework, which is a set of guiding tenets that can be used to improve the quality of a workload. For more information, see [Microsoft Azure Well-Architected Framework](#).

## Cost optimization

Cost optimization is about reducing unnecessary expenses and improving operational efficiencies. For more information, see [Overview of the cost optimization pillar](#).

Use the [Azure pricing calculator](#) to estimate the cost of implementing this solution.

## Reliability

Reliability ensures that your application can meet the commitments you make to your customers. For more information, see [Overview of the reliability pillar](#).

- Deploy Model9 cloud data manager in the cloud on Azure Virtual Machines, and on the customer's virtual network for superior availability.
- Deploy an agent in each z/OS LPAR to allow better availability across the systems complex (*sysplex*), or the mainframe cluster.
- Combine the Application Insights and Log Analytics features of Monitor to stay informed about the health of Azure resources.
- For guidance on resiliency in Azure, see [Design reliable Azure applications](#).

## Security

Security provides assurances against deliberate attacks and the abuse of your valuable data and systems. For more information, see [Overview of the security pillar](#).

- Authenticate Azure resources by using Microsoft Entra ID. Manage permissions by using role-based access control (RBAC).
- Model9 uses the z/OS Security Authorization Facility (SAF) for authentication of actions. Traffic between the Model9 agent and Azure Blob Storage is encrypted.
- The security options in Azure database services are:
  - Data encryption at rest.
  - Dynamic data masking.
  - Always-encrypted database.
- For general guidance on designing secure solutions, see [Azure security documentation](#).

## Performance efficiency

Performance efficiency is the ability of your workload to scale to meet the demands placed on it by users in an efficient manner. For more information, see [Performance efficiency pillar overview](#).

- Use multiple agents to increase scalability and throughput on all the LPARs within the same sysplex.
- Use multiple transformation instances behind a load balancer to increase scalability and performance.
- Blob Storage is a scalable system for storing backups, archival data, secondary data files, and other unstructured digital objects.

# Contributors

*This article is maintained by Microsoft. It was originally written by the following contributors.*

Principal author:

- [Seetharaman Sankaran](#) | Senior Engineering Architect

Other contributors:

- [Pratim Dasgupta](#) | Senior Engineering Architect
- [Ashish Khandelwal](#) | Principal Engineering Architect Manager

*To see non-public LinkedIn profiles, sign in to LinkedIn.*

## Next steps

- [Model9 Cloud data management for Mainframe](#)
- [What is Azure ExpressRoute?](#)
- [What is VPN Gateway?](#)
- [Introduction to Azure data Lake Storage Gen2](#)
- [What is Azure SQL Database?](#)
- [What is Azure Synapse Analytics?](#)
- [What is Azure Database for PostgreSQL?](#)
- [What is Azure Database for MySQL?](#)
- [What is Power BI?](#)
- [Azure Monitor overview](#)
- For more information, contact [Mainframe Modernization](#)

## Related resources

- [Modernize mainframe and midrange data](#)
- [Re-engineer mainframe batch applications on Azure](#)
- [Replicate and sync mainframe data in Azure](#)
- [Mainframe access to Azure databases](#)
- [Mainframe file replication and sync on Azure](#)

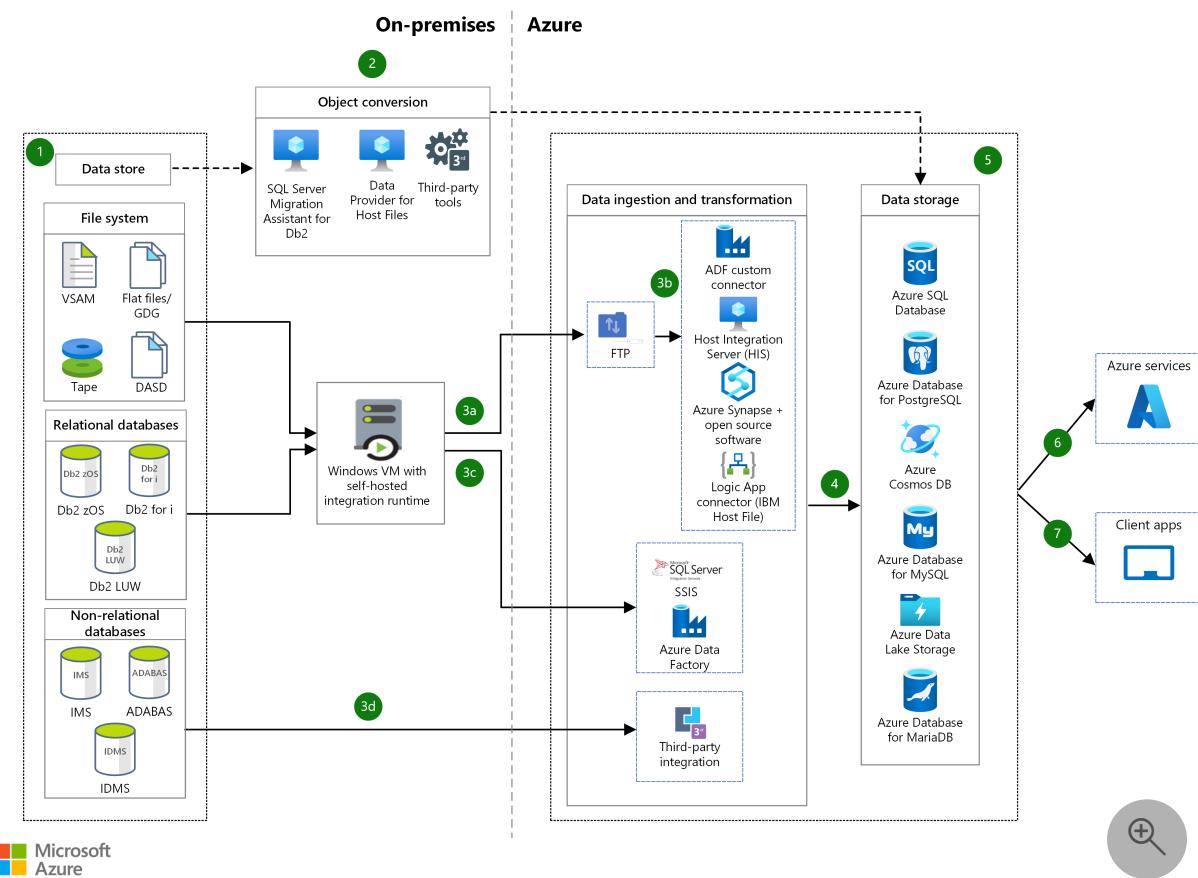
# Modernize mainframe and midrange data

Azure Cosmos DB   Azure Data Lake   Azure SQL Database   Azure SQL Managed Instance   Azure Storage

Apache®, [Spark](#), and the flame logo are either registered trademarks or trademarks of the Apache Software Foundation in the United States and/or other countries. No endorsement by The Apache Software Foundation is implied by the use of these marks.

This article describes an end-to-end modernization plan for mainframe and midrange data sources.

## Architecture



Download a [Visio file](#) of this architecture.

## Dataflow

The following dataflow outlines a process for modernizing a mainframe data tier. It corresponds to the preceding diagram.

1. Mainframe and midrange systems store data in data sources, like file systems (VSAM, flat file, LTFS), relational databases (Db2 for z/OS, Db2 for IBM i, Db2 for Linux UNIX and Windows), or non-relational databases (IMS, ADABAS, IDMS).
2. The object conversion process extracts object definitions from source objects. The definitions are then converted into corresponding objects in the target data store.
  - [SQL Server Migration Assistant](#) (SSMA) for Db2 migrates schemas and data from IBM Db2 databases to Azure databases.
  - [Managed Data Provider for Host Files](#) converts objects by:
    - Parsing COBOL and RPG record layouts, or *copybooks*.
    - Mapping the copybooks to C# objects that .NET applications use.
  - Third-party tools perform automated object conversion on non-relational databases, file systems, and other data stores.
3. Data is ingested and transformed. Mainframe and midrange systems store their file system data in EBCDIC-encoded format in file formats like:
  - Indexed [VSAM](#) files
  - Non-indexed [GDG](#) files
  - Flat files

COBOL, PL/I, and assembly language copybooks define the data structure of these files.

- a. FTP transfers mainframe and midrange file system datasets with single layouts and unpacked fields in binary format and corresponding copybook to Azure.
- b. Data is converted. The Azure Data Factory custom connector is a solution developed by using the Host File client component of Host Integration Server to convert mainframe datasets.

[Host Integration Server](#) integrates existing IBM host systems, programs, messages, and data with Azure applications. Host Integration Server is a Host File client component that you can use to develop a custom solution for dataset conversion.

The Azure Data Factory custom connector is based on the open-source Spark framework, and it runs on [Azure Synapse Analytics](#). Like other solutions, it can parse the copybook and convert data. Manage the service for data conversion by using the [Azure Logic Apps](#) Parse Host File Contents connector.

- c. Relational database data is migrated.

IBM mainframe and midrange systems store data in relational databases like these:

- [Db2 for z/OS ↗](#)
- [Db2 for Linux UNIX and Windows ↗](#)
- [Db2 for IBM i ↗](#)

These services migrate the database data:

- Data Factory uses a Db2 connector to extract and integrate data from the databases.
- SQL Server Integration Services handles various data [ETL ↗](#) tasks.

d. Non-relational database data is migrated.

IBM mainframe and midrange systems store data in non-relational databases like these:

- [IDMS ↗](#), a [network model ↗](#) database management system (DBMS)
- [IMS ↗](#), a [hierarchical model ↗](#) DBMS
- [Adabas ↗](#)
- [Datacom ↗](#)

Third-party products integrate data from these databases.

4. Azure services like Data Factory and [AzCopy](#) load data into Azure databases and Azure data storage. You can also use third-party solutions and custom loading solutions to load data.

5. Azure provides many managed data storage solutions:

- Databases:
  - [Azure SQL Database](#)
  - [Azure Database for PostgreSQL](#)
  - [Azure Cosmos DB](#)
  - [Azure Database for MySQL](#)
  - [Azure Database for MariaDB](#)
  - [Azure SQL Managed Instance](#)
- Storage:
  - [Azure Data Lake Storage](#)
  - [Azure Blob Storage](#)

6. Azure services use the modernized data tier for computing, analytics, storage, and networking.

7. Client applications also use the modernized data tier.

# Components

## Data storage

- [SQL Database](#) is part of the [Azure SQL family](#). It's built for the cloud and provides all the benefits of a fully managed and evergreen platform as a service. SQL Database also provides AI-powered automated features that optimize performance and durability. Serverless compute and [Hyperscale storage options](#) automatically scale resources on demand.
- [Azure Database for PostgreSQL](#) is a fully managed relational database service that's based on the community edition of the open-source [PostgreSQL](#) database engine.
- [Azure Cosmos DB](#) is a globally distributed [multimodel](#) [NoSQL](#) database.
- [Azure Database for MySQL](#) is a fully managed relational database service that's based on the community edition of the open-source [MySQL](#) database engine.
- [Azure Database for MariaDB](#) is a cloud-based relational database service. It's based on the [MariaDB](#) community edition database engine.
- [SQL Managed Instance](#) is an intelligent, scalable cloud database service that offers all the benefits of a fully managed and evergreen platform as a service. SQL Managed Instance has near-100% compatibility with the latest SQL Server Enterprise edition database engine. It also provides a native virtual network implementation that addresses common security concerns.
- [Azure Data Lake Storage](#) is a storage repository that holds large amounts of data in its native, raw format. Data lake stores are optimized for scaling to terabytes and petabytes of data. The data typically comes from multiple heterogeneous sources. It can be structured, semi-structured, or unstructured.

## Compute

- Data Factory integrates data across different network environments by using an [integration runtime](#) (IR), which is a compute infrastructure. Data Factory copies data between cloud data stores and data stores in on-premises networks by using [self-hosted IRs](#).
- [Azure Virtual Machines](#) provides on-demand, scalable computing resources. An Azure virtual machine (VM) provides the flexibility of virtualization but eliminates the maintenance demands of physical hardware. Azure VMs offer a choice of operating systems, including Windows and Linux.

## Data integrators

- [Azure Data Factory](#) is a hybrid data integration service. In this solution, an Azure Data Factory custom connector uses the Host File client component of Host Integration Server to convert mainframe datasets. With minimal setup, you can use a custom connector to convert your mainframe dataset just as you'd use any other Azure Data Factory connector.
- [AzCopy](#) is a command-line utility that moves blobs or files into and out of storage accounts.
- [SQL Server Integration Services](#) is a platform for creating enterprise-level data integration and transformation solutions. You can use it to solve complex business problems by:
  - Copying or downloading files.
  - Loading data warehouses.
  - Cleansing and mining data.
  - Managing SQL Server objects and data.
- [Host Integration Server](#) technologies and tools enable you to integrate existing IBM host systems, programs, messages, and data with Azure applications. The Host File client component provides flexibility for data that's converted from EBCDIC to ASCII. For example, you can generate JSON/XML from the data that's converted.
- [Azure Synapse](#) brings together data integration, enterprise data warehousing, and big data analytics. The Azure Synapse conversion solution used in this architecture is based on Apache Spark and is a good candidate for large mainframe-dataset workload conversion. It supports a wide range of mainframe data structures and targets and requires minimal coding effort.

## Other tools

- [SQL Server Migration Assistant for Db2](#) automates migration from Db2 to Microsoft database services. When it runs on a VM, this tool converts Db2 database objects into SQL Server database objects and creates those objects in SQL Server.
- [Data Provider for Host Files](#) is a component of [Host Integration Server](#) that uses offline, SNA, or TCP/IP connections.
  - With offline connections, Data Provider reads and writes records in a local binary file.
  - With SNA and TCP/IP connections, Data Provider reads and writes records stored in remote z/OS (IBM Z Series Mainframe) datasets or remote i5/OS (IBM AS/400 and iSeries systems) physical files. Only i5/OS systems use TCP/IP.
- [Azure services](#) provide environments, tools, and processes for developing and scaling new applications in the public cloud.

# Scenario details

Modern data storage solutions like the Azure data platform provide better scalability and performance than mainframe and midrange systems. By modernizing your systems, you can take advantage of these benefits. However, updating technology, infrastructure, and practices is complex. The process involves an exhaustive investigation of business and engineering activities. Data management is one consideration when you modernize your systems. You also need to look at data visualization and integration.

Successful modernizations use a [data-first strategy](#). When you use this approach, you focus on the data rather than the new system. Data management is no longer just an item on the modernization checklist. Instead, the data is the centerpiece. Coordinated, quality-oriented data solutions replace fragmented, poorly governed ones.

This solution uses Azure data platform components in a data-first approach. Specifically, the solution involves:

- **Object conversion.** Converting object definitions from the source data store to corresponding objects in the target data store.
- **Data ingestion.** Connecting to the source data store and extracting data.
- **Data transformation.** Transforming extracted data into appropriate target data store structures.
- **Data storage.** Loading data from the source data store to the target data store, both initially and continually.

## Potential use cases

Organizations that use mainframe and midrange systems can benefit from this solution, especially when they want to achieve these goals:

- Modernize mission-critical workloads.
- Acquire business intelligence to improve operations and gain a competitive advantage.
- Remove the high costs and rigidity that are associated with mainframe and midrange data stores.

## Considerations

These considerations implement the pillars of the Azure Well-Architected Framework, a set of guiding tenets that you can use to improve the quality of a workload. For more information, see [Microsoft Azure Well-Architected Framework](#). When you use the Data

Provider for Host Files client to convert data, [turn on connection pooling](#) to reduce the connection startup time. When you use Data Factory to extract data, [tune the performance of the copy activity](#).

## Security

Security provides assurances against deliberate attacks and the abuse of your valuable data and systems. For more information, see [Overview of the security pillar](#).

- Be aware of the differences between on-premises client identities and client identities in Azure. You need to compensate for any differences.
- Use [managed identities](#) for component-to-component data flows.
- When you use Data Provider for Host Files to convert data, follow the recommendations in [Data Providers for Host Files security and protection](#).

## Cost optimization

Cost optimization is about reducing unnecessary expenses and improving operational efficiencies. For more information, see [Overview of the cost optimization pillar](#).

- SQL Server Migration Assistant is a free, supported tool that simplifies database migration from Db2 to SQL Server, SQL Database, and SQL Managed Instance. SQL Server Migration Assistant automates all aspects of migration, including migration assessment analysis, schema and SQL statement conversion, and data migration.
- The Azure Synapse Spark-based solution is built from open-source libraries. It eliminates the financial burden of licensing conversion tools.
- Use the [Azure pricing calculator](#) to estimate the cost of implementing this solution.

## Performance efficiency

Performance efficiency is the ability of your workload to scale to meet the demands placed on it by users in an efficient manner. For more information, see the [Performance efficiency pillar overview](#).

- The key pillars of performance efficiency are performance management, capacity planning, [scalability](#), and choosing an appropriate performance pattern.
- You can [scale out the self-hosted IR](#) by associating the logical instance with multiple on-premises machines in active-active mode.
- Azure SQL Database offers the ability to dynamically scale your databases. In a serverless tier, it can automatically scale the compute resources. Elastic Pool, which

allows databases to share resources in a pool, can only be scaled manually.

# Contributors

*This article is maintained by Microsoft. It was originally written by the following contributors.*

Principal author:

- [Ashish Khandelwal](#) | Principal Engineering Architect Manager

Other contributors:

- [Mick Alberts](#) | Technical Writer
- [Nithish Aruldoss](#) | Engineering Architect

*To see non-public LinkedIn profiles, sign in to LinkedIn.*

## Next steps

Review the [Azure Database Migration Guides](#). Contact [Azure Data Engineering - Mainframe & Midrange Modernization](#) for more information.

See these articles:

- [IBM workloads on Azure](#)
- [Mainframe rehosting on Azure virtual machines](#)
- [Mainframe workloads supported on Azure](#)
- [Move mainframe compute to Azure](#)

## Related resources

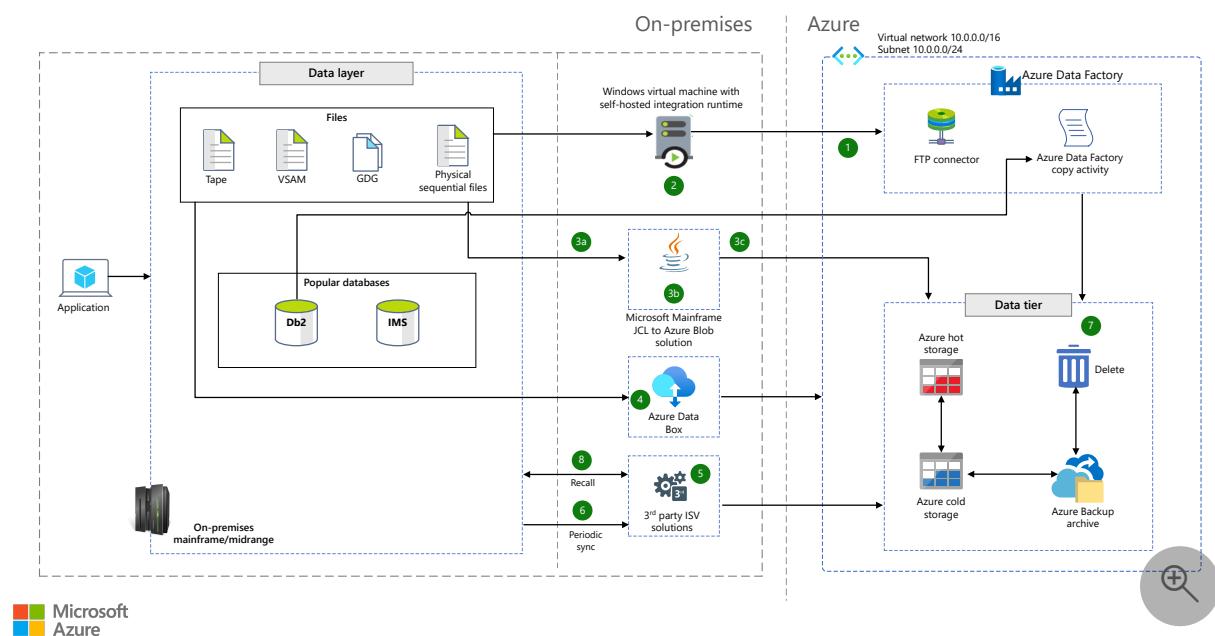
- [Azure data platform end-to-end](#)

# Move archive data from mainframe systems to Azure

Azure Data Factory   Azure Storage   Azure Files   Azure Blob Storage   Azure Data Box

This reference architecture shows how to move data from mainframe and midrange systems to Azure. In this architecture, archived data is serviced and used only in the mainframe system. Azure is used only as a storage medium.

## Architecture



Download a [Visio file](#) of this architecture.

To decide which method to use for moving data between the mainframe system and Azure storage, consider the frequency of data retrieval and the amount of data. Microsoft and third-party solutions are available:

- **Microsoft solutions.**
  - The Azure Data Factory FTP connector.
  - The Data Factory copy activity, which can copy data to any Azure storage solution.
  - *Mainframe JCL to Azure Blob using Java*, a custom solution for moving data from the mainframe system to Azure via Job Control Language (JCL). For more information, contact [datasqlninja@microsoft.com](mailto:datasqlninja@microsoft.com).

- **Third-party archive solutions.** Solutions that you can easily integrate with mainframe systems, midrange systems, and Azure services.

## Workflow

1. The Azure Data Factory [FTP connector moves data from the mainframe system to Azure Blob Storage](#). This solution requires an intermediate virtual machine (VM) on which a self-hosted integration runtime is installed.
2. The Data Factory [copy activity connects to the Db2 database to copy data into Azure storage](#). This solution also requires an intermediate VM on which a self-hosted integration runtime is installed.
3. The Microsoft *Mainframe JCL to Azure Blob using Java* custom solution moves data between the mainframe system and Blob Storage, and vice versa. This solution is based on Java and runs on Unix System Services on the mainframe. You can get this solution by contacting [datasqlninja@microsoft.com](mailto:datasqlninja@microsoft.com).
  - a. You need to complete a one-time configuration of the solution. This configuration involves getting the Blob Storage access keys and moving required artifacts to the mainframe system.
  - b. A JCL submission moves files to and from the mainframe and Blob Storage.
  - c. Files are stored in binary format on Azure. You can configure the custom solution to convert EBCDIC to ASCII for simple data types.
4. Optionally, Azure Data Box can help you physically transfer mainframe data to Azure. This option is appropriate when a large amount of data needs to be migrated and online methods of transmission take too long. (For example, if migration takes weeks.)
5. Easy interaction with the mainframe or midrange environment is provided by [third-party archive solutions](#).

These solutions interact with the mainframe and handle various mainframe parameters, like data types, record types, storage types, and access methods. They serve as a bridge between Azure and the mainframe. Some third-party solutions connect a storage drive to the mainframe and help transfer data to Azure.

6. Data is periodically synced and archived via the third-party archive solution. After the data is available via the third-party solution, the solution can easily push it to Azure by using available connectors.

7. Data is [stored in Azure](#).

8. As needed, [data is recalled from Azure](#) back to the mainframe or midrange systems.

## Components

- [Azure storage](#) provides massively scalable, highly secure cloud storage for your data, apps, and workloads. [Azure Files](#) provides simple and secure serverless cloud file shares. These components are used for synchronization and data retention.
- [Azure Data Factory](#) is a hybrid data integration service that you can use to create, schedule, and orchestrate your ETL and ELT workflows.
- [Azure Data Box](#) is a physical device that you can use to move on-premises data to Azure.

## Alternatives

You can use the classic method of moving the data out of the mainframe or midrange system via FTP. Data Factory provides an [FTP connector](#) that you can use to archive the data on Azure.

## Scenario details

Mainframe and midrange systems generate, process, and store huge amounts of data. When this data gets old, it's not typically useful. However, compliance and regulatory rules sometimes require this data to be stored for a certain number of years, so archiving it is critical. By archiving this data, you can reduce costs and optimize resources. Archiving data also helps with data analytics and provides a history of your data.

## Potential use cases

Archiving data to the cloud can help you:

- Free up storage resources in mainframe and midrange systems.
- Optimize performance for queries by storing only relevant data on the active system.
- Reduce operational costs by storing data in a more economical way.
- Use archived data for analytics to create new opportunities and make better business decisions.

# Recommendations

Depending on how you use data, you might want to convert it to ASCII from binary and then upload it to Azure. Doing so makes analytics easier on Azure.

## Considerations

- Complex data types on the mainframe must be handled during archive.
- Application subject matter experts can identify which data needs to be archived.
- To determine the amount of time between syncs, consider factors like business criticality, compliance needs, and frequency of data access.

## Third-party archive solutions

Some third-party solutions are available on [Azure Marketplace](#). Each of these solutions requires unique configuration. Setting up these solutions is one of the primary tasks of implementing this architecture.

## Azure storage

Azure has a variety of options for different application and technical requirements, like frequent versus infrequent access, and structured versus unstructured data. You can set up various storage lifecycle configurations in Azure storage. You can define the rules to manage the lifecycle. For an overview, see [Configure a lifecycle management policy](#).

## Data recall

Recall of archived data is an important aspect of archive solutions. Few of the third-party solutions provide a seamless experience for recalling archived data. It's as simple as running a command on-premises. The third-party agent automatically gets the data from Azure and ingests it back into the mainframe system.

## Cost optimization

Use the Azure [pricing calculator](#) to estimate the cost of implementing this solution.

## Contributors

*This article is maintained by Microsoft. It was originally written by the following contributors.*

Principal author:

- [Pratim Dasgupta](#) | Engineering Architect

Other contributors:

- [Mick Alberts](#) | Technical Writer
- [Ashish Khandelwal](#) | Senior Engineering Architect Manager
- [Ramanath Nayak](#) | Engineering Architect

## Next steps

For more information, contact [Azure Data Engineering - Mainframe/Midrange Modernization](#).

See these resources:

- [Azure Database Migration Guides](#)
- [What is Azure Data Factory?](#)
- [Introduction to Azure Storage](#)
- [What is Azure Files?](#)
- [What is Azure Data Box?](#)
- [Explore Azure Storage services](#)

## Related resources

- [Azure mainframe and midrange architecture concepts and patterns](#)
- [Modernize mainframe and midrange data](#)
- [Re-engineer IBM z/OS batch applications on Azure](#)
- [Replicate and sync mainframe data in Azure](#)

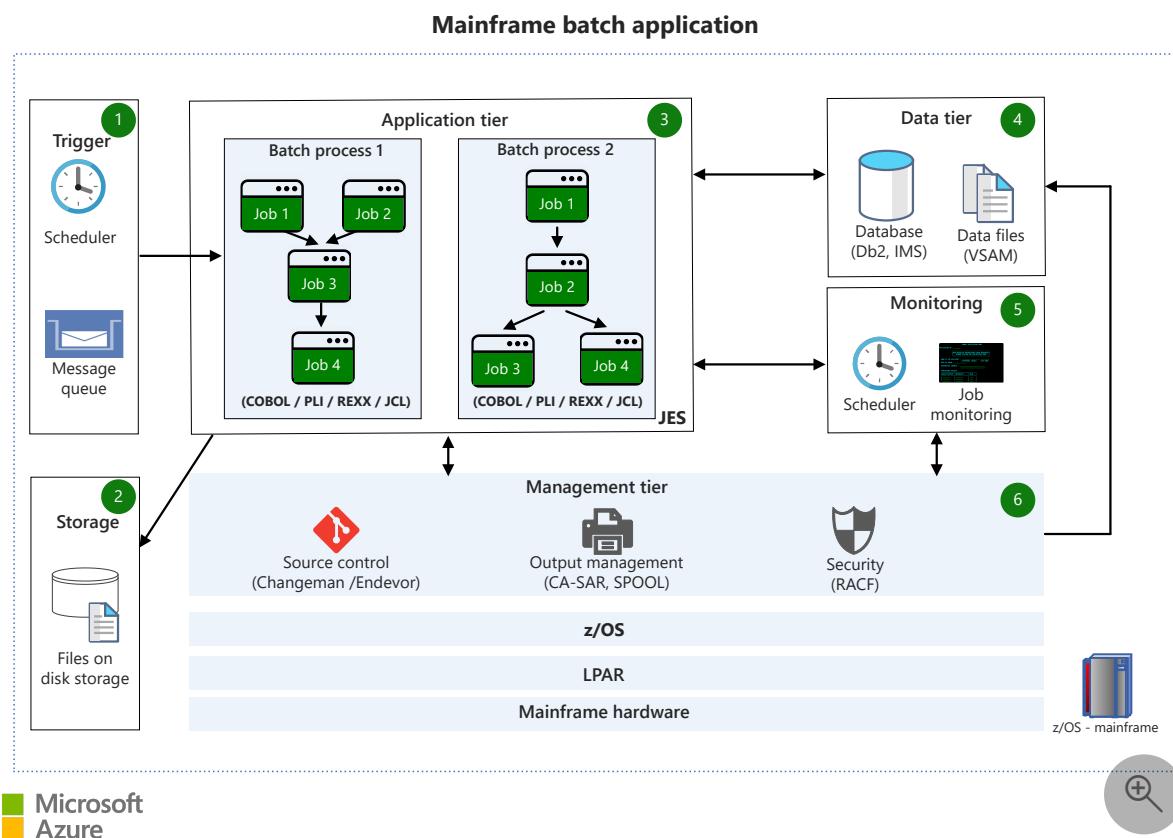
# Re-engineer mainframe batch applications on Azure

Azure Data Factory   Azure Databricks   Azure Kubernetes Service (AKS)   Azure SQL Database   Azure Storage

This reference architecture shows how you can use Azure to re-engineer a z/OS mainframe batch application to deliver a secure, scalable, and highly available system in the cloud using Azure. Because of ever evolving business needs, data and applications need to deliver and scale without affecting your infrastructure. Re-engineering to the cloud can help businesses in finance, health, insurance, and retail minimize their product or feature delivery times, and reduce costs.

## Mainframe Architecture

The first diagram shows the architecture of a typical batch application running on a z/OS mainframe.



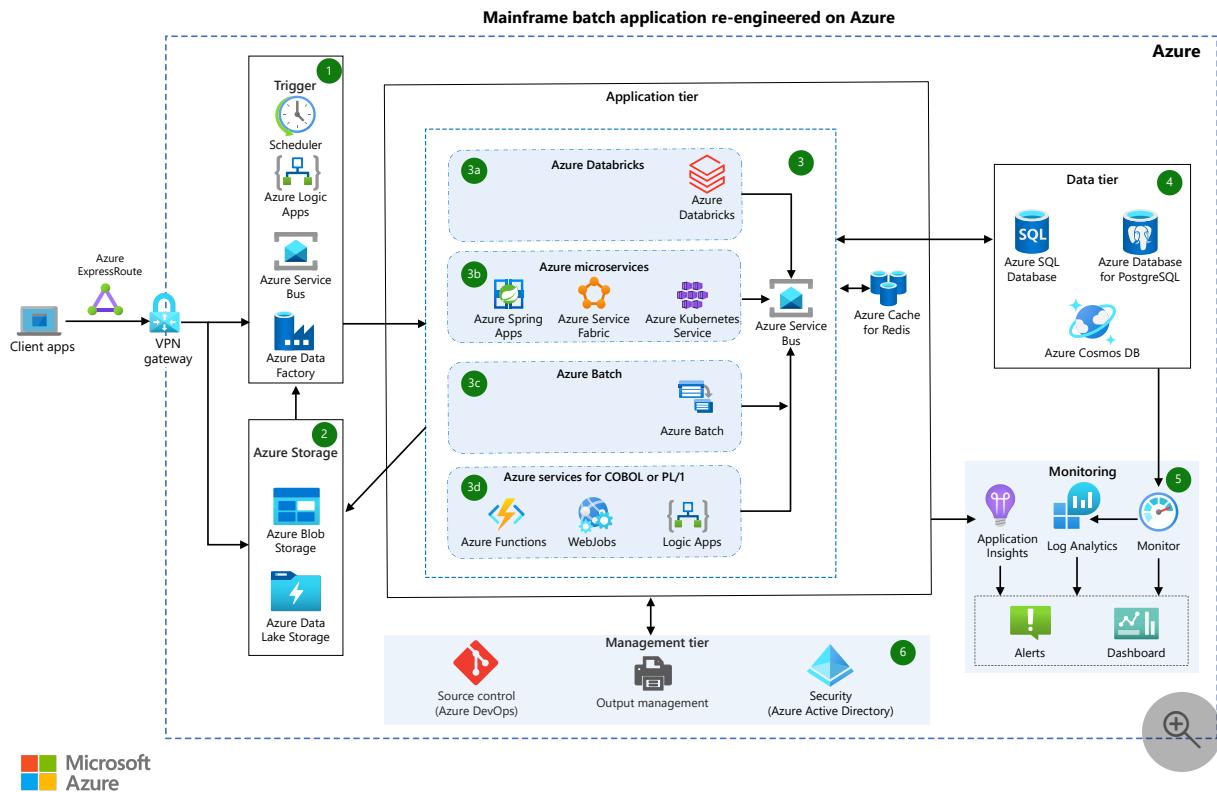
Download a [Visio file](#) of this architecture.

## Workflow

1. Mainframe batch processes can be triggered at a scheduled time using an Operation, Planning, and Control (OPC) Scheduler. They can also be triggered by a message placed in a message queue, like a message that announces that a file was created.
2. A mainframe direct-access storage device (DASD) is used to store input and output files; for example, flat files that are required by the application. You can trigger the batch process by creating a file on the DASD storage.
3. The batch process is an execution of a set of jobs, like a job internally running a user or system program to do a specific task. Usually, batch processes are run without user interaction. All batch jobs on a mainframe are executed under the control of a Job Execution System (JES).
4. Programs in batch processes can read/write data from:
  - A file-based database like Virtual Storage Access Method (VSAM).
  - A relational database like Db2 or Informix.
  - A non-relational database like Information Management System (IMS).
  - A message queue.
5. The output of the job execution can be monitored through an OPC scheduler or Tivoli Workload Scheduler (TWS). A System Display and Search Facility (SDSF) in the JES is also used on the mainframe to check job execution status.
6. The management tier provides the following services:
  - Source control, like Endevor or Changeman.
  - Security, like Resource Access Control Facility (RACF). This security provides authentication for running batches, accessing files, and accessing the database.
  - Output management that supports the storage and search of job execution logs.

## Azure architecture

The second diagram shows how you can use Azure services to re-engineer a similar application with added capabilities and flexibility.



Download a [Visio file](#) of this architecture.

## Workflow

1. Use one of the following triggers to start the Azure batch process.
  - Use the **Azure Databricks** job scheduler or **Azure Function** scheduler.
  - Create a recurring batch process task with **Azure Logic Apps**.
  - Use a storage event, like the creation or deletion of a file in **Azure Blob** or **File storage**.
  - Use a message-based trigger, like the arrival of a message on the **Azure Service Bus**.
  - Create an **Azure Data Factory** trigger.
2. Store files migrated from the mainframe by using **Azure Blob Storage** or **Azure Files**. Batch processes re-engineered on Azure can read/write data from this storage.
3. Azure provides various services to implement a mainframe batch workload. Select specific services that are based on your business requirements. For example, compute power required, total execution time, the ability to split mainframe batch process into smaller units, and cost sensitivity.
  - a. Azure Databricks is an Apache Spark-based analytics platform. Jobs can be written in the R, Python, Java, Scala, and Spark SQL languages. It provides a compute environment with fast cluster start times, auto termination, and

autoscaling. It has built-in integration with Azure storage like Azure Blob Storage and Azure Data Lake storage. Use Azure Databricks if you need to process large amounts of data in a short time. It's also a good choice if you need to run Extract, Transform, and Load (ETL) workloads.

- b. AKS and Service Fabric provide an infrastructure to implement a service-based application architecture. It might not be cost effective for a single application. You can refactor your mainframe application using Java Spring Boot. The best way to run Spring Boot apps on Azure is to use Azure Spring Apps, a fully managed Spring service. Java developers can use it to easily build and run Spring Boot Microservices on Azure.
- c. You can re-engineer your mainframe batch application using .NET or Java. Batch provides the infrastructure to run this application at scale. It creates and manages a pool of virtual machines (VMs), installs the applications, and then schedules jobs to run on the VMs. There's no cluster or job scheduler software to install, manage, or scale. Write applications in any programming language supported by Windows or Linux.
- d. You can re-engineer short running COBOL or PL/1 batch programs. For these programs, use Azure services like Functions, WebJobs, or Logic Apps.

#### 4. Azure provides various data services to store and retrieve data.

- You can migrate mainframe relational databases like Db2 and Informix with minimal changes to the Azure relational database offerings' visibility. For example, relational database services like Azure SQL VM, Azure SQL DB, or Azure SQL MI. You can also use any open-source Relational Database Management System (RDBMS) like Azure PostgreSQL. The selection of an Azure database depends on the type of workload, cross-database queries, two-phase commit requirements, and many other factors.
- You can migrate mainframe non-relational databases like IMS, Integrated Data Management System (IDMS), or VSAM to Azure Cosmos DB. Azure Cosmos DB provides fast response times, automatic and instant scalability, and guaranteed speed at any scale. It's a cost-effective option for unpredictable or sporadic workloads of any size or scale. Developers can easily get started without having to plan for or manage capacity.
- You can use Azure Cache for Redis to speed up a re-engineered application.

#### 5. Applications, the OS, and Azure resources can use agents to send logs and metrics to Azure Monitor Logs.

- **Application Insight** monitors your migrated application. It automatically detects performance anomalies and includes powerful analytics tools to help you diagnose issues.

- Azure Log Analytics helps store, index, query, and derive analytics from the log data collected.

You can use the output of Log Analytics and Application Insights to create alerts and dashboards, or export to external services. You can also use the output to do an action like the scaling of a VM.

6. This tier provides Azure services for source control, security, and output management. These services might consist of Azure DevOps and Microsoft Entra ID.

## Components

### Network and identity

- [Azure ExpressRoute](#): ExpressRoute lets you extend your on-premises networks into the Microsoft cloud over a private connection from a connectivity provider. With ExpressRoute, you can establish connections to Microsoft cloud services like Microsoft Azure and Office 365.
- [Azure VPN Gateway](#): A VPN gateway is a specific type of virtual network gateway that is used to send encrypted traffic between an Azure virtual network and an on-premises location over the public Internet.
- [Microsoft Entra ID](#): Microsoft Entra ID is an identity and access management service that can sync with an on-premises directory.

### Application

- [Logic Apps](#): Logic Apps helps you create and run automated recurring tasks and processes on a schedule. You can call services inside and outside Azure like HTTP or HTTPS endpoints. You can also post messages to Azure services like Azure Service Bus, or get files uploaded to a file share.
- [Service Bus](#): You can use the Service Bus for messaging between a user interface and back-end services. This system can decouple applications and services and increase reliability and use.
- [Azure Databricks](#): Azure Databricks is a cloud-based data engineering tool that's used for processing and transforming large amounts of data. You can then explore that data through machine learning models.
- [Azure Spring Apps](#): Azure Spring Apps makes it easy to deploy, manage, and run Spring microservices to Azure. It supports both Java and .NET Core.
- [AKS](#): AKS simplifies deploying a managed Kubernetes cluster in Azure by offloading the operational overhead to Azure.

- [Batch](#) : Batch is designed to run general purpose batch computing in the cloud across many VMs that can scale based on the workload being executed. It's a perfect fit for ETL or AI use cases where multiple tasks are executed in parallel, independent from each other.
- [Functions](#) : Use Functions to run small pieces of code without worrying about application infrastructure. With Functions, the cloud infrastructure provides all the up-to-date servers you need to keep your application running at scale.
- [Azure App Service](#) : With [WebJobs](#), a feature of App Service, you can code reusable background business logic as web jobs.
- [Azure Cache for Redis](#) : Applications that use a high volume of backend data can be developed to scale and deliver a highly optimized performance by integrating with an in-memory data store like Redis. Azure Cache for Redis offers both the Redis open-source (OSS Redis) and a commercial product from Redis Labs, Redis Enterprise, as a managed service.

## Storage

Azure storage provides multiple tiers of hot, cool, and archive data. Effective usage of these storage tiers can give you a price-to-performance advantage.

- [Blob Storage](#) : Scalable and secure object storage for cloud-native workloads, archives, data lakes, high-performance computing, and machine learning.
- [Azure Files](#) : Simple, secure, and serverless enterprise-grade cloud file shares. Azure Files can particularly come in handy for re-engineered mainframe solutions. It provides an effective add-on for the managed SQL storage.
- [Table Storage](#) : A NoSQL key-value store for rapid development using large semi-structured datasets.
- [Azure Queue Storage](#) : Simple, cost-effective, durable message queueing for large workloads.
- [Azure SQL](#) : Azure's fully managed family of services for SQL Server. You can migrate and use the relational data efficiently with other Azure services like Azure SQL Managed Instance, SQL Server on Azure Virtual Machines, and Azure Database for MariaDB.
- [Azure Cosmos DB](#) : A no-SQL offering that you can use to migrate non-tabular data from the mainframes.

## Monitoring

- [Azure Monitor](#) : Azure Monitor delivers a comprehensive solution for collecting, analyzing, and acting on telemetry from cloud and on-premises environments. It

contains the Application Insights, Azure Monitor Logs, and Azure Log Analytics features.

## Management

- [Azure DevOps](#) : Re-engineer mainframe applications on Azure during every phase of software development and team collaboration. DevOps provides the following services:
  - **Azure Boards**: Agile planning, work item tracking, visualization, and reporting tool.
  - **Azure Pipelines**: A language, platform, and cloud agnostic CI/CD platform with support for containers or Kubernetes.
  - **Azure Repos**: Provides cloud-hosted private git repos.
  - **Azure Artifacts**: Provides integrated package management with support for Maven, npm, Python, and NuGet package feeds from public or private sources.
  - **Azure Test Plans**: provides an integrated, planned, and exploratory testing solution.

## Scenario details

Mainframes are primarily used for processing large amounts of data. Batch processing is a way of processing a high volume of transactions that are grouped together, and then making bulk updates against the database. Once triggered, they require minimal to no user interaction. For example, mainframe systems make it possible for banks and other financial institutions to do end-of-quarter processing and produce reports, like quarterly stock or pension statements.

## Potential use cases

This solution is ideal for the finance, insurance, healthcare, and retail industries. Use this architecture to re-engineer mainframe applications on Azure. The architecture works best for:

- Resource-intensive mainframe batch applications.
- Batch applications that need high compute during a certain time, like end of month, quarter, or year.
- Mainframe batch processes that are repetitive and not resource-intensive but might need utilization by external systems.

## Considerations

## Availability

- The batch architecture in this article uses multi-node computing or PaaS services, which provide high availability.
- Azure database services support zone redundancy, and you can design them to fail over to a secondary node if there's an outage or during a maintenance window.

## Scalability

- The following Azure services in this architecture have autoscaling capabilities:
  - Azure Databricks
  - AKS
  - Spring Apps
  - Batch
  - Azure Functions
  - Logic Apps
- For more information on autoscaling in Azure, see the [autoscaling guide](#).

## Security

- This reference architecture uses ExpressRoute for a private and efficient connection to Azure from the on-premises environment. However, you can also create a [site to site VPN](#).
- You can authenticate Azure resources by using Microsoft Entra ID. You can manage permissions with role-based access control (RBAC).
- Database services in Azure support various security options like Data Encryption at Rest.
- For more information on designing secure solutions, see [Azure security documentation](#).

## Resiliency

- You can use Azure Monitor and Application Insights, in addition to Log Analytics, to monitor the health of an Azure resource. Set alerts to proactively manage your resource health.
- For more information on resiliency in Azure, see [Designing reliable Azure applications](#).

## Cost optimization

Use the [Azure pricing calculator](#) to estimate costs for Azure resources.

See [Azure mainframes batch application](#) for an example cost estimate of services.

## Contributors

*This article is maintained by Microsoft. It was originally written by the following contributors.*

Principal author:

- [Ashish Khandelwal](#) | Principal Engineering Architecture Manager

*To see non-public LinkedIn profiles, sign in to LinkedIn.*

## Next steps

- For more information, contact [datasqlninja@microsoft.com](mailto:datasqlninja@microsoft.com).
- See the [Azure database migration guides](#).

## Related resources

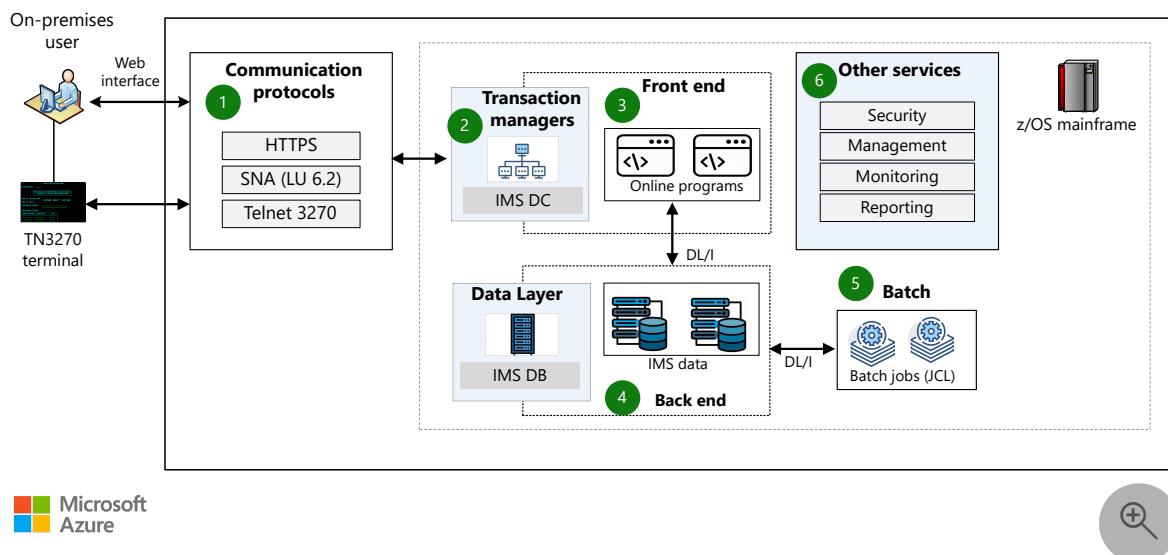
- [Azure mainframe and midrange architecture concepts and patterns](#)
- [High-volume batch transaction processing](#)

# Rehost IMS DC and IMS DB on Azure by using Raincode IMSql

Azure Virtual Machine Scale Sets   Azure Logic Apps   Azure SQL Managed Instance   Azure Virtual Network  
Azure ExpressRoute

This architecture describes how to implement an Information Management System (IMS) mainframe application workload on Azure by using Raincode's IMSql. Migrating an IMS database (DB) application to a cloud-native solution is more complex than migrating a relational database application. This article describes how to seamlessly rehost a mainframe IMS workload that has critical IMS features and capabilities to Azure. You don't need to translate or modify your existing application.

## IMS DB/DC workload architecture, before migration



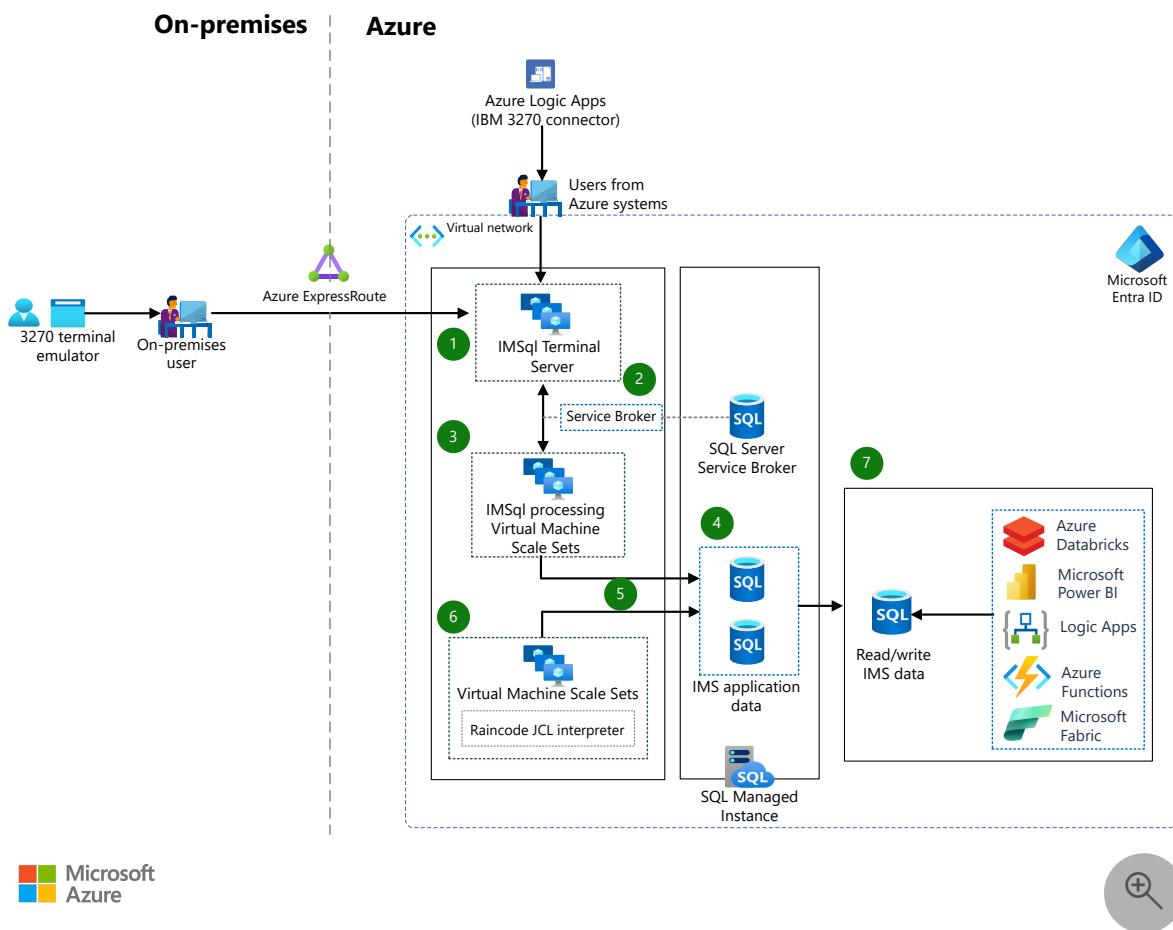
Download a [Visio file](#) of this architecture.

## Dataflow

1. Users connect to the mainframe over TCP/IP by using standard mainframe protocols like TN3270 and HTTPS.
2. Transaction managers interact with users and invoke the application to satisfy user requests.

3. In the front end of the application layer, users interact with IMS screens or with web pages.
4. Application code uses the storage capabilities of the IMS DB (hierarchical) back-end data layer.
5. All offline big data operations are performed via batch jobs.
6. Along with transaction processing, other services provide authentication, security, management, monitoring, and reporting. These services interact with all other services in the system.

## IMSsql architecture on Azure



Download a [Visio file](#) of this architecture.

## Workflow

### 1. IMSsql Terminal Server

Traditionally, the Mainframe z/OS interface is accessed via an IBM in-house terminal or via terminal emulation software. An application that has a geographically dispersed network with thousands of users can connect to the

mainframes through any form of terminal. When an IMS Data Communications (DC) application is rehosted on the distributed cloud-based system, you need to centrally host the application and resource and publish them for the remote client devices. You can accomplish these tasks on Azure by using IMSql Terminal Servers.

## 2. SQL Server Service Broker

In Mainframe, IMS DC orchestrates the communication layer between the user terminals and the application programs by transmitting and processing messages in a control region. After the rehost, SQL Server Service Broker orchestrates this asynchronous communication layer. Service Broker helps with communication through its message delivery framework and scales out messages to separate processing servers, current users, and their transaction processing.

## 3. IMSql Processing Server

The Processing Server runs the Raincode-recompiled code for the IMS programs in .NET Framework or .NET Core. It contains the underlying infrastructure that lets the recompiled programs run effectively with the correct functional equivalence. IMSql Processing Server can generate dynamic queries and call SQL stored procedures that are created during the recompilation of DL/I calls.

## 4. SQL Server as a hierarchical data store

Data is stored as hierarchical data in IMS. IMSql uses the same model on SQL Server. This model lets IMSql take advantage of the high performance of relational databases and logically implement the hierarchical segments from IMS. It also lets the system scale independently with segments. The segment data is stored in raw EBCDIC format, so it doesn't need to be converted for the application. By using SQL platform as a service (PaaS), IMSql can take advantage of the underlying HA/DR capabilities that are provided by Azure.

## 5. DL/I call API

The IMSql API ensures that the COBOL IMS DL/I calls are translated to equivalent SQL queries. It then fetches the data and returns it back to the application program in the expected format. IMSql also tracks the Program position on the Table record to perform the create, read, update, and delete (CRUD) operations, like the hierarchical DB. IMSql can create SQL stored procedures during compilation to respond to performance-intensive DL/I calls.

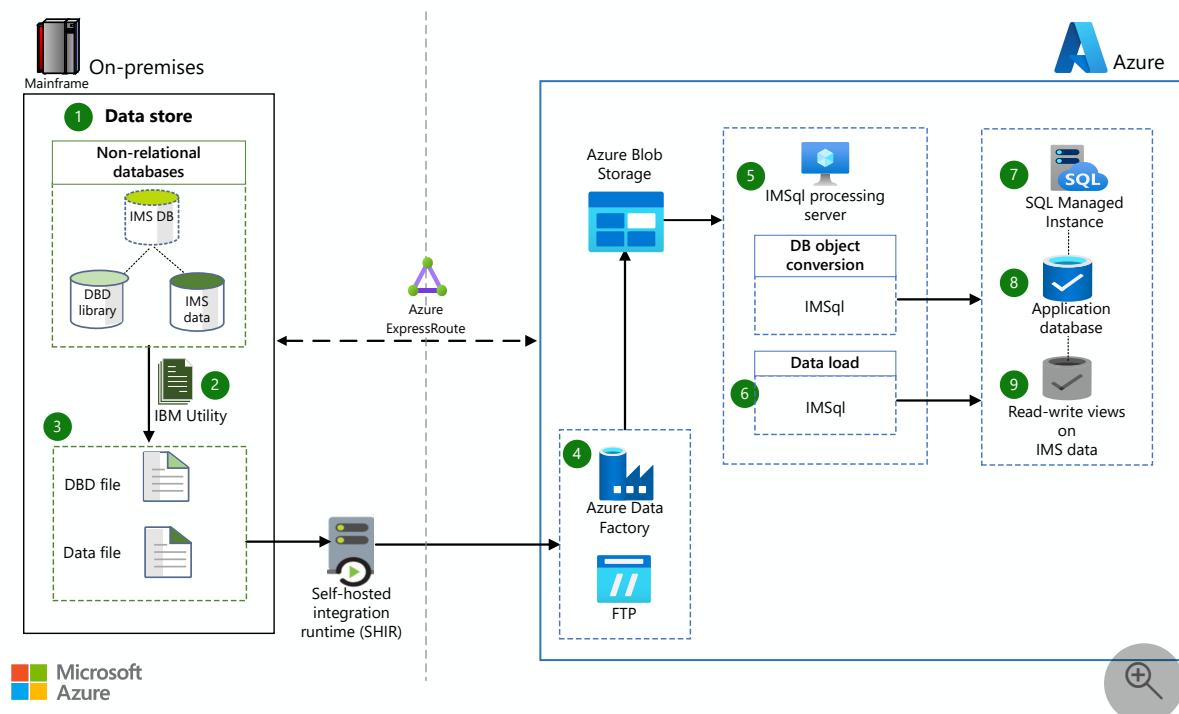
## 6. Raincode JCL

Raincode job control language (JCL) is an interpreter that's compatible with z/OS JCL. The Raincode JCL interpreter makes the transition from the intricate business logic embedded in JCL to the Azure and .NET Core platforms as smooth as possible. Raincode JCL is designed to run code compiled by the Raincode COBOL, PL/I, and ASM370 compilers. It can easily run steps written in virtually any language. It can be configured and fine-tuned with user-written code, so you can adapt it to your own needs for batch scheduling.

## 7. IMSql data view

IMSql defines relational SQL views based on copybooks (record layouts), so that the IMS segments can be accessed via plain SQL statements by any Azure service and by new applications. IMSql views are also writable, so modern applications can interact with IMS in both ways via SQL Server.

# Data migration via IMSql



Download a [Visio file](#) of this architecture.

## Database object migration

- The original IMS DB database description (DBD) is extracted and transferred from Mainframe. IMSql uses the DBD information to produce SQL scripts for generating a target database and tables in Azure SQL.
- Each segment in an IMS DBD is translated as a table on Azure.

- The tables consist of a key field, search fields, and the complete IMS segment data represented in EBCDIC.
- The IMS segment tree structure is retained with the primary and foreign key relationship in Azure SQL tables.

## Initial data load

- The data from IMS DB is extracted via a mainframe job and commonly available download utilities like DFSRRC00 and DFSURGL0.
- You can transfer the extracted binary files to Azure by using Azure Data Factory connectors like FTP and SFTP and a Java-based solution that runs on Unix Subsystem Services (USS).
- IMSql has a built-in load utility for completing the initial data loads. This tool uses the SQL Server bulk copy program (bcp) utility. It ensures bcp execution and the required referential integrity between the tables to match the expected hierarchical structure.
- This migration addresses a one-time data load from IMS DB, not coexistence and associated data synchronization.

## Dataflow for migration

1. The mainframe nonrelational datastore (IMS DB) has two components: the DBD and the actual segment data.
2. IBM utilities extract and unload the IMS DB information.
3. The DBD file and corresponding binary data files are generated separately.
4. Data ingestion:
  - a. The Data Factory FTP connector copies Mainframe IMS datasets to Azure data storage.
  - b. Mainframe IMS data files are copied to Azure Blob Storage via SFTP.
  - c. Mainframe JCL is used to run a custom Java solution that moves data between the mainframe system and SFTP Azure Blob Storage.
5. By using the DBD file, IMSql creates the target DB and tables, with necessary referential integrity.
6. After data objects are created, IMSql loads the data to the corresponding table in sequential order.
7. All migrated IMS data is hosted in Azure SQL Managed Instance.
8. The application database consists of the raw segment data for processing IMS online and batch processing.
9. The IMS read/write views consist of segment data that is expanded based on the copybook layout.

# Components

- [Azure Logic Apps](#) lets you quickly build powerful integration solutions. Mainframe users are familiar with 3270 terminals and on-premises connectivity. They can use the Logic Apps [IBM 3270 connector](#) to access and run IBM mainframe apps. In the migrated system, they interact with Azure applications via the public internet or a private connection that is implemented via Azure ExpressRoute. [Microsoft Entra ID](#) provides authentication.
- [Azure Virtual Network](#) is the fundamental building block for your private network on Azure. Virtual Network enables many types of Azure resources, like Azure virtual machines (VMs), to communicate with each other, the internet, and on-premises networks, all with improved security. Virtual Network is like a traditional network that you operate in your own datacenter, but it brings more of the benefits of the Azure infrastructure, like scale, availability, and isolation.
- [ExpressRoute](#) lets you extend your on-premises networks into the Microsoft Cloud over a private connection that a connectivity provider facilitates. You can use ExpressRoute to establish connections to Microsoft Cloud services like Azure and Office 365.
- [Azure Virtual Machine Scale Sets](#) provides automated and load balanced VM scaling that simplifies the management of your applications and increases availability.
- [SQL Managed Instance](#), part of the Azure SQL service portfolio, is a managed, highly secure, always up-to-date SQL instance in the cloud.
- [Microsoft Entra ID](#) is a cloud-based enterprise identity and access management service. Microsoft Entra single sign-on and multifactor authentication help users sign in and access resources while helping to protect against cybersecurity attacks.

# Alternatives

- You can use SQL Server in an Azure virtual machine as an alternative to SQL Managed Instance. We recommend SQL Managed Instance in this architecture because of benefits like high availability, seamless integration with various Azure services, and management of underlying security patches and maintenance.
- You can use an Azure single-VM architecture as an alternative to Virtual Machine Scale Sets. You might want to use single VMs for workloads that have constant load and performance demands and no need for scaling. This architecture uses Virtual Machine Scale Sets to handle typical IMS workloads.

# Scenario details

Mainframe OLTP systems can process millions of transactions for vast numbers of users. IBM IMS is a robust classic mainframe transaction manager used by major companies for online transaction processing. It has two main components: the IMS DC component and the underlying hierarchical DBMS IMS DB component.

IMSql provides a way to host IMS-based workloads on Azure or on-premises distributed implementations that are based on SQL Server. IMSql provides a holistic solution for running an IMS workload, including the app, data, and middleware components. It can ingest the hierarchical (IMS DB) data structure to a relational data model in SQL Server, SQL Server on Azure Virtual Machines, and SQL Managed Instance. It has built-in APIs for IMS application program DL/I calls and extends the data layer beyond the hierarchical workload to cloud-native apps that are used for relational data.

This solution provides the following benefits:

- Modernize infrastructure and reduce the high costs, limitations, and rigidity associated with monolithic mainframe IMS workloads.
- Reduce technical debt by implementing cloud-native solutions and DevOps.
- Provide IMS DB data to non-mainframe and cloud-based applications, including AI and analytics applications.

## Potential use cases

- Banking, finance, insurance, government, and retail industries that use Mainframe IMS. Many of these organizations run their primary OLTP and batch applications on IMS DB/DC.
- IBM zSeries mainframe customers who need to migrate mission-critical applications while maintaining continuity with other on-premises applications and avoiding the side effects of a complete redevelopment.

## Considerations

These considerations implement the pillars of the Azure Well-Architected Framework, which is a set of guiding tenets that you can use to improve the quality of a workload. For more information, see [Microsoft Azure Well-Architected Framework](#).

## Reliability

Reliability ensures your application can meet the commitments you make to your customers. For more information, see [Overview of the reliability pillar](#).

- This OLTP architecture can be deployed in multiple regions and can incorporate a geo-replication data layer.
- The Azure database services support zone redundancy and can fail over to a secondary node during outages or to enable maintenance activities.

## Security

Security provides assurances against deliberate attacks and the abuse of your valuable data and systems. For more information, see [Overview of the security pillar](#).

- ExpressRoute provides a private and efficient connection to Azure from on-premises.
- You can use Microsoft Entra ID to authenticate Azure resources. You can use role-based access control to manage permissions.
- This solution uses an Azure network security group to manage traffic to and from Azure resources. For more information, see [Network security groups](#).
- These security options are available in Azure database services:
  - Data encryption at rest
  - Dynamic data masking
  - Always Encrypted data

For general guidance on designing highly secure data solutions, see [Azure security recommendations](#).

## Cost optimization

Cost optimization is about looking at ways to reduce unnecessary expenses and improve operational efficiencies. For more information, see [Overview of the cost optimization pillar](#).

- Virtual Machine Scale Sets optimizes costs by minimizing the number of unnecessary hardware instances that run your application when demand is low.
- SQL Managed Instance provides various pricing tiers, like general purpose and business critical, to optimize costs based on usage and business criticality.
- [Azure Reservations](#) and [Azure savings plan for compute](#) with a one-year or three-year contract provide significant savings off pay-as-you-go prices. In many cases, you can further reduce your costs by implementing reserved-instance size flexibility.
- [Azure Hybrid Benefit](#) is a licensing benefit that can help you significantly reduce the costs of running your workloads in the cloud. It works by letting you use your

on-premises Software Assurance-enabled Windows Server and SQL Server licenses on Azure.

Use the [Azure pricing calculator](#) to estimate the cost of implementing this solution. Here's an [estimate based on the components of this solution, at a reasonable scale](#).

## Performance efficiency

Performance efficiency is the ability of your workload to scale to meet the demands placed on it by users in an efficient manner. For more information, see [Performance efficiency pillar overview](#).

- Virtual Machine Scale Sets ensures that enough VMs are available to meet mission-critical online and batch processing needs.
- Azure Blob Storage is a scalable system for storing backups, archival data, secondary data files, and other unstructured digital objects.
- [Database Engine Tuning Advisor](#) analyzes databases and makes recommendations that you can use to optimize query performance. You can use Database Engine Tuning Advisor to select and create an optimal set of indexes, indexed views, or table partitions.
- [Scalability](#) is one of the most important characteristics of PaaS. It lets you dynamically add resources to your service when they're needed. You can use Azure SQL Database to easily change the resources (CPU power, memory, I/O throughput, and storage) that are allocated to your databases. You can use SQL Managed Instance to dynamically add resources to your database with minimal downtime.
- [In-Memory OLTP](#) is a technology available in SQL Server and SQL Database for optimizing the performance of transaction processing, data ingestion, data load, and transient data scenarios.

## Contributors

*This article is maintained by Microsoft. It was originally written by the following contributors.*

Principal authors:

- [Nithish Aruldoss](#) | Engineering Architect
- [Amethyst Solomon](#) | Senior Engineering Architect

Other contributors:

- [Mick Alberts](#) | Technical Writer

*To see non-public LinkedIn profiles, sign in to LinkedIn.*

## Next steps

- [Mainframe to Azure Data Factory using FTP Connector](#)
- [Mainframe to Azure Data Platform using SFTP](#)
- [What is Azure Virtual Network?](#)
- [What is Azure ExpressRoute?](#)
- [Microsoft Fabric documentation](#)

For more information, contact [Azure Data Engineering - Mainframe Modernization](#).

## Related resources

See the companion architecture:

- [Rehost IMS workloads to virtual machines by using IMSql](#)

More related resources:

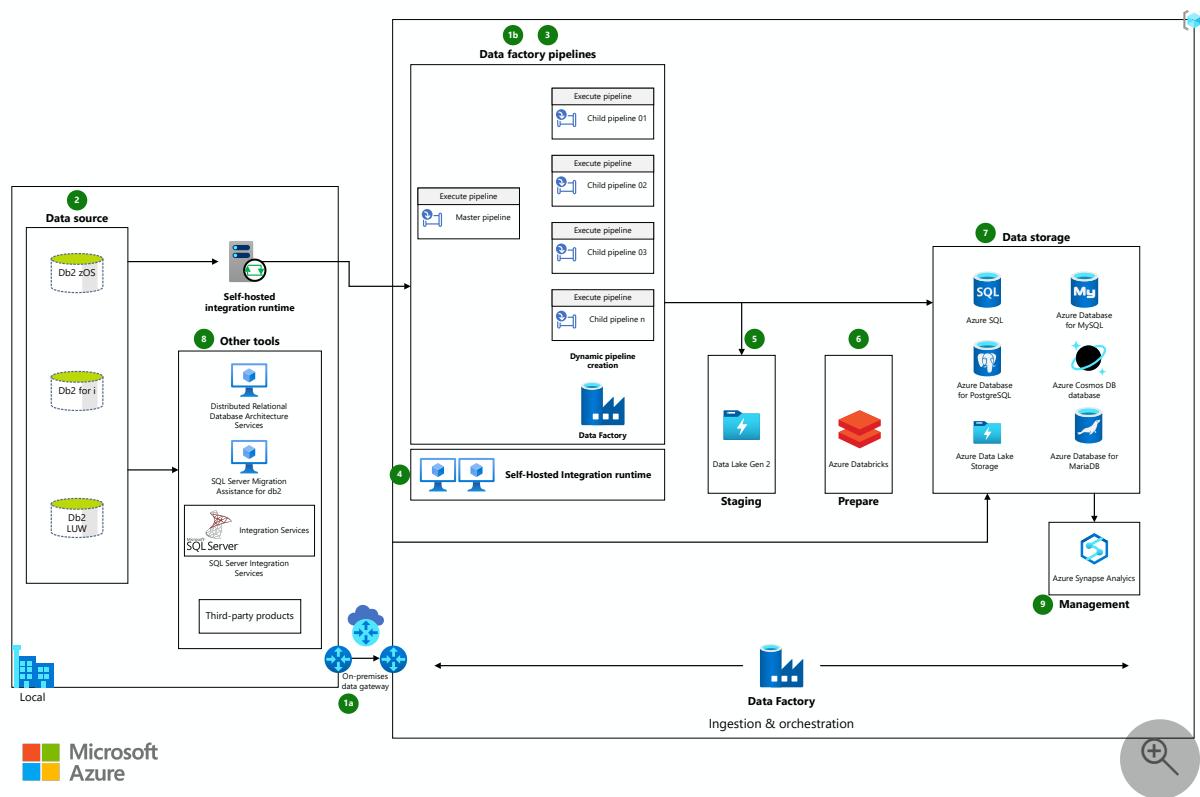
- [General mainframe refactor to Azure](#)
- [Mainframe access to Azure databases](#)
- [Re-engineer mainframe batch applications on Azure](#)
- [Rehost a general mainframe on Azure](#)

# Replicate and sync mainframe data in Azure

Azure Data Factory    Azure Databricks

This reference architecture outlines an implementation plan for replicating and syncing data during modernization to Azure. It discusses technical aspects like data stores, tools, and services.

## Architecture



Download a [Visio file](#) of this architecture.

## Workflow

Mainframe and midrange systems update on-premises application databases on a regular interval. To maintain consistency, the solution syncs the latest data with Azure databases. The sync process involves the following steps:

1. These actions occur throughout the process:

- a. An on-premises data gateway transfers data quickly and securely between on-premises systems and Azure services. With this configuration, the on-premises data gateway can receive instructions from Azure and replicate data without the on-premises network directly exposing the local data assets.
  - b. Azure Data Factory pipelines orchestrate activities that range from data extraction to data loading. You can schedule pipeline activities, start them manually, or automatically trigger them.
2. On-premises databases like Db2 zOS, Db2 for i, and Db2 LUW store the data.
  3. Pipelines group the activities that perform tasks. To extract data, Data Factory dynamically creates one pipeline per on-premises table. You can then use a massively parallel implementation when you replicate data in Azure. But you can also configure the solution to meet your requirements:
    - Full replication: You replicate the entire database, making necessary modifications to data types and fields in the target Azure database.
    - Partial, delta, or incremental replication: You use *watermark columns* in source tables to sync updated rows with Azure databases. These columns contain either a continuously incrementing key or a time stamp indicating the table's last update.
- Data Factory also uses pipelines for the following transformation tasks:
- Data type conversion
  - Data manipulation
  - Data formatting
  - Column derivation
  - Data flattening
  - Data sorting
  - Data filtering
4. A self-hosted integration runtime (IR) provides the environment that Data Factory uses to run and dispatch activities.
  5. Azure Data Lake Storage Gen2 and Azure Blob Storage provide a place for data staging. This step is sometimes required for transforming and merging data from multiple sources.
  6. Data preparation takes place next. Data Factory uses Azure Databricks, custom activities, and pipeline data flows to transform data quickly and effectively.
  7. Data Factory loads data into relational and non-relational Azure databases:

- Azure SQL
- Azure Database for PostgreSQL
- Azure Cosmos DB
- Azure Data Lake Storage
- Azure Database for MariaDB
- Azure Database for MySQL

In certain use cases, other tools can also load data.

## 8. Other tools can also replicate and transform data:

- Microsoft Service for Distributed Relational Database Architecture (DRDA): These DRDA services can connect to the Azure SQL family of databases and keep on-premises databases up to date. These services run on an on-premises virtual machine (VM) or an Azure VM.
- SQL Server Migration Assistance (SSMA) for Db2: This tool migrates schemas and data from IBM Db2 databases to Azure databases.
- SQL Server Integration Services (SSIS): This platform can extract, transform, and load data.
- Third-party tools: When the solution requires near real-time replication, you can use third-party tools. Some of these agents are available in [Azure Marketplace](#).

## 9. Azure Synapse Analytics manages the data and makes it available for business intelligence and machine learning applications.

# Components

The solution uses the following components:

## Tools

- [Microsoft Service for DRDA](#) is a component of [Host Integration Server \(HIS\)](#). Microsoft Service for DRDA is an Application Server (AS) that DRDA Application Requester (AR) clients use. Examples of DRDA AR clients include IBM Db2 for z/OS and Db2 for i5/OS. These clients use the AS to convert Db2 SQL statements and run them on SQL Server.
- [SSMA for Db2](#) automates migration from Db2 to Microsoft database services. While running on a VM, this tool converts Db2 database objects into SQL Server database objects and creates those objects in SQL Server. SSMA for Db2 then migrates data from Db2 to the following services:

- SQL Server 2012
  - SQL Server 2014
  - SQL Server 2016
  - SQL Server 2017 on Windows and Linux
  - SQL Server 2019 on Windows and Linux
  - Azure SQL Database
- [Azure Synapse Analytics](#) is an analytics service for data warehouses and big data systems. This tool uses Spark technologies and has deep integration with Power BI, Azure Machine Learning, and other Azure services.

## Data integrators

- [Azure Data Factory](#) is a hybrid data integration service. You can use this fully managed, serverless solution to create, schedule, and orchestrate ETL and [ELT](#) workflows.
- [Azure Synapse Analytics](#) is an enterprise analytics service that accelerates time to insight, across data warehouses and big data systems. Azure Synapse brings together the best of SQL technologies (that are used in enterprise data warehousing), Spark technologies used for big data, Data Explorer for log and time series analytics, Pipelines for data integration and ETL/ELT, and deep integration with other Azure services, such as Power BI, Azure Cosmos DB, and Azure Machine Learning.
- [SQL Server Integration Services \(SSIS\)](#) is a platform for building enterprise-level data integration and transformation solutions. You can use SSIS to manage, replicate, cleanse, and mine data.
- [Azure Databricks](#) is a data analytics platform. Based on the Apache Spark open-source distributed processing system, Azure Databricks is optimized for the Azure cloud platform. In an analytics workflow, Azure Databricks reads data from multiple sources and uses Spark to provide insights.

## Data storage

- [Azure SQL Database](#) is part of the [Azure SQL](#) family and is built for the cloud. This service offers all the benefits of a fully managed and evergreen platform as a service. SQL Database also provides AI-powered, automated features that optimize performance and durability. Serverless compute and [Hyperscale storage options](#) automatically scale resources on demand.

- [SQL Managed Instance](#) is part of the Azure SQL service portfolio. This intelligent, scalable, cloud database service combines the broadest SQL Server engine compatibility with all the benefits of a fully managed and evergreen platform as a service. With SQL Managed Instance, you can modernize existing apps at scale.
- [SQL Server on Azure VMs](#) provides a way to lift and shift SQL Server workloads to the cloud with 100 percent code compatibility. As part of the Azure SQL family, SQL Server on Azure VMs offers the combined performance, security, and analytics of SQL Server with the flexibility and hybrid connectivity of Azure. With SQL Server on Azure VMs, you can migrate existing apps or build new apps. You can also access the latest SQL Server updates and releases, including SQL Server 2019.
- [Azure Database for PostgreSQL](#) is a fully managed relational database service that's based on the community edition of the open-source [PostgreSQL](#) database engine. With this service, you can focus on application innovation instead of database management. You can also scale your workload quickly and easily.
- [Azure Cosmos DB](#) is a globally distributed, [multimodel](#) database. With Azure Cosmos DB, your solutions can elastically and independently scale throughput and storage across any number of geographic regions. This fully managed [NoSQL](#) database service guarantees single-digit millisecond latencies at the ninety-ninth percentile anywhere in the world.
- [Data Lake Storage](#) is a storage repository that holds a large amount of data in its native, raw format. Data lake stores are optimized for scaling to terabytes and petabytes of data. The data typically comes from multiple, heterogeneous sources and may be structured, semi-structured, or unstructured. [Data Lake Storage Gen2](#) combines Data Lake Storage Gen1 capabilities with Blob Storage. This next-generation data lake solution provides file system semantics, file-level security, and scale. But it also offers the tiered storage, high availability, and disaster recovery capabilities of Blob Storage.
- [Azure Database for MariaDB](#) is a cloud-based relational database service. This service is based on the [MariaDB](#) community edition database engine.
- [Azure Database for MySQL](#) is a fully managed relational database service based on the [community edition of the open-source MySQL database engine](#).
- [Blob Storage](#) provides optimized cloud object storage that manages massive amounts of unstructured data.

## Networking

- An [on-premises data gateway](#) acts as a bridge that connects on-premises data with cloud services. Typically, you [install the gateway on a dedicated on-premises VM](#). Cloud services can then securely use on-premises data.
- An [IR](#) is the compute infrastructure that Data Factory uses to integrate data across different network environments. Data Factory uses [self-hosted IRs](#) to copy data between cloud data stores and data stores in on-premises networks. You can also use [Azure Synapse Pipelines](#).

## Scenario details

Data availability and integrity play an important role in mainframe and midrange modernization. [Data-first strategies](#) help to keep data intact and available during migration to Azure. To avoid impacting applications during modernization, sometimes you need to replicate data quickly or keep on-premises data in sync with Azure databases.

Specifically, this solution covers:

- Extraction: Connecting to and extracting from a source database.
- Transformation:
  - Staging: Temporarily storing data in its original format and preparing it for transformation.
  - Preparation: Transforming and manipulating data by using mapping rules that meet target database requirements.
- Loading: Inserting data into a target database.

## Potential use cases

Data replication and sync scenarios that can benefit from this solution include:

- Command Query Responsibility Segregation (CQRS) architectures that use Azure to service all inquire channels.
- Environments that test on-premises applications and rehosted or re-engineered applications in parallel.
- On-premises systems with tightly coupled applications that require phased remediation or modernization.

## Recommendations

When you use Data Factory to extract data, take steps to [tune the performance of the copy activity](#).

## Considerations

These considerations implement the pillars of the Azure Well-Architected Framework, which is a set of guiding tenets that can be used to improve the quality of a workload. For more information, see [Microsoft Azure Well-Architected Framework](#).

Keep these points in mind when considering this architecture.

## Reliability

Reliability ensures your application can meet the commitments you make to your customers. For more information, see [Overview of the reliability pillar](#).

- Infrastructure management, including [availability](#), is automated in Azure databases.
- See [Pooling and failover](#) for information on the failover protection that Microsoft Service for DRDA provides.
- You can cluster the on-premises data gateway and IR to provide higher availability guarantees.

## Security

Security provides assurances against deliberate attacks and the abuse of your valuable data and systems. For more information, see [Overview of the security pillar](#).

- Make use of [network security groups](#) to limit access of services to only what they need to function.
- Use [private endpoints](#) for your PaaS (Platform as a Service) services. Use service firewalls to supplement security for your services that are both reachable and unreachable through the Internet.
- Be aware of the differences between on-premises client identities and client identities in Azure. You will need to compensate for any differences.
- Use managed identities for component-to-component data flows.

- See [Planning and Architecting Solutions Using Microsoft Service for DRDA](#) to learn about the types of client connections that Microsoft Service for DRDA supports. Client connections affect the nature of transactions, pooling, failover, authentication, and encryption on your network.

## Cost optimization

Cost optimization is about looking at ways to reduce unnecessary expenses and improve operational efficiencies. For more information, see [Overview of the cost optimization pillar](#).

- Pricing models vary between component services. Review the pricing models of the available component services to ensure the pricing models fit your budget.
- Use the [Azure pricing calculator](#) to estimate the cost of implementing this solution.

## Operational excellence

Operational excellence covers the operations processes that deploy an application and keep it running in production. For more information, see [Overview of the operational excellence pillar](#).

- Infrastructure management, including [scalability](#), is automated in Azure databases.
- You can [scale out the self-hosted IR](#) by associating the logical instance with multiple on-premises machines in active-active mode.
- You can cluster the on-premises data gateway and IR for scalability.

## Performance efficiency

Performance efficiency is the ability of your workload to scale to meet the demands placed on it by users in an efficient manner. For more information, see [Performance efficiency pillar overview](#).

- When you use an on-premises application gateway, be aware of [limits on read and write operations](#).
- Consider [Azure ExpressRoute](#) as a high-scale option if your implementation uses significant bandwidth for initial replication or ongoing changed data replication.

- The [self-hosted IR](#) can only run on a Windows operating system.

## Next steps

- Contact [Azure Data Engineering - On-premises Modernization](#) for more information.
- Read the [Migration guide](#).

## Related resources

- [\[Azure data architecture guide\]](#)
- [Azure data platform end-to-end](#)

# Mainframe and midrange Db2 applications accessing Azure SQL databases

Azure SQL Database

Azure Virtual Machines

SQL Server

## 💡 Solution ideas

This article is a solution idea. If you'd like us to expand the content with more information, such as potential use cases, alternative services, implementation considerations, or pricing guidance, let us know by providing [GitHub feedback](#).

This solution outlines a way for IBM mainframe and midrange applications to access remote Azure databases. The approach requires zero or minimal changes in application code.

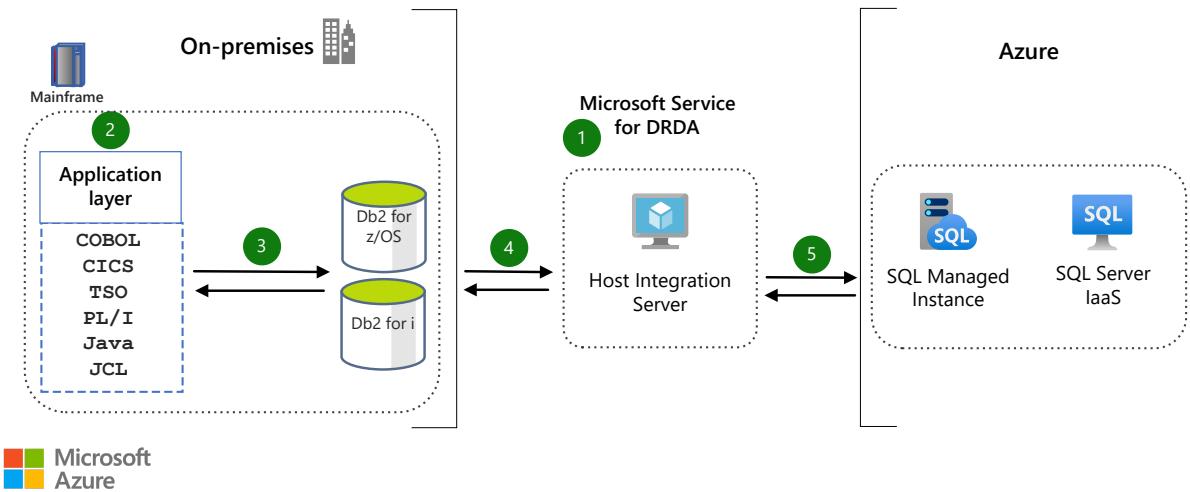
IBM Db2 clients and servers use the Distributed Relational Database Architecture (DRDA) protocol to communicate. In this solution, Microsoft Service for DRDA connects Db2 clients on IBM z/OS and IBM i to SQL Server–based databases by supporting this protocol.

## Potential use cases

Various scenarios can benefit from this solution:

- *Coexistent* environments that have modernized data as part of a [data-first](#) migration but still run mainframe or midrange applications.
- *Hybrid* situations, or environments that combine on-premises and cloud datacenters. This case covers systems with mainframe applications in COBOL, PL/I, or assembly language that need access to an SQL Server database hosted in Azure.
- Mainframe or midrange systems with workloads that need remote access to SQL Server databases.

## Architecture



Download a [Visio file](#) of this architecture.

1. Host Integration Server (HIS) software runs on an on-premises or Azure virtual machine (VM). HIS connects IBM systems with Azure systems.
2. Mainframe and midrange applications run on the on-premises system. These applications use languages and environments like COBOL, CICS, TSO, PL1, Java, and JCL. The solution involves adjusting the Db2 database configuration. The applications can then access Azure databases in the same way that they access local mainframe or midrange tables.
3. A mainframe or midrange application sends a SQL request to the local Db2 subsystem. Db2 configurations reroute the request to the HIS server.
4. The HIS server receives the request and forwards it to the target database. Microsoft Service for DRDA is a component of HIS that functions as a DRDA Application Server (AS). In this role, Microsoft Service for DRDA converts the Db2 SQL statements and runs them on the Azure database.
5. The target database handles the request. This solution can configure the following target databases:
  - Azure SQL Database, which offers the benefits of a fully managed platform as a service (PaaS).
  - SQL Server on Azure Virtual Machines. As an infrastructure as a service (IaaS) offering, this service provides a customizable database engine.
  - SQL Server, a database engine for structured and unstructured data.

These database services can also form the core of business intelligence solutions that offer analytics and insights.

# Components

This solution uses the following components. See the [Azure pricing calculator](#) to estimate costs for Azure resources.

## Data stores

- [SQL Database](#) is a relational database service that's part of the [Azure SQL](#) family. As a fully managed service, SQL Database handles database management functions like upgrading, patching, backups, and monitoring. SQL Database also provides AI-powered, automated features that optimize performance and durability. Serverless compute and Hyperscale storage options automatically scale resources on demand.
- [SQL Server on Azure Virtual Machines](#) provides a way to migrate SQL Server workloads to the cloud with 100 percent code compatibility. As part of the Azure SQL family, SQL Server on Azure Virtual Machines offers the flexibility and hybrid connectivity of Azure. But this database solution also provides the performance, security, and analytics of SQL Server. With SQL Server on Azure Virtual Machines, you can migrate existing apps or build new apps. You can also access the latest SQL Server updates and releases.
- [SQL Server](#) provides a solution for storing and querying structured and unstructured data. This database engine features industry-leading performance and security.

## Tools

- [HIS](#) software connects IBM systems with Azure systems. HIS runs on an on-premises or Azure VM. HIS provides integration services for networks, data, applications, messaging, and security features.
- [Microsoft Service for DRDA](#) is a component of HIS. Microsoft Service for DRDA is an Application Server (AS) that DRDA Application Requester (AR) clients use. Examples of DRDA AR clients include IBM Db2 for z/OS and Db2 for i. These clients use the AS to convert Db2 SQL statements and run them on SQL Server.

# Contributors

*This article is maintained by Microsoft. It was originally written by the following contributors.*

Principal authors:

- [Nithish Aruldoss](#) | Engineering Architect
- [Ashish Khandelwal](#) | Senior Engineering Architect

*To see non-public LinkedIn profiles, sign in to LinkedIn.*

## Next steps

- For general information on mainframe modernization and database migration:
  - Contact Azure Data Engineering - Mainframe & Midrange Modernization at [datasqlninja@microsoft.com](mailto:datasqlninja@microsoft.com).
  - See [Azure Database Migration Guides](#).
  - See [Planning and architecting solutions using Microsoft Service for DRDA](#).
  - See [Migrate databases and data](#).
- For implementation information:
  - See [Install and configure HIS 2020](#).
  - Learn how to [add information on a target database to an HIS server configuration](#).
  - See how to [configure a Db2 database to reroute requests to an HIS server](#).

## Related resources

- [Mainframe file replication and sync on Azure](#)
- [Replicate and sync mainframe data in Azure](#)
- [Modernize mainframe and midrange data](#)
- [Migrate IBM mainframe applications to Azure with TmaxSoft OpenFrame](#)

# Mainframe file replication and sync on Azure

Azure Data Factory

Azure Data Lake

Azure SQL Database

Azure Storage

Azure Virtual Machines

## 💡 Solution ideas

This article is a solution idea. If you'd like us to expand the content with more information, such as potential use cases, alternative services, implementation considerations, or pricing guidance, let us know by providing [GitHub feedback](#).

When you migrate an on-premises mainframe or midrange application to Azure, moving the data is a primary consideration. Several modernization scenarios require replicating files to Azure quickly, or maintaining a sync between on-premises and Azure files.

This article discusses several different processes for moving files to Azure, converting and transforming file data, and storing the data on-premises and in Azure. The article highlights a wide range of Azure services to demonstrate the possibilities.

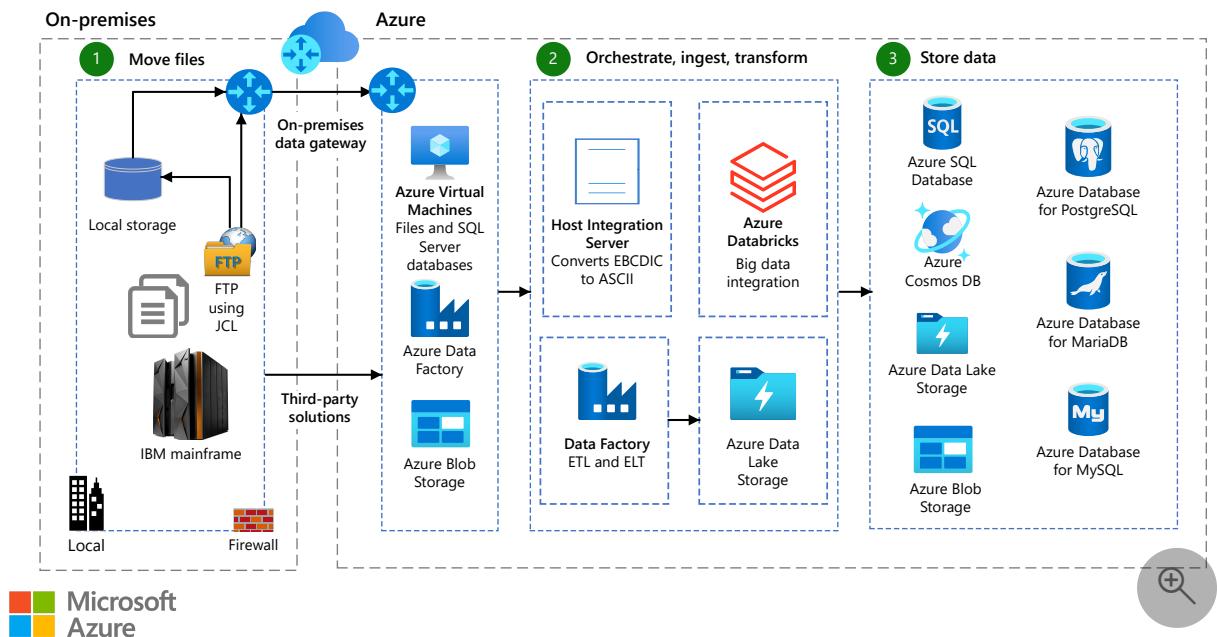
## Potential use cases

On-premises file replication and sync use cases include:

- Downstream or upstream dependency, if applications running on the mainframe and applications running on Azure need to exchange data via files.
- Parallel testing of rehosted or re-engineered applications on Azure with on-premises applications.
- Tightly coupled on-premises applications on systems that can't immediately be remediated or modernized.

## Architecture

The following diagram shows some of the options for replicating and syncing on-premises files to Azure:



Download a [Visio file](#) of this architecture.

## Dataflow

### 1. Move files to Azure:

- The easiest way to move files on-premises or to Azure is by using [File Transfer Protocol \(FTP\)](#). You can host an FTP server on an Azure virtual machine (VM). A simple FTP job control language (JCL) sends files to Azure in binary format, which is essential to preserving mainframe and midrange computation and binary data types. You can store transmitted files in on-premises disks, Azure VM file storage, or Azure Blob Storage.
- An on-premises data gateway and firewall provide secure connections to cloud services for on-premises data.
- You can also upload on-premises files to Blob Storage by using tools like [AzCopy](#).
- Azure Data Factory also hosts various data source connectors for migrating file data into Azure.
- There are also third-party solutions that can help move files from mainframes to Azure. You can find some of them in the [Azure Marketplace](#).

### 2. Orchestrate, convert, and transform data:

- Azure can't read IBM Extended Binary Coded Decimal Interchange Code (EBCDIC) code page files in Azure VM disks or Blob Storage. To make these

files compatible with the Azure character set, Host Integration server (HIS) converts them from EBCDIC to American Standard Code for Information Interchange (ASCII) format.

Copybooks define the data structure of COBOL, PL/I, and assembly language files. HIS converts these files to ASCII based on the copybook layouts.

- Before moving data to Azure data stores, you might need to transform the data or use it for analytics. Azure Data Factory can manage these *extract-transform-load (ETL)* and *extract-load-transform (ELT)* activities, and store the data directly in Azure Data Lake Storage.
- For big data integrations, Azure Databricks can perform all transformation activities fast and effectively by using the Apache Spark engine to do in-memory computations.

### 3. Store data:

You can store moved data in one of several available persistent Azure storage modes, depending on your requirements.

- If there's no need for analytics, Azure Data Factory can store data directly in a wide range of storage options, such as Data Lake Storage and Blob Storage.
- Azure hosts various databases, which address different needs:
  - Relational databases include the SQL Server family, and open-source databases like PostgreSQL, MariaDB, and MySQL.
  - Non-relational databases include Azure Cosmos DB, a fast, multi-model, globally distributed NoSQL database.

## Components

Various file moving, integration, and storage scenarios use different components. See the [Azure pricing calculator](#) to estimate costs for Azure resources.

## Networking

- An [on-premises data gateway](#) is bridge software that connects on-premises data to cloud services. The gateway typically [installs on a dedicated on-premises VM](#).

## Data integration and transformation

- [Data Provider for Host Files](#) is a component of [HIS](#) that converts EBCDIC code page files to ASCII. The provider can read and write records offline in a local binary file, or use Systems Network Architecture (SNA) or Transmission Control Protocol/Internet Protocol (TCP/IP) to read and write records in remote IBM z/OS mainframe data sets or i5/OS physical files. HIS connectors are available for [BizTalk](#) and [Azure Logic Apps](#).
- [Azure Data Factory](#) is a hybrid data integration service that helps you create, schedule, and orchestrate ETL and ELT workflows.
- [Azure Databricks](#) is an Apache Spark-based analytics platform optimized for Azure. You can use Databricks to correlate incoming data, and enrich it with other data stored in Databricks.

## Databases

- [Azure SQL Database](#) is a scalable relational cloud database service. Part of the Azure SQL family, Azure SQL Database is evergreen and always up to date, with AI-powered and automated features that optimize performance and durability. Serverless compute and hyperscale storage options automatically scale resources on demand. With [Azure Hybrid Benefit](#), you can use your existing on-premises SQL Server licenses on the cloud with no extra cost.
- [Azure SQL Managed Instance](#) combines the broadest SQL Server database engine compatibility with all the benefits of a fully managed and evergreen platform as a service. With SQL Managed Instance, you can modernize your existing apps at scale with familiar tools, skills, and resources.
- [Azure SQL on VM](#) lifts and shifts your SQL Server workloads to the cloud to combine the flexibility and hybrid connectivity of Azure with SQL Server performance, security, and analytics. You can access the latest SQL Server updates and releases with 100 percent code compatibility.
- [Azure Database for PostgreSQL](#) is a fully managed relational database service based on the community edition of the open-source PostgreSQL database engine.
- [Azure Database for MySQL](#) is a fully managed relational database service based on the community edition of the open-source MySQL database engine.
- [Azure Database for MariaDB](#) combines the MariaDB community edition with the benefits of a fully managed service.

- [Azure Cosmos DB](#) is a fully managed, multi-model NoSQL database service for building and modernizing scalable, high-performance applications. Azure Cosmos DB scales throughput and storage elastically and independently across geographic regions, and guarantees single-digit-millisecond latencies at 99th percentile availability anywhere in the world.

## Other data stores

- [Blob Storage](#) stores large amounts of unstructured data, such as text or binary data, that you can access from anywhere via HTTP or HTTPS. You can use Blob Storage to expose data publicly, or to store application data privately.
- [Data Lake Storage](#) is a storage repository that holds a large amount of data in native, raw format. Data Lake Storage is optimized for scaling to big data analytics workloads with terabytes and petabytes of data. The data typically comes from multiple heterogeneous sources, and may be structured, semi-structured, or unstructured.

## Contributors

*This article is maintained by Microsoft. It was originally written by the following contributors.*

Principal author:

- [Ashish Khandelwal](#) | Senior Engineering Architect

## Next steps

- For more information, contact Azure Data Engineering On-premises Modernization at [datasqlninja@microsoft.com](mailto:datasqlninja@microsoft.com).
- Read the [Azure Database Migration Guides](#).

## Related resources

- [Replicate and sync mainframe data in Azure](#)
- [Modernize mainframe and midrange data](#)
- [Migrate IBM mainframe applications to Azure with TmaxSoft OpenFrame](#)
- [Unisys mainframe migration with Avanade AMT](#)

# SMA OpCon in Azure

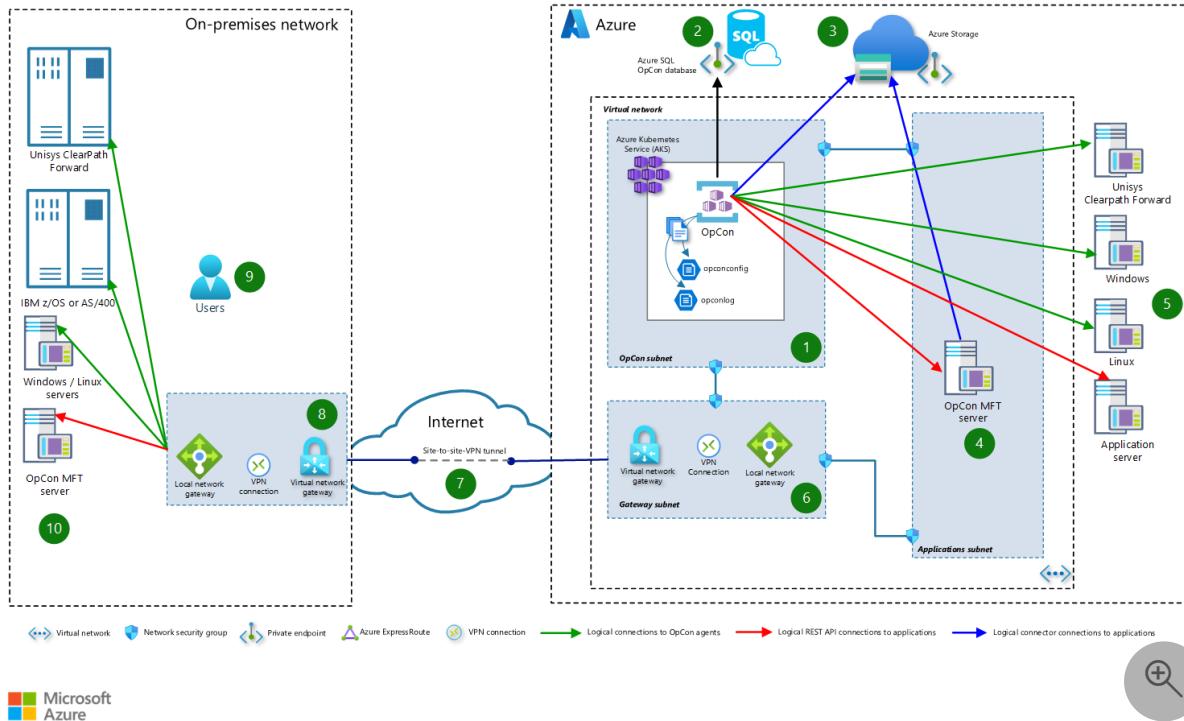
Azure Kubernetes Service (AKS)   Azure Private Link   Azure SQL Database   Azure Storage  
Azure VPN Gateway

## 💡 Solution ideas

This article is a solution idea. If you'd like us to expand the content with more information, such as potential use cases, alternative services, implementation considerations, or pricing guidance, let us know by providing [GitHub feedback](#).

This article presents a solution for automating workloads that run on various types of servers and systems throughout an enterprise. The solution uses OpCon from SMA Technologies in a Kubernetes configuration in Azure. From a single automation control point, OpCon facilitates workflows across the enterprise—both on-premises and in Azure.

## Architecture



Download a [Visio file](#) of this architecture.

# Workflow

1. An OpCon container provides core services, which are deployed within Azure Kubernetes Service (AKS). These core services include Solution Manager, a web-based user interface. Users can interact with the entire OpCon environment by using Solution Manager. Besides other components, the environment includes:
  - Persistent volumes that store logs and configuration information and provide data persistence across container restarts. For these volumes, the solution uses Azure Files, which is configured in the `StorageClass` value.
  - The OpCon database.
  - Virtual machines (VMs) that run workloads.
2. The solution uses Azure SQL Database as the OpCon database. The core services have access to this database through an Azure Private Link private endpoint.
3. OpCon core services use OpCon connector technology to interact with Azure Storage and manage data in Azure Blob Storage. OpCon Managed File Transfer also provides support for Storage.
4. The Applications subnet contains an OpCon Managed File Transfer server that provides comprehensive file-transfer functionality. Capabilities include compression, encryption, decryption, decompression, file watching, and enterprise-grade automated file routing.
5. Azure VMs make up the application infrastructure. The placement of these VMs in subnets and virtual networks is flexible. For more information, see [Component placement](#).
  - To manage workloads on these VMs and on-premises legacy systems, OpCon core services communicate with OpCon agents that are installed on the VMs. The core services communicate with on-premises systems through a site-to-site connection on a virtual network gateway.
  - OpCon core services communicate directly with applications that provide REST API endpoints. These applications don't need extra software to connect to the core services. With on-premises systems, the communication uses REST API connectivity options and travels via a virtual network gateway.
6. In a hybrid environment, the Gateway subnet uses a site-to-site VPN tunnel to help secure the connection between the on-premises environment and the Azure cloud environment.

7. The gateway includes a cross-premises IPsec/IKE VPN tunnel connection between Azure VPN Gateway and an on-premises VPN device. All data that passes between the Azure cloud and the on-premises environment is encrypted in this site-to-site private tunnel as it crosses the internet.
8. A local network gateway in the on-premises environment represents the gateway on the on-premises end of the tunnel. The local network gateway holds configuration information that's needed to build a VPN tunnel and to route traffic from or to on-premises subnets.
9. All user requests are routed via the gateway connection to the OpCon core services environment. Through that access, users interact with Solution Manager for:
  - OpCon administration.
  - OpCon Managed File Transfer administration.
  - OpCon workflow development, execution, and monitoring.
  - Self Service, an OpCon interface for running tasks.
  - Vision, the OpCon task dashboard.
  - OpCon Managed File Transfer Central Application, a dashboard and query application.
10. OpCon agents and application REST API endpoints are installed on legacy systems in the on-premises environment. OpCon core services use the site-to-site connection on the virtual network gateway to communicate with those agents and endpoints.

Throughout the solution, you can use network security groups to limit traffic flow between subnets.

## Components

- [Azure Virtual Machines](#) is one of several types of on-demand, scalable computing resources that Azure offers. An Azure VM gives you the flexibility of virtualization but eliminates the maintenance demands of physical hardware. With Azure, you can choose Windows or Linux VMs.
- [Azure Virtual Network](#) is the fundamental building block for your private network in Azure. Through Virtual Network, Azure resources like VMs can securely communicate with each other, the internet, and on-premises networks. An Azure virtual network is like a traditional network that operates in a datacenter. But an Azure virtual network also provides scalability, availability, isolation, and other benefits of the Azure infrastructure.

- [Private Link](#) provides a private endpoint in a virtual network. You can use the private endpoint to connect to Azure platform as a service (PaaS) services like Storage and SQL Database or to customer or partner services.
- [Storage](#) offers highly available, scalable, secure cloud storage for data, applications, and workloads.
- [Azure Files](#) is a service that's part of Storage. Azure Files offers fully managed file shares in the cloud that are accessible via the industry-standard Server Message Block (SMB) protocol. Azure file shares can be mounted concurrently by cloud or on-premises deployments of Windows, Linux, and macOS clients.
- [Blob Storage](#) is a service that's part of Storage. Blob Storage offers optimized cloud object storage for large amounts of unstructured data. This service is a good fit for high-performance computing, machine learning, and cloud-native workloads.
- [VPN Gateway](#) is a specific type of virtual network gateway. You can use VPN Gateway to transmit encrypted traffic. That traffic can flow between an Azure virtual network and an on-premises location over the public internet. It can also flow between Azure virtual networks over the Azure backbone network.
- [Azure ExpressRoute](#) extends your on-premises networks into the Microsoft cloud over a private connection that's facilitated by a connectivity provider. With ExpressRoute, you can establish connections to cloud services, such as Microsoft Azure and Microsoft 365.
- [Azure Site Recovery](#) helps ensure business continuity by keeping business apps and workloads running during outages. Site Recovery can replicate workloads that run on physical machines and VMs from a primary site to a secondary location. When an outage occurs at your primary site, you fail over to a secondary location and access apps from there. After the primary location is running again, you can fail back to it.
- [Azure SQL](#) is a family of Azure databases that are powered by the SQL Server engine. Azure SQL includes SQL Server on Azure Virtual Machines, Azure SQL Managed Instance, and SQL Database.
- [SQL Database](#) is a fully managed PaaS database engine with AI-powered, automated features. The OpCon back end can use SQL Database to manage OpCon entries.
- [SQL Managed Instance](#) is an intelligent and scalable cloud database service that combines the broadest SQL Server engine compatibility with all the benefits of a

fully managed and evergreen PaaS. The OpCon back end can use SQL Managed Instance to manage OpCon entries.

- [OpCon](#) core services run in a Linux container within a Kubernetes replica set. This solution uses SQL Database for the OpCon database.
- [OpCon Self Service](#) is a web-based implementation that provides a way for users to run on-demand tasks and optionally enter arguments within an OpCon environment.
- [OpCon Vision](#) provides a dashboard for monitoring OpCon tasks. The dashboard displays a logical representation of the tasks across all flows. Vision uses tags to group associated tasks together. When problems occur, you can drill down from the dashboard to failed tasks. Vision also provides a way to set SLA values for each group. The dashboard gives early warning when defined SLA values might not be met.
- [OpCon Managed File Transfer](#) provides managed file transfer services within an OpCon environment. The OpCon Managed File Transfer solution provides file transfer and monitoring functionality across an enterprise by using an integrated managed file transfer agent and a file transfer server.

## Alternatives

The following sections describe alternatives to consider when you implement the solution.

## Component placement

The placement of the VMs and OpCon database is flexible.

- The application subnet can include the application VMs. You can also install the application servers in multiple subnets or virtual networks. Use this approach when you want to create separate environments for different types of servers, such as web and application servers.
- You can place the database inside or outside the OpCon subnet.

## SQL Managed Instance

Instead of using SQL Database, you can use SQL Managed Instance as the OpCon database. You can install the SQL managed instance in the OpCon subnet. Alternatively,

you can install the managed instance in a separate subnet that you use exclusively for SQL managed instances in the existing virtual network.

## ExpressRoute

Instead of using VPN Gateway and a site-to-site VPN tunnel, you can use ExpressRoute, which uses a connectivity provider to establish a private connection to the Microsoft global network. ExpressRoute connections don't go over the public internet.

We recommend ExpressRoute for hybrid applications that run large-scale business-critical workloads that require a high degree of scalability and resiliency.

## Scenario details

The core OpCon module that facilitates workloads is the Schedule Activity Monitor (SAM). This module communicates with agents on target systems to schedule and monitor tasks. SAM also receives external events. You can install OpCon agents on the following platforms:

- Windows
- Linux or Unix
- Unisys ClearPath Forward mainframes (MCP and 2200)
- IBM z/OS
- IBM AIX

SAM draws the various platforms together under one automation umbrella.

You can install OpCon in an Azure cloud environment. OpCon supports cloud-only infrastructures and also hybrid infrastructures that contain cloud and on-premises systems.

The OpCon software is available from Docker Hub as Docker images that you can deploy in a cloud environment. For the Azure cloud, this solution uses AKS to deploy the OpCon environment within a Kubernetes cluster. SQL Database is used as the database.

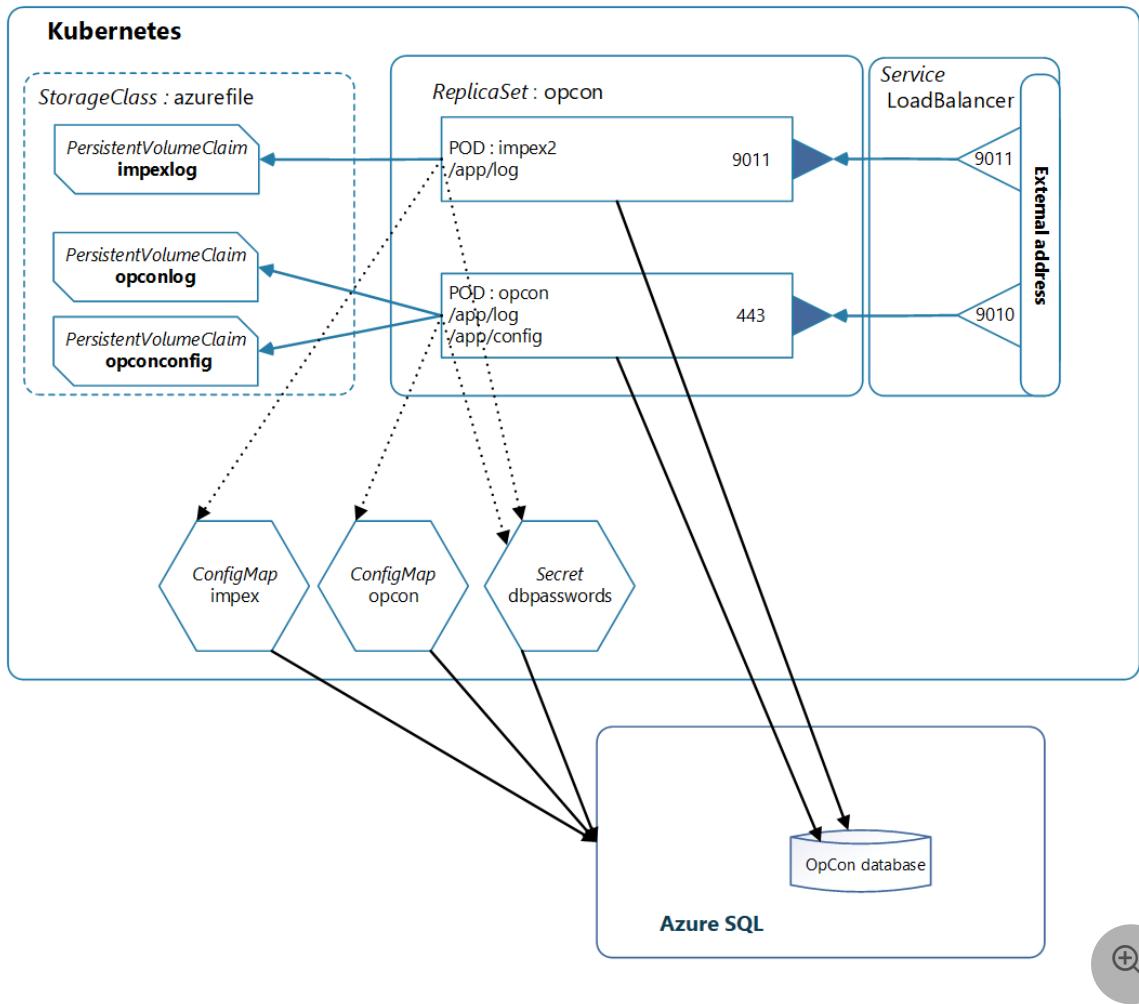
For hybrid environments, VPN Gateway provides a secure link between cloud infrastructure and on-premises infrastructure.

The implementation uses a single virtual network and multiple subnets to support various functions. You can use network security groups to filter network traffic between Azure resources in the virtual network.

# AKS configuration

The deployed OpCon environment consists of two pods within a single replica set and an instance of SQL Database. A load balancer controls access to the pods. The load balancer maps external addresses and ports to internal REST API server addresses and ports.

The following diagram shows configuration requirements for an environment with two pods: OpCon and Impex2. The diagram also shows the relationship between various definitions in the Kubernetes configuration YAML file.



[Download a Visio file](#) of this architecture.

The following table provides detailed information about each definition.

[Expand table](#)

Kind	Value	Description
Secret	dbpasswords	Contains the database passwords that are required to connect to the OpCon database.
ConfigMap	opcon	Contains the OpCon REST API information, the time zone, and the language information. Also contains OpCon database information, such as the address, the database name, and the database user.
ConfigMap	impex	Contains the Impex2 REST API information. Also contains OpCon database information, such as the address, the database name, and the database user.
PersistentVolumeClaim	opconconfig	Contains various .ini files and the OpCon license file.
PersistentVolumeClaim	opconlog	Contains the log files that are associated with the OpCon environment.
PersistentVolumeClaim	impexlog	Contains the log files that are associated with the Impex2 environment.
ReplicaSet	opcon	Specifies the OpCon and Impex2 container definitions that reference the previously defined Secret, ConfigMap, and PersistentVolumeClaim definitions.
Service	loadbalancer	Defines the mapping of the internal REST API ports for the OpCon and Impex2 REST servers to external addresses and ports.

## Potential use cases

Many scenarios can benefit from this solution:

- Workload automation and orchestration across an entire IT enterprise
- Disaster recovery automation
- Cross-platform file transfers
- IT environment operations

- Batch scheduling
- Running self-service automation workflows
- Server update automation and deployment
- Patch management automation and deployment
- Automation of the provisioning and decommissioning of Azure resources
- Monitoring an entire IT environment from a single interface
- Codifying repeatable or on-demand processes

## Deploy this scenario

You can use the following template to deploy the OpCon environment within an AKS cluster.

```

yml

#
# Full OpCon deployment for Kubernetes
#
# This deployment uses Azure SQL Database.
#
apiVersion: v1
kind: Secret
metadata:
  name: dbpasswords
stringData:
  saPassword: ""
  dbPassword: ""
  sqlAdminPassword: ""
  dbPasswordEncrypted: ""

---
# OpCon environment values
apiVersion: v1
kind: ConfigMap
metadata:
  name: opconenv
data:
  DB_SERVER_NAME: "sqlopcon.database.windows.net"
  DATABASE_NAME: "opcon"
  DB_USER_NAME: "opconadmin"
  SQL_ADMIN_USER: "opconadmin"
  API_USES_TLS: "true"
  CREATE_API_CERTIFICATE: "true"
  DB_SETUP: "true"
  TZ: "America/Chicago"
  LANG: "en_US.utf-8"
  LICENSE: ""

---
# Impex environment values
apiVersion: v1

```

```
kind: ConfigMap
metadata:
  name: impexenv
data:
  opcon.server.name: "sqlopcon.database.windows.net"
  opcon.db.name: "opcon"
  opcon.db.user: "opconadmin"
  web.port: "9011"
  web.ssl: "true"
  system.debug: "false"
  TZ: "America/Chicago"
  LANG: "en_US.utf-8"
---
# OpCon persistent storage for configuration information
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: opconconfig
spec:
  accessModes:
  - ReadWriteOnce
  resources:
    requests:
      storage: 100Mi
---
# OpCon persistent storage for log information
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: opconlog
spec:
  accessModes:
  - ReadWriteOnce
  resources:
    requests:
      storage: 100Mi
---
# Impex persistent storage for log information
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: impexlog
spec:
  accessModes:
  - ReadWriteOnce
  resources:
    requests:
      storage: 100Mi
---
# OpCon and deploy pods in a single replica set
apiVersion: apps/v1
kind: ReplicaSet
metadata:
  name: opcon
spec:
```

```
replicas: 1
selector:
  matchExpressions:
    - key: app
      operator: In
      values:
        - opconservices
template:
  metadata:
    labels:
      app: opconservices
spec:
  containers:
    - env:
        - name: DB_PASSWORD
          valueFrom:
            secretKeyRef:
              name: dbpasswords
              key: dbPassword
        - name: SQL_ADMIN_PASSWORD
          valueFrom:
            secretKeyRef:
              name: dbpasswords
              key: sqlAdminPassword
  envFrom:
    - configMapRef:
        name: opconenv
  image: smatechnologies/opcon-server:22.0-latest
  name: opcon
  ports:
    - containerPort: 443
      protocol: TCP
  volumeMounts:
    - name: opconconfig
      mountPath: /app/config
    - name: uat-opconlog
      mountPath: /app/log
  - env:
      - name: opcon.db.password
        valueFrom:
          secretKeyRef:
            name: dbpasswords
            key: dbPasswordEncrypted
  envFrom:
    - configMapRef:
        name: impexenv
  image: smatechnologies/deploy-impex2:22.0-latest
  name: impex
  volumeMounts:
    - name: impexlog
      mountPath: /app/log
  hostname: opcon
  volumes:
    - name: opconconfig
      persistentVolumeClaim:
```

```
        claimName: opconconfig
      - name: opconlog
        persistentVolumeClaim:
          claimName: opconlog
      - name: impexlog
        persistentVolumeClaim:
          claimName: impexlog
    ---
# OpCon service
apiVersion: v1
kind: Service
metadata:
  name: lbopcon
spec:
  type: LoadBalancer
  ports:
    - name: apiport
      port: 9010
      targetPort: 443
    - name: impexport
      port: 9011
      targetPort: 9011
  selector:
    app: opconservices
```

## Contributors

*This article is maintained by Microsoft. It was originally written by the following contributors.*

- [Philip Brooks](#) | Senior Program Manager
- [Bertie van Hinsbergen](#) | Principal Automation Consultant

*To see non-public LinkedIn profiles, sign in to LinkedIn.*

## Next steps

- [What is Azure Kubernetes Service?](#)
- [What is a private endpoint?](#)
- [Network security groups](#)
- [Quickstart: Set up disaster recovery to a secondary Azure region for an Azure VM](#)

For more information about this solution:

- Contact [legacy2azure@microsoft.com](mailto:legacy2azure@microsoft.com).
- Contact [SMA](#). A Microsoft Gold-level partner, [SMA Technologies](#) is a leader in the IT automation space. SMA is dedicated to the single purpose of giving time

back to clients and their employees by automating processes, applications, and workflows.

## Related resources

- [Unisys ClearPath Forward OS 2200 enterprise server virtualization on Azure](#)
- [Unisys ClearPath Forward MCP mainframe rehost to Azure using Unisys virtualization](#)
- [Azure Automation in a hybrid environment](#)
- [Manage hybrid Azure workloads using Windows Admin Center](#)

# Management and governance architecture design

Article • 06/30/2023

Management and governance includes critical tasks like:

- The monitoring, auditing, and reporting of security and business requirements.
- Implementing backup, disaster recovery, and high availability.
- Ensuring compliance with internal requirements and external regulations.
- The protection of sensitive data.

Azure provides a wide range of services to help you with management and governance.

Here are a few examples:

- [Azure Attestation](#) . Remotely verify the trustworthiness of a platform and the integrity of the binaries running inside it.
- [Azure confidential ledger](#) . Store and process confidential data with confidence.
- [Azure Purview](#) . Govern, protect, and manage your data.
- [Azure Policy](#) . Achieve real-time cloud compliance at scale with consistent resource governance.
- [Azure Stack](#) . Place technologies and services in appropriate locations, based on your business requirements. Meet custom compliance, sovereignty, and data gravity requirements.
- [Azure Backup](#) . Define backup policies and provide protection for a wide range of enterprise workloads.
- [Azure Site Recovery](#) . Keep your business running with built-in disaster recovery.
- [Azure Archive Storage](#) . Store rarely accessed data.
- [Azure Monitor](#) . Get full observability into your applications, infrastructure, and network.

## Introduction to management and governance on Azure

If you're new to management and governance on Azure, the best way to learn more is with [Microsoft Learn training](#), a free online training platform. Microsoft Learn provides interactive training for Microsoft products and more.

Here are some resources to get you started:

- Learning path: [Manage information protection and governance](#)

- Module: [Design an enterprise governance strategy](#)
- Module: [Design a solution for backup and disaster recovery](#)

## Path to production

The following sections provide links to reference architectures in some key management and governance categories:

### Backup

- [Azure Backup architecture and components](#)
- [Support matrix for Azure Backup](#)
- [Backup cloud and on-premises workloads to cloud](#)

### Disaster recovery

- [Azure to Azure disaster recovery architecture](#)
- [Support matrix for Azure VM disaster recovery between Azure regions](#)
- [Integrate Azure ExpressRoute with disaster recovery for Azure VMs](#)
- [Recover from a region-wide service disruption](#)
- [Move Azure VMs to another Azure region](#)
- [Business continuity and disaster recovery \(BCDR\) for Azure VMware Solution enterprise-scale scenario](#)
- [Enterprise-scale disaster recovery](#)
- [SMB disaster recovery with Azure Site Recovery](#)
- [SMB disaster recovery with Double-Take DR](#)
- [Disaster recovery for enterprise bots](#)
- [Use Azure Stack HCI stretched clusters for disaster recovery](#)

### High availability

- [Build high availability into your BCDR strategy](#)
- [High availability and disaster recovery scenarios for IaaS apps](#)
- [High availability enterprise deployment using App Service Environment](#)
- [Baseline zone-redundant web application](#)
- [Highly available multi-region web application](#)
- [Deploy highly available NVAs](#)
- [Highly available SharePoint farm](#)
- [Run a highly available SharePoint Server 2016 farm in Azure](#)
- [Build solutions for high availability using availability zones](#)

## Compliance and governance

- [Manage virtual machine compliance](#)
- [End-to-end governance in Azure when using CI/CD](#)
- [Introduction of an AKS regulated cluster for PCI-DSS 3.2.1](#)

## Hybrid management

- [Azure Arc hybrid management and deployment for Kubernetes clusters](#)
- [Azure Automation in a hybrid environment](#)
- [Azure Automation update management](#)
- [Back up files and applications on Azure Stack Hub](#)
- [Disaster recovery for Azure Stack Hub virtual machines](#)
- [Hybrid availability and performance monitoring](#)
- [Manage configurations for Azure Arc-enabled servers](#)
- [Manage hybrid Azure workloads using Windows Admin Center](#)

## Update management

- [Plan deployment for updating Windows VMs in Azure](#)
- [Azure Automation update management](#)

## Best practices

The Azure Well-Architected Framework is a set of guiding tenets that you can use to improve the quality of your architectures. For management and governance best practices, see:

- [Regulatory compliance](#)
- [Administrative account security](#)

For additional guidance, see:

- [Design area: Management for Azure environments](#)
- [Governance best practices](#)

## Stay current with management and governance

Get the latest updates on [Azure management](#) and [Azure governance](#) technologies.

# Additional resources

Following are a few more management and governance architectures to consider:

- [Archive on-premises data to the cloud](#)
- [Management and monitoring for an Azure VMware Solution enterprise-scale scenario](#)
- [Computer forensics chain of custody in Azure](#)
- [Deploy a line-of-business application using Azure App Service Environment v3](#)
- [Centralized app configuration and security](#)

## AWS or Google Cloud professionals

- [AWS to Azure services comparison - Management and governance](#)
- [Google Cloud to Azure services comparison - Management](#)

# Recommendations for establishing a security baseline

Article • 11/14/2023

Applies to Azure Well-Architected Framework Security checklist recommendation:

**SE:01 Establish a security baseline aligned to compliance requirements, industry standards, and platform recommendations. Regularly measure your workload architecture and operations against the baseline to sustain or improve your security posture over time.**

This guide describes the recommendations for establishing a security baseline. A security baseline is a document that specifies your organization's bare minimum security requirements and expectations across a range of areas. A good security baseline helps you:

- Keep your data and systems secure.
- Comply with regulatory requirements.
- Minimize risk of oversight.
- Reduce the likelihood of breaches and subsequent business effects.

Security baselines should be published widely throughout your organization so that all stakeholders are aware of the expectations.

This guide provides recommendations about setting a security baseline that's based on internal and external factors. Internal factors include business requirements, risks, and asset evaluation. External factors include industry benchmarks and regulatory standards.

## Definitions

Term	Definition
Baseline	The minimum level of security affordances that a workload must have to avoid being exploited.
Benchmark	A standard that signifies the security posture that the organization aspires to. It's evaluated, measured, and improved over time.
Controls	Technical or operational controls on the workload that help prevent attacks and increase attacker costs.
Regulatory requirements	A set of business requirements, driven by industry standards, that laws and authorities impose.

# Key design strategies

A security baseline is a structured document that defines a set of security criteria and capabilities that the workload must fulfill in order to increase security. In a more mature form, you can extend a baseline to include a set of policies that you use to set guardrails.

The baseline should be considered the standard for measuring your security posture. The goal should always be full attainment while keeping a broad scope.

Your security baseline should never be an ad-hoc effort. Industry standards, compliance (internal or external) or regulatory requirements, regional requirements, and the cloud platform benchmarks are main drivers for the baseline. Examples include Center for Internet Security (CIS) Controls, National Institute of Standards and Technology (NIST), and platform-driven standards, such as Microsoft cloud security benchmark (MCSB). All of these standards are considered a starting point for your baseline. Build the foundation by incorporating security requirements from the business requirements.

For links to the preceding assets, see [Related links](#).

Create the baseline by gaining consensus among business and technical leaders. The baseline shouldn't be restricted to technical controls. It should also include the operational aspects of managing and maintaining the security posture. So, the baseline document also serves as the organization's commitment to investment toward workload security. The security baseline document should be distributed widely within your organization to ensure there's awareness about the workload's security posture.

As the workload grows and the ecosystem evolves, it's vital to keep your baseline in sync with the changes to ensure the fundamental controls are still effective.

Creating a baseline is a methodical process. Here are some recommendations about the process:

- **Asset inventory.** Identify stakeholders of workload assets and the security objectives for those assets. In the asset inventory, classify by security requirements and criticality. For information about data assets, see [Recommendations on data classification](#).
- **Risk assessment.** Identify potential risks associated with each asset and prioritize them.
- **Compliance requirements.** Baseline any regulatory or compliance for those assets and apply industry best practices.

- **Configuration standards.** Define and document specific security configurations and settings for each asset. If possible, templatize or find a repeatable, automated way to apply the settings consistently across the environment.
- **Access control and authentication.** Specify the role-based access control (RBAC) and multifactor authentication (MFA) requirements. Document what *just enough* access means at the asset level. Always start with the principle of least privilege.
- **Patch management.** Apply latest versions on all the resource types to strengthen against attack.
- **Documentation and communication.** Document all configurations, policies, and procedures. Communicate the details to the relevant stakeholders.
- **Enforcement and accountability.** Establish clear enforcement mechanisms and consequences for noncompliance with the security baseline. Hold individuals and teams accountable for maintaining security standards.
- **Continuous monitoring.** Assess the effectiveness of the security baseline through observability and make improvements overtime.

## Composition of a baseline

Here are some common categories that should be part of a baseline. The following list isn't exhaustive. It's intended as an overview of the document's scope.

### Regulatory compliance

A workload might be subject to regulatory compliance for specific industry segments, there might be some geographic restrictions, and so on. It's key to understand the requirements as given in the regulatory specifications because those influence the design choices and in some cases must be included in the architecture.

The baseline should include regular evaluation of the workload against regulatory requirements. Take advantage of the platform-provided tools, such as Microsoft Defender for Cloud, which can identify areas of noncompliance. Work with the organization's compliance team to make sure all requirements are met and maintained.

### Architecture components

The baseline needs prescriptive recommendations for the main components of the workload. These usually include technical controls for networking, identity, compute,

and data. Reference the security baselines provided by the platform and add the missing controls to the architecture.

Refer to [Example](#).

## Development processes

The baseline must have recommendations about:

- System classification.
- The approved set of resource types.
- Tracking the resources.
- Enforcing policies for using or configuring resources.

The development team needs to have a clear understanding of the scope for security checks. For example, threat modeling is a requirement in making sure that potential threats are identified in code and in deployment pipelines. Be specific about static checks and vulnerability scanning in your pipeline and how regularly the team needs to perform those scans.

For more information, see [Recommendations on threat analysis](#).

The development process should also set standards on various testing methodologies and their cadence. For more information, see [Recommendations on security testing](#).

## Operations

The baseline must set standards on using threat detection capabilities and raising alerts on anomalous activities that indicate actual incidents. Threat detection needs to include all layers of the workload, including all the endpoints that are reachable from hostile networks.

The baseline should include recommendations for setting up incident response processes, including communication and a recovery plan, and which of those processes can be automated to expedite detection and analysis. For examples, see [Security baselines for Azure overview](#).

The incident response should also include a recovery plan and the requirements for that plan, such as resources for regularly taking and protecting backups.

You develop data breach plans by using industry standards and recommendations provided by the platform. The team then has a comprehensive plan to follow when a

breach is discovered. Also, check with your organization to see if there's coverage through cyberinsurance.

## Training

Develop and maintain a security training program to ensure the workload team is equipped with the appropriate skills to support the security goals and requirements. The team needs fundamental security training, but use what you can from your organization to support specialized roles. Role-based security training compliance and participation in drills are part of your security baseline.

## Use the baseline

Use the baseline to drive initiatives, such as:

- **Preparedness toward design decisions.** Create the security baseline and publish it before you start the architecture design process. Ensure team members are fully aware of your organization's expectations early, which avoids costly rework caused by a lack of clarity. You can use baseline criteria as workload requirements that the organization has committed to and design and validate controls against those constraints.
- **Measure your design.** Grade the current decisions against the current baseline. The baseline sets actual thresholds for criteria. Document any deviations that are deferred or deemed long-term acceptable.
- **Drive improvements.** While the baseline sets attainable goals, there are always gaps. Prioritize the gaps in your backlog and remediate based on prioritization.
- **Track your progress against the baseline.** Continuous monitoring of security measures against a set baseline is essential. Trend analysis is a good way of reviewing security progress over time and can reveal consistent deviations from the baseline. Use automation as much as possible, pulling data from various sources, internal and external, to address current issues and prepare for future threats.
- **Set guardrails.** Where possible, your baseline criteria must have guardrails. Guardrails enforce required security configurations, technologies, and operations, based on internal factors and external factors. Internal factors include business requirements, risks, and asset evaluation. External factors include benchmarks, regulatory standards, and threat environment. Guardrails help minimize the risk of inadvertent oversight and punitive fines for noncompliance.

Explore Azure Policy for custom options or use built-in initiatives like CIS benchmarks or Azure Security Benchmark to enforce security configurations and compliance requirements. Consider creating Azure Policies and initiatives out of baselines.

## Evaluate the baseline regularly

Continuously improve security standards incrementally towards the ideal state to ensure continual risk reduction. Conduct periodic reviews to ensure that the system is up-to-date and in compliance with external influences. Any change to the baseline must be formal, agreed upon, and sent through proper change management processes.

Measure the system against the new baseline and prioritize remediations based on their relevance and effect on the workload.

Ensure that the security posture doesn't degrade over time by instituting auditing and monitoring compliance with organizational standards.

## Azure facilitation

The Microsoft cloud security benchmark (MCSB) is a comprehensive security best practice framework that you can use as a starting point for your security baseline. Use it along with other resources that provide input to your baseline.

For more information, see [Introduction to the Microsoft cloud security benchmark](#).

Use the Microsoft Defender for Cloud (MDC) regulatory compliance dashboard to track those baselines and be alerted if a pattern outside of a baseline is detected. For more information, see the [Customize the set of standards in your regulatory compliance dashboard](#).

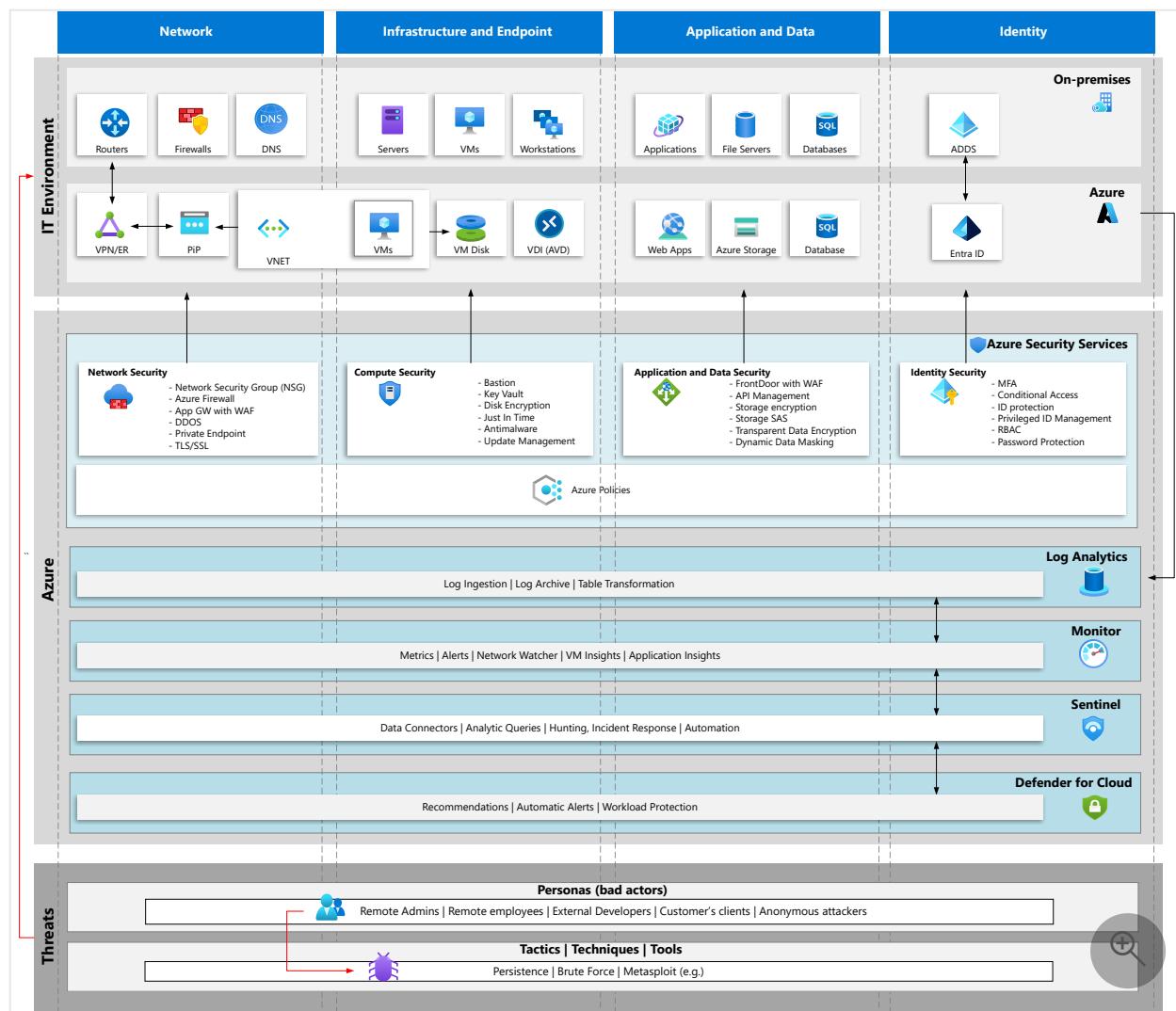
Other features that help in establishing and improving the baseline:

- [Create custom Azure security policies](#)
- [Understand security policies, initiatives, and recommendations](#)
- [Regulatory compliance checks](#)

## Example

This logical diagram shows an example security baseline for architectural components that encompass network, infrastructure, endpoint, application, data, and identity to

demonstrate how a common IT environment may be securely protected. Other recommendation guides build on this example.



## Infrastructure

A common IT environment, with an on-premises layer with basic resources.

## Azure Security services

Azure security services and features by the types of resources they protect.

## Azure security monitoring services

The monitoring services available on Azure that go beyond simple monitoring services, including security information event management (SIEM) and security orchestration automated response (SOAR) solutions and Microsoft Defender for Cloud.

## Threats

This layer brings a recommendation and reminder that threats may be mapped according to your organization's concerns regarding threats, regardless of the methodology or matrix-like Mitre Attack Matrix or Cyber Kill chain.

# Organizational alignment

Cloud Adoption Framework provides guidance for central teams about establishing a baseline with a suggested template:

- [Security Baseline discipline overview](#)
- [Security Baseline discipline template](#)

## Related links

- [Microsoft compliance](#)
- [Security baselines for Azure overview](#)
- [What is incident response? Plan and steps ↗](#)
- [Azure Security benchmarks](#)

## Community links

- [CIS Microsoft Azure Foundations Benchmark ↗](#)
- [Cybersecurity framework | NIST ↗](#)

## Security checklist

Refer to the complete set of recommendations.

[Security checklist](#)

# Plan deployment for updating Windows VMs in Azure

Azure

Azure Firewall

Azure Virtual Machines

Azure Virtual Network

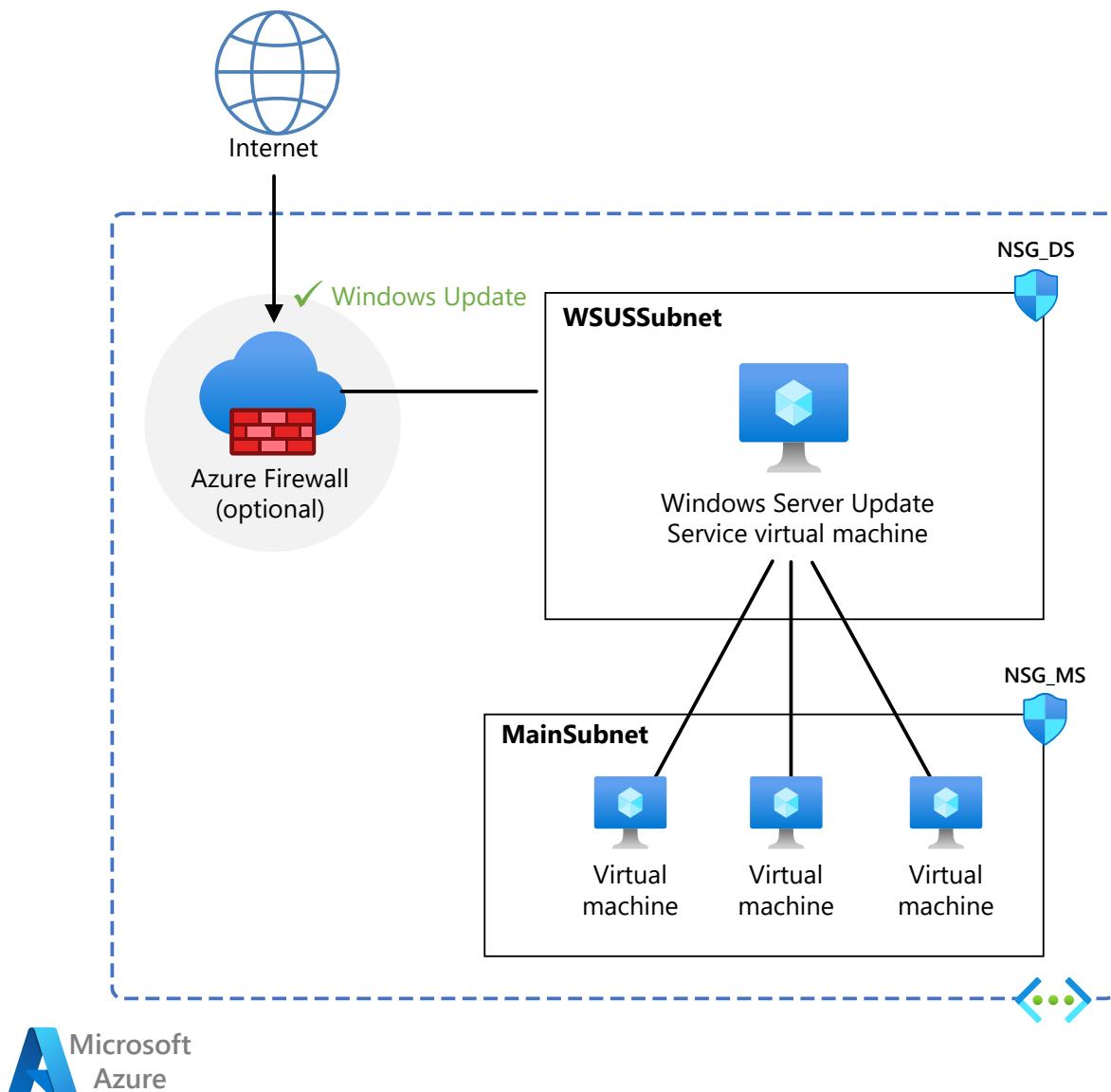
If you've locked down your Azure virtual network from the internet, you can still get Windows updates without jeopardizing security and opening up access to the internet as a whole. This article contains recommendations on how you can set up a perimeter network, also called a DMZ, to host a Windows Server Update Service (WSUS) instance to securely update virtual networks without internet connectivity.

If you're using Azure Firewall, you can use the Windows Update FQDN tag in application rules to allow the required outbound network traffic through your firewall. For more information, see [FQDN tags overview](#).

To implement the recommendations in this article, you should be familiar with Azure services. The following sections describe the recommended deployment design, which uses a hub-and-spoke configuration in a single-region or multiregion configuration.

## Azure Virtual Network hub-and-spoke network topology

We recommend that you set up a hub-and-spoke model network topology by creating a perimeter network. Host the WSUS server on an Azure virtual machine that's in the hub between the internet and the virtual networks. The hub should have open ports. WSUS uses port 80 for HTTP protocol and port 443 for HTTPS protocol to obtain updates from Microsoft. The spokes are all the other virtual networks, which will communicate with the hub and not with the internet. You can accomplish this by creating a subnet, network security groups (NSGs), and Azure virtual network peering that allows WSUS traffic while blocking other internet traffic. This image illustrates an example of hub-and-spoke topology:



Download a [Visio file](#) of this architecture.

In this image:

- **WSUSSubnet** is the hub of the hub and spoke.
- **NSG\_DS** is a network security group rule that allows traffic for WSUS while blocking other internet traffic.
- **WSUS VM** is the Azure virtual machine that's configured to run WSUS.
- **MainSubnet** is a virtual network, a spoke, containing virtual machines.
- **NSG\_MS** is a network security group policy that allows traffic from WSUS VM but denies internet traffic.

You can reuse an existing server or deploy a new one that will be the WSUS server. For the WSUS VM, we recommend the following, at a minimum:

- **Operating system:** Windows Server 2016 or later.
- **Processor:** Dual core, 2 GHz or faster.

- **Memory:** 2 GB of RAM, in addition to the RAM required by the server and all other running services and software.
- **Storage:** 40 GB or more.
- **Access:** Access this virtual machine more securely by using just-in-time (JIT). See [Manage virtual machine access using just-in-time](#).

Your network will have more than one Azure virtual network, which can be in the same region or in different regions. You'll need to evaluate all Windows Server VMs to see if one can be used as a WSUS server. If you have thousands of VMs to update, we recommend dedicating a Windows Server VM to the WSUS role.

If all your virtual networks are in the same region, we suggest having one WSUS for every 18,000 VMs. This suggestion is based on a combination of the VM requirements, the number of client VMs being updated, and the cost of communicating between virtual networks. For more information on WSUS capacity requirements, see [Plan your WSUS deployment](#).

You can determine the cost of these configurations by using the [Azure pricing calculator](#). You'll need to provide the following information:

- Virtual machine:
  - Region: The region where your Azure virtual network is deployed.
  - Operating system: **Windows**
  - Tier: **Standard**
  - Instance: **D4 configuration**
  - Managed disks: **Standard HDD, 64 GB**
- Virtual network:
  - Type
    - **Same Region** if transfer is in the same region.
    - **Across Region** if you're moving data from one region to another.
  - Data Transfer: **2 GB**
  - Region
    - If transfer is within one region, choose the region the WSUS server and virtual networks are in.
    - If transfer crosses regions, the source virtual network region is where the WSUS server is. The destination virtual network region is where the data is going.
  - If you have multiple regions, you'll need to select **Virtual Network** multiple times.

Note that prices will vary by region.

# Manual deployment

After you either identify the Azure virtual network to use as the hub or determine you need to create a new Windows Server instance, you need to create an NSG rule. The rule will allow internet traffic, which allows Windows Update metadata and content to sync with the WSUS server that you'll create. Here are the rules that you need to add:

- Inbound/outbound NSG rule to allow traffic to and from the internet on port 80 (for content).
- Inbound/outbound NSG rule to allow traffic to and from the internet on port 443 (for metadata).
- Inbound/outbound NSG rule to allow traffic from the client VMs on port 8530 (default unless configured).

## Set up WSUS

There are two approaches you can use to set up your WSUS server:

- If you want to automatically set up a server that's configured to handle a typical workload with minimal administration required, you can use the PowerShell automation script.
- If you need to handle thousands of clients that run many different operating systems and languages, or if you want to configure WSUS in a way that the PowerShell script can't handle, you can set up WSUS manually. Both approaches are described later in this article.

You can also combine the two approaches by using the automation script to do most of the work and then using the WSUS administrative console to fine-tune the server settings.

## Set up WSUS by using the automation script

The Configure-WSUSServer script allows you to quickly set up a WSUS server that will automatically synchronize and approve updates for a chosen set of products and languages.

### Note

The script always sets up WSUS to use Windows Internal Database to store its update data. This speeds up setup and reduces administration complexity. But if your server will support thousands of client computers, especially if you also need

to support a wide variety of products and languages, you should set up WSUS manually instead so that you can use SQL Server as the database.

The latest version of this script is [available on GitHub](#).

You configure the script by using a JSON file. You can currently configure these options:

- Whether update payloads should be stored locally (and, if so, where they should be stored), or left on the Microsoft servers.
- Which products, update classifications, and languages should be available on the server.
- Whether the server should automatically approve updates for installation or leave updates unapproved unless an administrator approves them.
- Whether the server should automatically retrieve new updates from Microsoft, and, if so, how often.
- Whether Express update packages should be used. (Express update packages reduce server-to-client bandwidth at the expense of client CPU/disk usage and server-to-server bandwidth.)
- Whether the script should overwrite its previous settings. (Normally, to avoid inadvertent reconfiguration that might disrupt server operation, the script will run only once on a given server.)

Copy the script and its configuration file to local storage, and edit the configuration file to suit your needs.

### Warning

Be careful when you edit the configuration file. The syntax used for JSON configuration files is strict. If you inadvertently change the structure of the file rather than just the parameter values, the configuration file won't load.

You can run this script in one of two ways:

- You can run the script manually, from the WSUS VM.

The following command, run from an elevated Command Prompt window, will install and configure WSUS. It will use the script and configuration file in the current directory.

```
powershell.exe -ExecutionPolicy Unrestricted -File .\Configure-WSUSServer.ps1  
-WSUSConfigJson .\WSUS-Config.json
```

- You can use the [Custom Script Extension for Windows](#).

Copy the script and the JSON configuration file to your own storage container.

In typical VM and Azure Virtual Network configurations, the Custom Script Extension needs only the following two parameters to run the script correctly. (You need to replace the values shown here with the URLs for your storage locations.)

JSON

```
"fileUris":  
["https://mystorage.blob.core.windows.net/mycontainer/Configure-  
WSUSServer.ps1","https://mystorage.blob.core.windows.net/container/WSUS  
-Config.json"],  
"commandToExecute": "powershell.exe -ExecutionPolicy Unrestricted -File  
.\\Configure-WSUSServer.ps1 -WSUSConfigJson .\\WSUS-Config.json"
```

The script will start the initial synchronization needed to make updates available to client computers. But it won't wait for that synchronization to complete. Depending on the products, classifications, and languages you've selected, the initial synchronization might take several hours. All synchronizations after that should be faster.

## Set up WSUS manually

1. From your WSUS VM, open Server Manager and select **Add roles and features**.
2. Select **Next** until you get to the **Select server roles** page. Select **Windows Server Update Services**. Select **Add Features** when you're prompted with **Add features that are required for Windows Server Update Services?**
3. Select **Next** until you get to the **Select role services** page.
  - By default, you can use **WID Connectivity**.
  - Use **SQL Server Connectivity** if you need to support clients that use many different versions of Windows (for example, Windows 11 and Windows 10).
4. Select **Next** until you get to the **Content location selection** page. Enter the location where you want to store the updates.
5. Select **Next** until you get to the **Confirm installation selections** page. Select **Install**.
6. Open the installed Windows Server Update Services and select **Run**.
7. Select **Next** until you get to the **Connect to Upstream Server** page. Select **Start Connecting**.

8. Select **Next** until you get to the **Choose Languages** page. Choose the languages you need.
9. Select **Next** until you get to the **Choose Products** page. Choose the products you need.
10. Select **Next** until you get to the **Choose Classifications** page. Choose the updates you need.
11. Select **Next** until you get to the **Set Sync Schedule** page. Choose your sync preference.
12. Select **Next** until you get to the **Finished** page. Select **Begin initial synchronization** and then select **Next**.
13. Select **Next** until you get to the **What's Next** page and then select **Finish**.
14. If you select your WSUS name (for example, **WsusVM**) in the navigation pane, you should see that **Synchronization Status** is **Idle** and **Last synchronization result** is **Succeeded**.
15. In the navigation pane, select **Options > Computers > Use Group Policy or registry settings on computers**. Select **OK**.

During synchronization, WSUS determines if any new updates have been made available since the last time you synchronized. If it's your first time synchronizing WSUS, the metadata is downloaded immediately. The payload downloads only if local storage is turned on and the update is approved for at least one computer group.

 **Note**

Initial synchronization can take more than an hour. All synchronizations after that should be significantly faster.

## Configure virtual networks to communicate with WSUS

Next, set up Azure virtual network peering or global virtual network peering to communicate with the hub. We recommend that you set up a WSUS server in each region you've deployed to minimize latency.

On each Azure virtual network that's a spoke, you'll need to create an NSG policy that has these rules:

- An inbound/outbound NSG rule to allow traffic from the WSUS VM on port 8530 (default unless configured).
- An inbound/outbound NSG rule to deny traffic from the internet.

Next, create the Azure virtual network peering from the spoke to the hub.

## Client VM

- For extra security, you can remove a VM's associated public IP address. For more information, see [View, change settings for, or delete a public IP address](#).
- For information about how to access your virtual machine more securely by using JIT, see [Manage virtual machine access using just-in-time](#).

## Configure client virtual machines

WSUS can be used to update any virtual machine that runs Windows (except for the Home SKU). Complete the following steps on each client virtual machine to enable communication between WSUS and the client:

### From your client VM

1. Open Local Group Policy Editor (or Group Policy Management Editor).
2. Go to **Computer Configuration > Administrative Templates > Windows Components > Windows Update**.
3. Enable **Specify intranet Microsoft update service location**.
4. Enter the URL `http://\<WSUS name>:8530`. (You can find your WSUS name (for example, WsusVM) on the Update Services page.) It might take some time (up to few hours) for this setting to be reflected.
5. Go to **Settings > Update & Security > Windows Update**.
6. Select **Check for updates**.

### From your WSUS VM

1. Open **Windows Server Update Services**. You should be able to see your client VM listed under **Computers > All Computers**.
2. Select **Updates > All Updates**.
3. Set **Approval to Any Except Declined**.
4. Set **Status to Needed**. Now you can see all the updates needed for your client VM.

5. Right-click any of the updates and select **Approve**.

## Verification

1. On the client VM, go to **Settings > Update & Security > Windows Update**.
2. Select **Check for updates**. You should see an update with the same KB article number (for example, 4480056) that you approved from the WSUS VM.

If you're an administrator managing a large network, see [Configure automatic updates and update service location](#) for information about how to use Group Policy settings to automatically configure clients.

## WSUS deployment for multiple clouds

It's not possible to set up virtual network peering across public and private clouds. Networks that are deployed across public and private clouds will need to have at least one WSUS server in each cloud.

## Support notes

Currently, WSUS doesn't support synchronization with the Windows Home SKU.

## Azure Update Management

You can use the Update Management solution in Azure to manage and schedule operating system updates for VMs that are syncing against WSUS. The patch status of the VM (that is, which patches are missing) is assessed based on the source that the VM is configured to sync with. If the Windows VM is configured to report to WSUS, the results might differ from what Microsoft Update shows, depending on when WSUS last synced with Microsoft Update. After you configure your WSUS environment, you can enable Update Management. For more information, see [Update Management overview and onboarding steps](#).

## Contributors

*This article is maintained by Microsoft. It was originally written by the following contributors.*

Principal author:

- [Paul Reed](#) | Azure Compliance Senior Program Manager

## Next steps

- For more information on planning a deployment, see [Plan your WSUS deployment](#).
- For more information on managing WSUS, setting up a WSUS synchronization schedule, and more, see [WSUS administration](#).

## Related resources

- [Run a Windows VM on Azure](#)
- [Azure Automation update management](#)
- [Run SAP NetWeaver in Windows on Azure](#)
- [Manage hybrid Azure workloads using Windows Admin Center](#)
- [CI/CD for Azure VMs](#)

# Azure Backup architecture and components

Article • 01/31/2023

You can use the [Azure Backup service](#) to back up data to the Microsoft Azure cloud platform. This article summarizes Azure Backup architecture, components, and processes.

## What does Azure Backup do?

Azure Backup backs up the data, machine state, and workloads running on on-premises machines and Azure virtual machine (VM) instances. There are a number of Azure Backup scenarios.

## How does Azure Backup work?

You can back up machines and data by using a number of methods:

- **Back up on-premises machines:**
  - You can back up on-premises Windows machines directly to Azure by using the Azure Backup Microsoft Azure Recovery Services (MARS) agent. Linux machines aren't supported.
  - You can back up on-premises machines to a backup server - either System Center Data Protection Manager (DPM) or Microsoft Azure Backup Server (MABS). You can then back up the backup server to a Recovery Services vault in Azure.
- **Back up Azure VMs:**
  - You can back up Azure VMs directly. Azure Backup installs a backup extension to the Azure VM agent that's running on the VM. This extension backs up the entire VM.
  - You can back up specific files and folders on the Azure VM by running the MARS agent.
  - You can back up Azure VMs to the MABS that's running in Azure, and you can then back up the MABS to a Recovery Services vault.

Learn more about [what you can back up](#) and about [supported backup scenarios](#).

## Where is data backed up?

Azure Backup stores backed-up data in vaults - Recovery Services vaults and Backup vaults. A vault is an online-storage entity in Azure that's used to hold data, such as backup copies, recovery points, and backup policies.

Vaults have the following features:

- Vaults make it easy to organize your backup data, while minimizing management overhead.
- You can monitor backed-up items in a vault, including Azure VMs and on-premises machines.
- You can manage vault access with [Azure role-based access control \(Azure RBAC\)](#).
- You specify how data in the vault is replicated for redundancy:
  - **Locally redundant storage (LRS):** To protect your data against server rack and drive failures, you can use LRS. LRS replicates your data three times within a single data center in the primary region. LRS provides at least 99.999999999% (11 nines) durability of objects over a given year. [Learn more](#)
  - **Geo-redundant storage (GRS):** To protect against region-wide outages, you can use GRS. GRS replicates your data to a secondary region. [Learn more](#).
  - **Zone-redundant storage (ZRS):** replicates your data in [availability zones](#), guaranteeing data residency and resiliency in the same region. [Learn more](#)
- By default, Recovery Services vaults use GRS.

Recovery Services vaults have the following additional features:

- In each Azure subscription, you can create up to 500 vaults.

## Backup agents

Azure Backup provides different backup agents, depending on what type of machine is being backed up:

Agent	Details
MARS agent	<ul style="list-style-type: none"><li>• Runs on individual on-premises Windows Server machines to back up files, folders, and the system state.</li><li>• Runs on Azure VMs to back up files, folders, and the system state.</li><li>• Runs on DPM/MABS servers to back up the DPM/MABS local storage disk to Azure.</li></ul>
Azure VM extension	Runs on Azure VMs to back them up to a vault.

# Backup types

The following table explains the different types of backups and when they're used:

Backup type	Details	Usage
Full	A full backup contains the entire data source. Takes more network bandwidth than differential or incremental backups.	Used for initial backup.
Differential	A differential backup stores the blocks that changed since the initial full backup. Uses a smaller amount of network and storage, and doesn't keep redundant copies of unchanged data.  Inefficient because data blocks that are unchanged between later backups are transferred and stored.	Not used by Azure Backup.
Incremental	An incremental backup stores only the blocks of data that changed since the previous backup. High storage and network efficiency.  With incremental backup, there's no need to supplement with full backups.	Used by DPM/MABS for disk backups, and used in all backups to Azure. Not used for SQL Server backup.

# SQL Server backup types

The following table explains the different types of backups used for SQL Server databases and how often they're used:

Backup type	Details	Usage
Full backup	A full database backup backs up the entire database. It contains all the data in a specific database or in a set of filegroups or files. A full backup also contains enough logs to recover that data.	At most, you can trigger one full backup per day.  You can choose to make a full backup on a daily or weekly interval.

Backup type	Details	Usage
Differential backup	<p>A differential backup is based on the most recent, previous full-data backup.</p> <p>It captures only the data that's changed since the full backup.</p>	<p>At most, you can trigger one differential backup per day.</p> <p>You can't configure a full backup and a differential backup on the same day.</p>
Transaction log backup	<p>A log backup enables point-in-time restoration up to a specific second.</p>	<p>At most, you can configure transactional log backups every 15 minutes.</p>

## SAP HANA backup types

The following table explains the different types of backups used for SAP HANA databases and how often they're used:

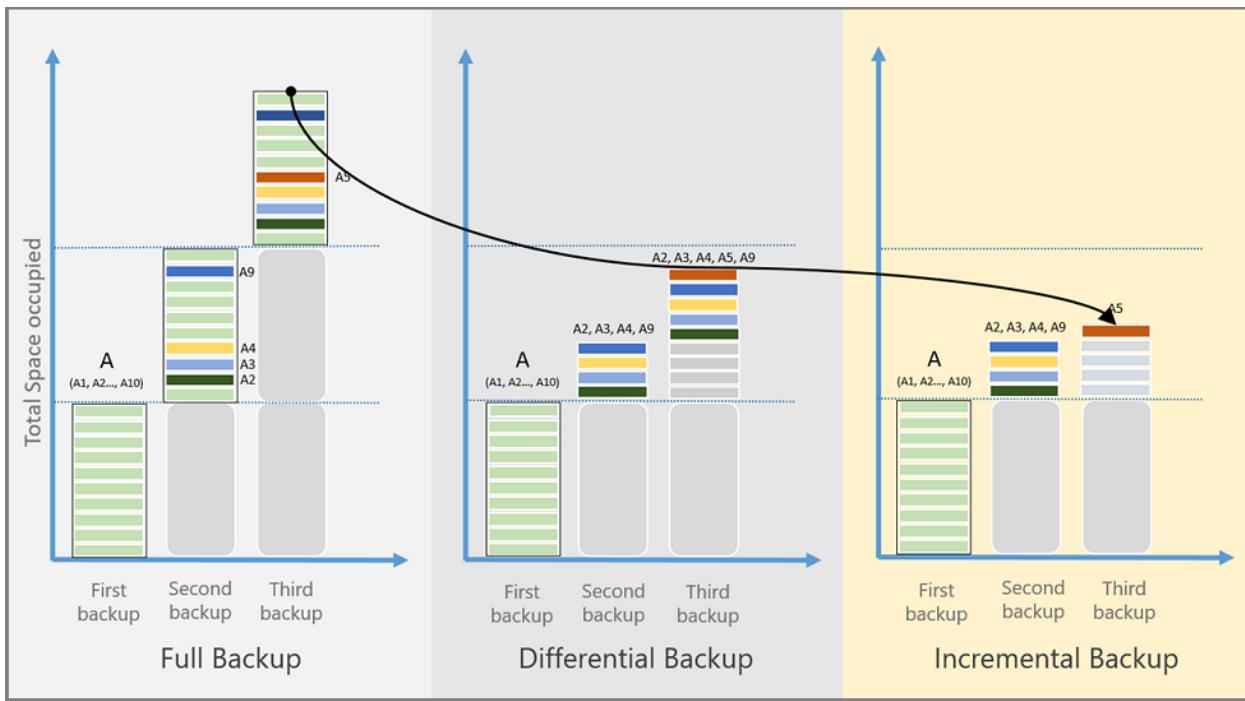
Backup type	Details	Usage
Full backup	<p>A full database backup backs up the entire database. This type of backup can be independently used to restore to a specific point.</p>	<p>At most, you can schedule one full backup per day.</p> <p>You can choose to schedule a full backup on a daily or weekly interval.</p>
Differential backup	<p>A differential backup is based on the most recent, previous full-data backup.</p> <p>It captures only the data that's changed since the previous full backup.</p>	<p>At most, you can schedule one differential backup per day.</p> <p>You can't configure a full backup and a differential backup on the same day.</p>

Backup type	Details	Usage
Incremental backup	<p>An incremental backup is based on the most recent, previous full/ differential/ incremental-data backup.</p> <p>It captures only the data that's changed since this previous data backup.</p>	<p>At most, you can schedule one incremental backup per day.</p> <p>You can't schedule both differential and incremental backups on a database, only one delta backup type can be scheduled.</p> <p>You can't configure a full backup and a differential backup on the same day.</p>
Transaction log backup	A log backup enables point-in-time restoration up to a specific second.	At most, you can configure transactional log backups every 15 minutes.

## Comparison of backup types

Storage consumption, recovery time objective (RTO), and network consumption varies for each type of backup. The following image shows a comparison of the backup types:

- Data source A is composed of 10 storage blocks, A1-A10, which are backed up monthly.
- Blocks A2, A3, A4, and A9 change in the first month, and block A5 changes in the next month.
- For differential backups, in the second month changed blocks A2, A3, A4, and A9 are backed up. In the third month, these same blocks are backed up again, along with changed block A5. The changed blocks continue to be backed up until the next full backup happens.
- For incremental backups, in the second month blocks A2, A3, A4, and A9 are marked as changed and transferred. In the third month, only changed block A5 is marked and transferred.



## Backup features

The following table summarizes the supported features for the different types of backup:

Feature	Direct Backup of Files and Folders (using MARS Agent)	Azure VM Backup	Machines or apps with DPM/MABS
Back up to vault	●	●	●
Back up to DPM/MABS disk, then to Azure			●
Compress data sent for backup	●	No compression is used when transferring data. Storage is inflated slightly, but restoration is faster.	●
Run incremental backup	●	●	●

Feature	Direct Backup of Files and Folders (using MARS Agent)	Azure VM Backup	Machines or apps with DPM/MABS
Back up deduplicated disks			For DPM/MABS servers deployed on-premises only.

Key



= Supported



= Partially Supported

<blank> = Not Supported

## Backup policy essentials

- A backup policy is created per vault.
- A backup policy can be created for the backup of following workloads: Azure VMs, SQL in Azure VMs, SAP HANA in Azure VMs and Azure file shares. The policy for files and folder backup using the MARS agent is specified in the MARS console.
  - Azure File Share
- A policy can be assigned to many resources. An Azure VM backup policy can be used to protect many Azure VMs.
- A policy consists of two components
  - Schedule: When to take the backup
  - Retention: For how long each backup should be retained.
- Schedule can be defined as "daily" or "weekly" with a specific point of time.
- Retention can be defined for "daily", "weekly", "monthly", "yearly" backup points.
  - "weekly" refers to a backup on a certain day of the week
  - "monthly" refers a backup on a certain day of the month
  - "yearly" refers to a backup on a certain day of the year
- Retention for "monthly", "yearly" backup points is referred to as Long Term Retention (LTR)
- When a vault is created, a "DefaultPolicy" is also created and can be used to back up resources.
- Any changes made to the retention period of a backup policy will be applied retroactively to all the older recovery points aside from the new ones.

## Impact of policy change on recovery points

- **Retention duration is increased / decreased:** When the retention duration is changed, the new retention duration is applied to the existing recovery points as

well. As a result, some of the recovery points will be cleaned up. If the retention period is increased, the existing recovery points will have an increased retention as well.

- **Changed from daily to weekly:** When the scheduled backups are changed from daily to weekly, the existing daily recovery points are cleaned up.
- **Changed from weekly to daily:** The existing weekly backups will be retained based on the number of days remaining according to the current retention policy.

## Additional reference

- Azure VM machine: How to [create](#) and [modify](#) policy.
- SQL Server database in Azure VM machine: How to [create](#) and [modify](#) policy.
- Azure File share: How to [create](#) and [modify](#) policy.
- SAP HANA: How to [create](#) and [modify](#) policy.
- MARS: How to [create](#) and [modify](#) policy.
- [Are there any limitations on scheduling backup based on the type of workload?](#)
- [What happens to the existing recovery points if I change the retention policy?](#)

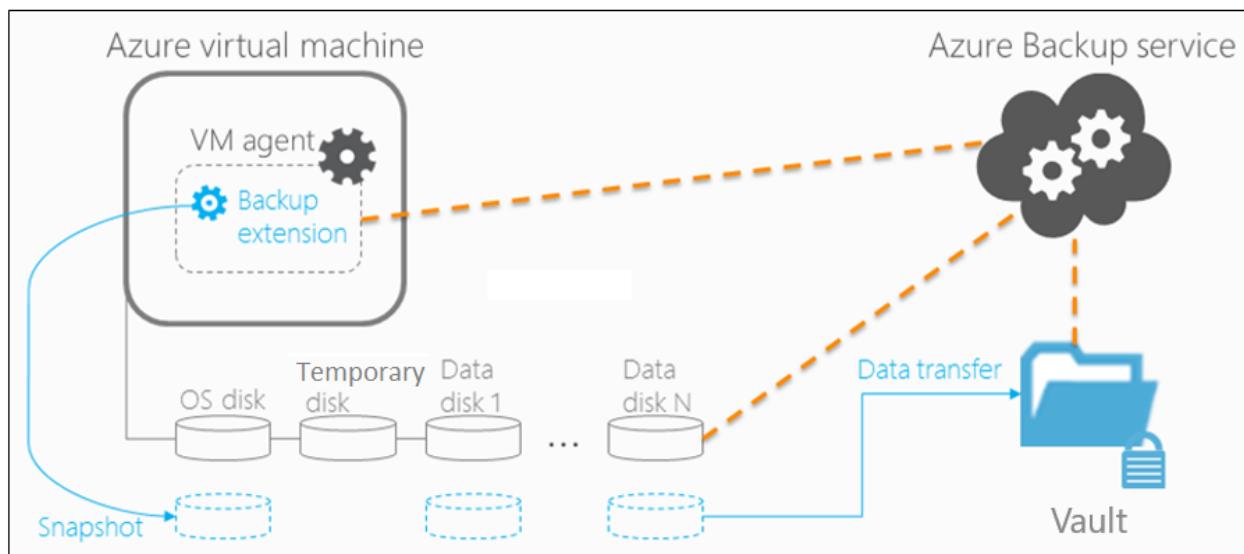
## Architecture: Built-in Azure VM Backup

1. For Azure VMs that are selected for backup, Azure Backup starts a backup job according to the backup schedule you specify.
2. During the first backup, a backup extension is installed on the VM if the VM is running.
  - For Windows VMs, the [VMSnapshot extension](#) is installed.
  - For Linux VMs, the [VMSnapshotLinux extension](#) is installed.
3. For Windows VMs that are running, Backup coordinates with Windows Volume Shadow Copy Service (VSS) to take an app-consistent snapshot of the VM.
  - By default, Backup takes full VSS backups.
  - If Backup can't take an app-consistent snapshot, then it takes a file-consistent snapshot of the underlying storage (because no application writes occur while the VM is stopped).
4. For Linux VMs, Backup takes a file-consistent backup. For app-consistent snapshots, you need to manually customize pre/post scripts.
5. After Backup takes the snapshot, it transfers the data to the vault.
  - The backup is optimized by backing up each VM disk in parallel.

- For each disk that's being backed up, Azure Backup reads the blocks on the disk and identifies and transfers only the data blocks that changed (the delta) since the previous backup.
- Snapshot data might not be immediately copied to the vault. It might take some hours at peak times. Total backup time for a VM will be less than 24 hours for daily backup policies.

6. Changes made to a Windows VM after Azure Backup is enabled on it are:

- Microsoft Visual C++ 2013 Redistributable(x64) - 12.0.40660 is installed in the VM
- Startup type of Volume Shadow Copy service (VSS) changed to automatic from manual
- IaaSVmProvider Windows service is added

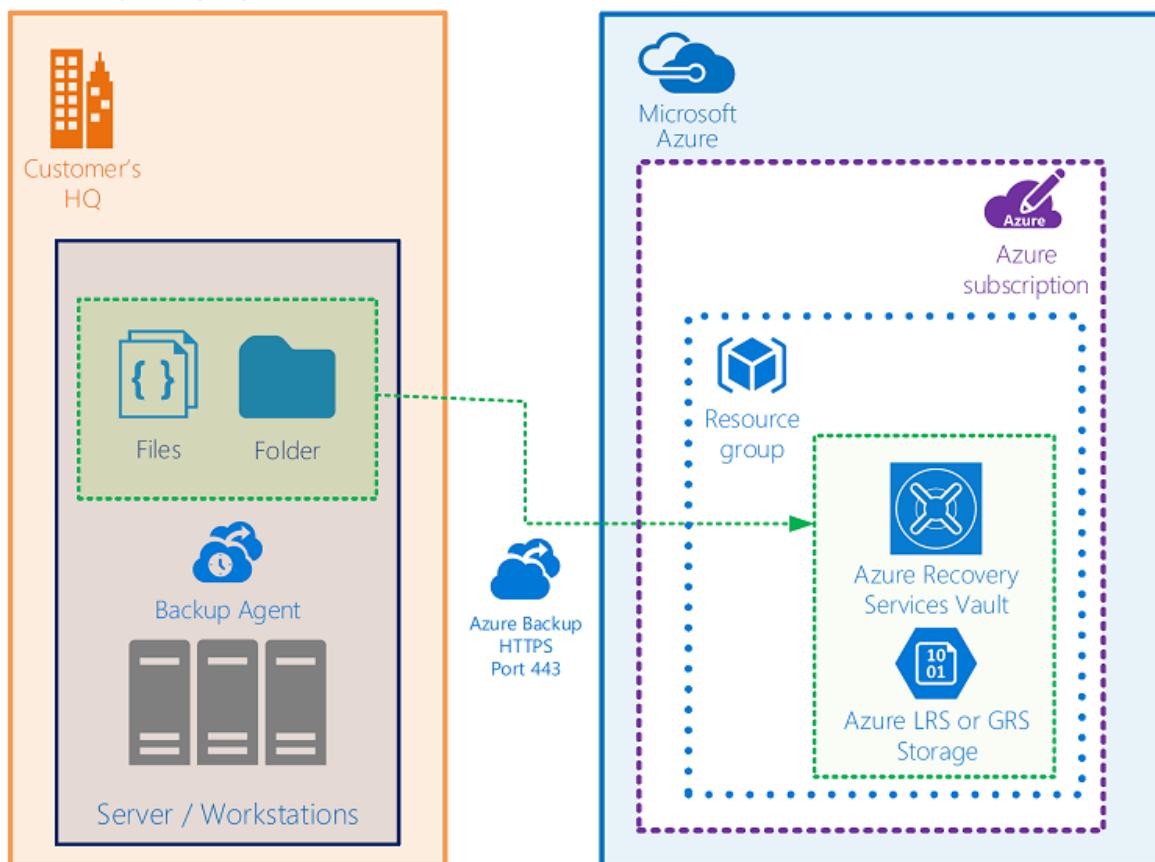


## Architecture: Direct backup of on-premises Windows Server machines or Azure VM files or folders

1. To set up the scenario, you download and install the MARS agent on the machine. You then select what to back up, when backups will run, and how long they'll be kept in Azure.
2. The initial backup runs according to your backup settings.
3. The MARS agent uses VSS to take a point-in-time snapshot of the volumes selected for backup.
  - The MARS agent uses only the Windows system write operation to capture the snapshot.

- Because the agent doesn't use any application VSS writers, it doesn't capture app-consistent snapshots.
4. After taking the snapshot with VSS, the MARS agent creates a virtual hard disk (VHD) in the cache folder you specified when you configured the backup. The agent also stores checksums for each data block. These are later used to detect changed blocks for subsequent incremental backups.
  5. Incremental backups run according to the schedule you specify, unless you run an on-demand backup.
  6. In incremental backups, changed files are identified and a new VHD is created. The VHD is compressed and encrypted, and then it's sent to the vault.
  7. After the incremental backup finishes, the new VHD is merged with the VHD created after the initial replication. This merged VHD provides the latest state to be used for comparison for ongoing backup.

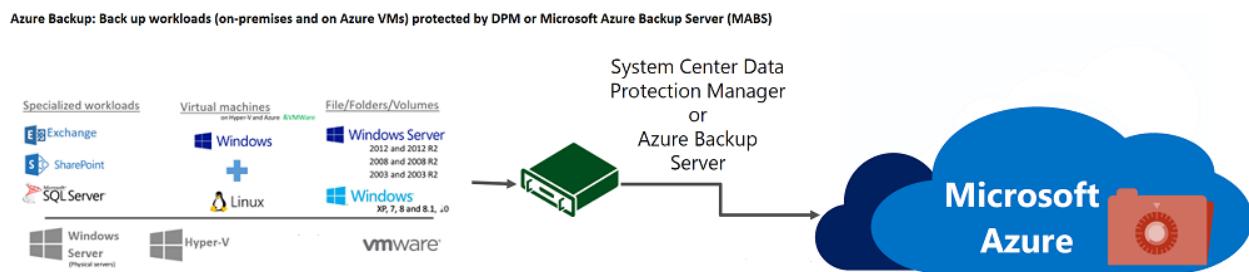
Azure Backup: Back up on-premises Windows files/folders to Azure



## Architecture: Back up to DPM/MABS

1. You install the DPM or MABS protection agent on machines you want to protect. You then add the machines to a DPM protection group.
  - To protect on-premises machines, the DPM or MABS server must be located on-premises.

- To protect Azure VMs, the MABS server must be located in Azure, running as an Azure VM.
  - With DPM/MABS, you can protect backup volumes, shares, files, and folders. You can also protect a machine's system state (bare metal), and you can protect specific apps with app-aware backup settings.
2. When you set up protection for a machine or app in DPM/MABS, you select to back up to the MABS/DPM local disk for short-term storage and to Azure for online protection. You also specify when the backup to local DPM/MABS storage should run and when the online backup to Azure should run.
  3. The disk of the protected workload is backed up to the local MABS/DPM disks, according to the schedule you specified.
  4. The DPM/MABS disks are backed up to the vault by the MARS agent that's running on the DPM/MABS server.



## Azure VM storage

Azure VMs use disks to store their operating system, apps, and data. Each Azure VM has at least two disks: a disk for the operating system and a temporary disk. Azure VMs can also have data disks for app data. Disks are stored as VHDs.

- VHDs are stored as page blobs in standard or premium storage accounts in Azure:
  - **Standard storage:** Reliable, low-cost disk support for VMs running workloads that aren't sensitive to latency. Standard storage can use standard solid-state drive (SSD) disks or standard hard disk drive (HDD) disks.
  - **Premium storage:** High-performance disk support. Uses premium SSD disks.
- There are different performance tiers for disks:
  - **Standard HDD disk:** Backed by HDDs, and used for cost-effective storage.
  - **Standard SSD disk:** Combines elements of premium SSD disks and standard HDD disks. Offers more consistent performance and reliability than HDD, but still cost-effective.
  - **Premium SSD disk:** Backed by SSDs, and provides high-performance and low-latency for VMs that are running I/O-intensive workloads.
- Disks can be managed or unmanaged:

- **Unmanaged disks:** Traditional type of disks used by VMs. For these disks, you create your own storage account and specify it when you create the disk. You then need to figure out how to maximize storage resources for your VMs.
- **Managed disks:** Azure creates and manages the storage accounts for you. You specify the disk size and performance tier, and Azure creates managed disks for you. As you add disks and scale VMs, Azure handles the storage accounts.

For more information about disk storage and the available disk types for VMs, see these articles:

- [Azure managed disks for Linux VMs](#)
- [Available disk types for VMs](#)

## Back up and restore Azure VMs with premium storage

You can back up Azure VMs by using premium storage with Azure Backup:

- During the process of backing up VMs with premium storage, the Backup service creates a temporary staging location, named *AzureBackup-*, in the storage account. The size of the staging location equals the size of the recovery point snapshot.
- Make sure that the premium storage account has adequate free space to accommodate the temporary staging location. For more information, see [Scalability targets for premium page blob storage accounts](#). Don't modify the staging location.
- After the backup job finishes, the staging location is deleted.
- The price of storage used for the staging location is consistent with [premium storage pricing](#).

When you restore Azure VMs by using premium storage, you can restore them to premium or standard storage. Typically, you would restore them to premium storage. But if you need only a subset of files from the VM, it might be cost effective to restore them to standard storage.

## Back up and restore managed disks

You can back up Azure VMs with managed disks:

- You back up VMs with managed disks in the same way that you do any other Azure VM. You can back up the VM directly from the virtual machine settings, or you can enable backup for VMs in the Recovery Services vault.
- You can back up VMs on managed disks through RestorePoint collections built on top of managed disks.

- Azure Backup also supports backing up VMs with managed disks that were encrypted by using Azure Disk Encryption.

When you restore VMs with managed disks, you can restore to a complete VM with managed disks or to a storage account:

- During the restore process, Azure handles the managed disks. If you're using the storage account option, you manage the storage account that's created during the restore process.
- If you restore a managed VM that's encrypted, make sure the VM's keys and secrets exist in the key vault before you start the restore process.

## Next steps

- Review the support matrix to [learn about supported features and limitations for backup scenarios](#).
- Set up backup for one of these scenarios:
  - [Back up Azure VMs](#).
  - [Back up Windows machines directly](#), without a backup server.
  - [Set up MABS](#) for backup to Azure, and then back up workloads to MABS.
  - [Set up DPM](#) for backup to Azure, and then back up workloads to DPM.

# Support matrix for Azure Backup

Article • 08/14/2023

You can use [Azure Backup](#) to back up data to the Microsoft Azure cloud platform. This article summarizes the general support settings and limitations for Azure Backup scenarios and deployments.

Other support matrices are available:

- Support matrix for [Azure virtual machine \(VM\) backup](#)
- Support matrix for backup by using [System Center Data Protection Manager \(DPM\)/Microsoft Azure Backup Server \(MABS\)](#)
- Support matrix for backup by using the [Microsoft Azure Recovery Services \(MARS\) agent](#)

## Note

This service supports [Azure Lighthouse](#), which lets service providers sign in to their own tenant to manage subscriptions and resource groups that customers have delegated.

## Vault support

Azure Backup uses Recovery Services vaults to orchestrate and manage backups for the following workload types - Azure VMs, SQL in Azure VMs, SAP HANA in Azure VMs, Azure File shares and on-premises workloads using Azure Backup Agent, Azure Backup Server and System Center DPM. It also uses Recovery Services vaults to store backed-up data for these workloads.

The following table describes the features of Recovery Services vaults:

Feature	Details
<b>Vaults in subscription</b>	Up to 500 Recovery Services vaults in a single subscription.
<b>Machines in a vault</b>	<p>Up to 2000 datasources across all workloads (like Azure VMs, SQL Server VM, MABS Servers, and so on) can be protected in a single vault.</p> <p>Up to 1,000 Azure VMs in a single vault.</p> <p>Up to 50 MABS servers can be registered in a single vault.</p>

Feature	Details
<b>Data sources</b>	Maximum size of an individual <a href="#">data source</a> is 54,400 GB. This limit doesn't apply to Azure VM backups. No limits apply to the total amount of data you can back up to the vault.
<b>Backups to vault</b>	<p>Azure VMs: Once a day.</p> <p>Machines protected by DPM/MABS: Twice a day.</p> <p>Machines backed up directly by using the MARS agent: Three times a day.</p>
<b>Backups between vaults</b>	<p>Backup is within a region.</p> <p>You need a vault in every Azure region that contains VMs you want to back up. You can't back up to a different region.</p>
<b>Move vaults</b>	You can <a href="#">move vaults</a> across subscriptions or between resource groups in the same subscription. However, moving vaults across regions isn't supported.
<b>Move data between vaults</b>	Moving backed-up data between vaults isn't supported.
<b>Modify vault storage type</b>	You can modify the storage replication type (either geo-redundant storage or locally redundant storage) for a vault before backups are stored. After backups begin in the vault, the replication type can't be modified.
<b>Private Endpoints</b>	See <a href="#">this section</a> for requirements to create private endpoints for a recovery service vault.

## On-premises backup support

Here's what's supported if you want to back up on-premises machines:

Machine	What's backed up	Location	Features
Direct backup of Windows machine with MARS agent	<ul style="list-style-type: none"> <li>- Files, folders</li> <li>- System state</li> </ul>	Back up to Recovery Services vault.	<ul style="list-style-type: none"> <li>- Back up three times a day</li> <li>- Back up once a day.</li> <li>No app-aware backup</li> <li>Restore file, folder, volume</li> </ul>

Machine	What's backed up	Location	Features
Direct backup of Linux machine with MARS agent	Backup not supported		
Back up to DPM	Files, folders, volumes, system state, app data	Back up to local DPM storage. DPM then backs up to vault.	App-aware snapshots Full granularity for backup and recovery Linux supported for VMs (Hyper-V/VMware) Oracle not supported
Back up to MABS	Files, folders, volumes, system state, app data	Back up to MABS local storage. MABS then backs up to the vault.	App-aware snapshots Full granularity for backup and recovery Linux supported for VMs (Hyper-V/VMware) Oracle not supported

## Azure VM backup support

### Azure VM limits

Limit	Details
Azure VM data disks	See the <a href="#">support matrix for Azure VM backup</a> .
Azure VM data disk size	Individual disk size can be up to 32 TB and a maximum of 256 TB combined for all disks in a VM.

## Azure VM backup options

Here's what's supported if you want to back up Azure VMs:

Machine	What's backed up	Location	Features
Azure VM backup by using VM extension	Entire VM	Back up to vault.  Back up once a day.	Extension installed when you enable backup for a VM.  App-aware backup for Windows VMs; file-consistent backup for Linux VMs. You can configure app-consistency for Linux machines by using custom scripts.  Restore VM or disk.
			<a href="#">Backup and restore of Active Directory domain controllers</a> is supported.  Can't back up an Azure VM to an on-premises location.
Azure VM backup by using MARS agent	- Files, folders  - System state	Back up to vault.	- Back up three times a day.  - Back up once a day.  If you want to back up specific files or folders rather than the entire VM, the MARS agent can run alongside the VM extension.
Azure VM with DPM	Files, folders, volumes, system state, app data	Back up to local storage of Azure VM that's running DPM.  DPM then backs up to vault.	App-aware snapshots.  Full granularity for backup and recovery.  Linux supported for VMs (Hyper-V/VMware).  Oracle not supported.
Azure VM with MABS	Files, folders, volumes, system state, app data	Back up to local storage of Azure VM that's running MABS.  MABS then backs up to the vault.	App-aware snapshots.  Full granularity for backup and recovery.  Linux supported for VMs (Hyper-

Machine	What's backed up	Location	Features
			V/VMware).  Oracle not supported.

## Linux backup support

Here's what's supported if you want to back up Linux machines:

Backup type	Linux (Azure endorsed)
<b>Direct backup of on-premises machine that's running Linux</b>	Not supported. The MARS agent can be installed only on Windows machines.
<b>Using agent extension to back up Azure VM that's running Linux</b>	App-consistent backup by using <a href="#">custom scripts</a> .  File-level recovery.  Restore by creating a VM from a recovery point or disk.
<b>Using DPM to back up on-premises machines running Linux</b>	File-consistent backup of Linux Guest VMs on Hyper-V and VMware.  VM restoration of Hyper-V and VMware Linux Guest VMs.
<b>Using MABS to back up on-premises machines running Linux</b>	File-consistent backup of Linux Guest VMs on Hyper-V and VMware.  VM restoration of Hyper-V and VMware Linux guest VMs.
<b>Using MABS or DPM to back up Linux Azure VMs</b>	Not supported.

## Daylight saving time support

Azure Backup doesn't support automatic clock adjustment for daylight saving time for Azure VM backups. It doesn't shift the hour of the backup forward or backwards. To ensure the backup runs at the desired time, modify the backup policies manually as required.

# Disk deduplication support

Disk deduplication support is as follows:

- Disk deduplication is supported on-premises when you use DPM or MABS to back up Hyper-V VMs that are running Windows. Windows Server performs data deduplication (at the host level) on virtual hard disks (VHDs) that are attached to the VM as backup storage.
- Deduplication isn't supported in Azure for any Backup component. When DPM and MABS are deployed in Azure, the storage disks attached to the VM can't be deduplicated.

## ⓘ Note

Azure VM backup does not support Azure VM with deduplication. This means Azure Backup does not deduplicate backup data, except in MABS/MARS.

# Security and encryption support

Azure Backup supports encryption for in-transit and at-rest data.

## Network traffic to Azure

- The backup traffic from servers to the Recovery Services vault is encrypted by using Advanced Encryption Standard 256.
- Backup data is sent over a secure HTTPS link.

## Data security

- Backup data is stored in the Recovery Services vault in encrypted form.
- When data is backed-up from on-premises servers with the MARS agent, data is encrypted with a passphrase before upload to the Azure Backup service and decrypted only after it's downloaded from Azure Backup.
- When you're backing up Azure VMs, you need to set up encryption *within* the virtual machine.
- Azure Backup supports Azure Disk Encryption, which uses BitLocker on Windows virtual machines and **dm-crypt** on Linux virtual machines.
- On the back end, Azure Backup uses [Azure Storage Service Encryption](#), which protects data at rest.

Machine	In transit	At rest
On-premises Windows machines without DPM/MABS		
Azure VMs		
On-premises Windows machines or Azure VMs with DPM		
On-premises Windows machines or Azure VMs with MABS		

## Compression support

Backup supports the compression of backup traffic, as summarized in the following table.

- For Azure VMs, the VM extension reads the data directly from the Azure storage account over the storage network, so it isn't necessary to compress this traffic.
- If you're using DPM or MABS, you can save bandwidth by compressing the data before it's backed up.

Machine	Compress to MABS/DPM (TCP)	Compress to vault (HTTPS)
Direct backup of on-premises Windows machines	NA	
Backup of Azure VMs by using VM extension	NA	NA
Backup on on-premises/Azure machines by using MABS/DPM		

## Retention limits

Setting	Limits
Maximum recovery points per protected instance (machine or workload)	9,999
Maximum expiry time for a recovery point	No limit
Maximum backup frequency to DPM/MABS	Every 15 minutes for SQL Server Once an hour for other workloads

Setting	Limits
Maximum backup frequency to vault	On-premises Windows machines or Azure VMs running MARS: Three per day
	DPM/MABS: Two per day
	Azure VM backup: One per day
Recovery point retention	Daily, weekly, monthly, yearly
Maximum retention period	Depends on backup frequency
Recovery points on DPM/MABS disk	64 for file servers; 448 for app servers
	Unlimited tape recovery points for on-premises DPM

## Cross Region Restore

Azure Backup has added the Cross Region Restore feature to strengthen data availability and resiliency capability, giving you full control to restore data to a secondary region. To configure this feature, see [Set Cross Region Restore](#). This feature is supported for the following management types:

Backup Management type	Supported	Supported Regions
Azure VM	Supported for Azure VMs (including encrypted Azure VMs) with both managed and unmanaged disks. Not supported for classic VMs.	Available in all Azure public regions and sovereign regions, except for UG IOWA.
SQL /SAP HANA	Available	Available in all Azure public regions and sovereign regions, except for France Central and UG IOWA.
MARS Agent (Preview)	Available in preview.  Not supported for vaults with Private Endpoint enabled.	Available in all Azure public regions.
DPM/MABS	No	N/A
AFS (Azure file shares)	No	N/A

# Resource health

The resource health check functions in following conditions:

Resource health check	Details
Supported Resources	Recovery Services vault, Backup vault
Supported Regions	<ul style="list-style-type: none"><li>- Recovery Services vault: Supported in all Azure public regions, US Sovereign cloud, and China Sovereign cloud.</li><li>- Backup vault: Supported in all Azure public regions, except Sovereign clouds.</li></ul>
For unsupported regions	The resource health status is shown as "Unknown".

## Zone-redundant storage support

Azure Backup now supports zone-redundant storage (ZRS).

## Supported regions

- Azure Backup currently supports ZRS for all workloads, except Azure Disk, in the following regions: UK South, South East Asia, Australia East, North Europe, Central US, East US 2, Brazil South, South Central US, Korea Central, Norway East, France Central, West Europe, East Asia, Sweden Central, Canada Central, India Central, South Africa North, West US 2, Japan East, East US, US Gov Virginia, Switzerland North, Qatar, UAE North, and West US 3.
- ZRS support for Azure Disk is generally available in the following regions: UK South, Southeast Asia, Australia East, North Europe, Central US, South Central US, West Europe, West US 2, Japan East, East US, US Gov Virginia, Qatar, and West US 3.

## Supported scenarios

Here's the list of scenarios supported even if zone gets unavailable in the supported regions:

- Create/List/Update Policy

- List backup jobs
- List of protected items
- Update vault config
- Create vault
- Get vault credential file

## Supported operations

The following table lists the workload specific operations supported even if zone gets unavailable in the supported regions:

Protected workload	Supported Operations
IAAS VM	<ul style="list-style-type: none"> <li>- Backups are successful, if the protected VM is in an active zone.</li> <li>- Original location recovery (OLR) is successful, if the protected VM is in an active zone.</li> <li>- Alternate location restores (ALR) to an active zone is successful.</li> </ul>
SQL/ SAP HANA database in Azure VM	<ul style="list-style-type: none"> <li>- Backups are successful, if the protected workload is in an active zone.</li> <li>- Original location recovery (OLR) is successful, if the protected workload is in an active zone.</li> <li>- Alternate location restores (ALR) to an active zone is successful.</li> </ul>
Azure Files	Backups, OLR, and ALR are successful, if the protected file share is in a ZRS account.
Blob	Recovery is successful, if the protected storage account is in ZRS.
Disk	<ul style="list-style-type: none"> <li>- Backups are successful, if the protected disk is in an active zone.</li> <li>- Restore to an active zone is successful.</li> </ul>
MARS	Backups and restores are successful.

## Next steps

- [Review support matrix for Azure VM backup.](#)

# Backup cloud and on-premises workloads to cloud

Article • 04/24/2023

Azure Backup comprehensively protects your data assets in Azure through a simple, secure, and cost-effective solution that requires zero-infrastructure. It's Azure's built-in data protection solution for a wide range of workloads. It helps protect your mission critical workloads running in the cloud, and ensures your backups are always available and managed at scale across your entire backup estate.

## Intended audience

The primary target audience for this article is the IT and application administrators, and implementers of large and mid-sized organizations, who want to learn about the capabilities of Azure's built-in data protection technology, Azure Backup, and to implement solutions to protect your deployments efficiently. The article assumes you're familiar with core Azure technologies, data protection concepts and have experience working with a backup solution. The guidance covered in this article can make it easier to design your backup solution on Azure using established patterns and avoid known pitfalls.

## How this article is organized

While it's easy to start protecting infrastructure and applications on Azure, when you ensure that the underlying Azure resources are set up correctly and being used optimally you can accelerate your time to value. This article covers a brief overview of design considerations and guidance for optimally configuring your Azure Backup deployment. It examines the core components (for example, Recovery Services vault, Backup Policy) and concepts (for example, governance) and how to think of them and their capabilities with links to detailed product documentation.

## Get started

### Subscription design strategy

Apart from having a clear roadmap to navigate through the Cloud Adoption Journey, you must plan your cloud deployment's subscription design and account structure to match your organization's ownership, billing, and management capabilities. As the vault

is scoped to a subscription, your Subscription design will highly influence your Vault design. [Learn more](#) about different Subscription Design Strategies and guidance on when to use them.

## Document your Backup requirements

To get started with Azure Backup, plan your backup needs. Following are some of the questions you should ask yourself while formulating a perfect backup strategy.

### **What workload type do you wish to protect?**

To design your vaults, ensure if you require a centralized/ decentralized mode of operation.

### **What's the required backup granularity ?**

Determine if it should be application consistent, crash consistent, or log backup.

### **Do you've any compliance requirements?**

Ensure if you need to enforce security standards and separate access boundaries.

### **What's the required RPO, RTO?**

Determine the backup frequency and the speed of restore.

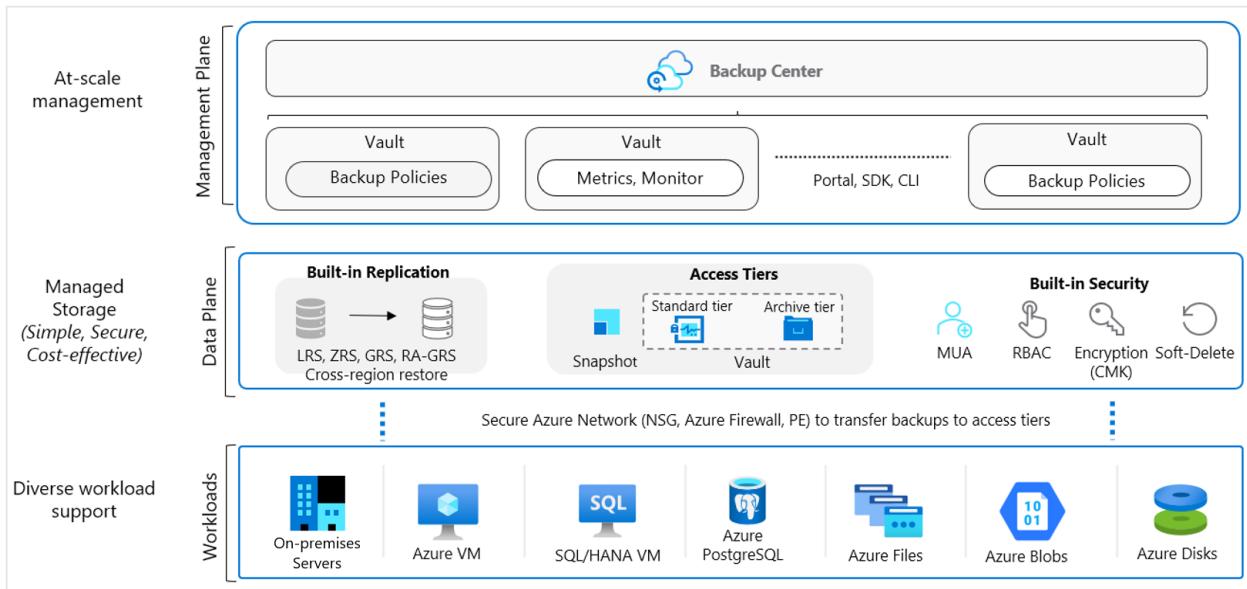
### **Do you've any Data Residency constraints?**

Determine the storage redundancy for the required Data Durability.

### **How long do you want to retain the backup data?**

Decide on the duration the backed-up data be retained in the storage.

## Architecture



## Workloads

Azure Backup enables data protection for various workloads (on-premises and cloud). It's a secure and reliable built-in data protection mechanism in Azure. It can seamlessly scale its protection across multiple workloads without any management overhead for you. There are multiple automation channels as well to enable this (via PowerShell, CLI, Azure Resource Manager templates, and REST APIs.)

- **Scalable, durable, and secure storage:** Azure Backup uses reliable Blob storage with in-built security and high availability features. You can choose LRS, GRS, or RA-GRS storages for your backup data.
- **Native workload integration:** Azure Backup provides native integration with Azure Workloads (VMs, SAP HANA, SQL in Azure VMs and even Azure Files) without requiring you to manage automation or infrastructure to deploy agents, write new scripts or provision storage.

[Learn more](#) about supported workloads.

## Data plane

- **Automated storage management:** Azure Backup automates provisioning and managing storage accounts for the backup data to ensure it scales as the backup data grows.
- **Malicious delete protection:** Protect against any accidental and malicious attempts for deleting your backups via soft delete of backups. The deleted backup data is stored for 14 days free of charge and allows it to be recovered from this state.

- **Secure encrypted backups:** Azure Backup ensures your backup data is stored in a secure manner, leveraging built-in security capabilities of the Azure platform, such as Azure role-based access control (Azure RBAC) and Encryption.
- **Backup data lifecycle management:** Azure Backup automatically cleans up older backup data to comply with the retention policies. You can also tier your data from operational storage to vault storage.
- **Protected critical operations:** Multi-user authorization (MUA) for Azure Backup allows you to add an additional layer of protection to critical operations on your Recovery Services vaults.

## Management plane

- **Access control:** Vaults (Recovery Services and Backup vaults) provide the management capabilities and are accessible via the Azure portal, Backup Center, Vault dashboards, SDK, CLI, and even REST APIs. It's also an Azure role-based access control (Azure RBAC) boundary, providing you with the option to restrict access to backups only to authorized Backup Admins.
- **Policy management:** Azure Backup Policies within each vault define when the backups should be triggered and the duration they need to be retained. You can also manage these policies and apply them across multiple items.
- **Monitoring and Reporting:** Azure Backup integrates with Log Analytics and provides the ability to see reports via Workbooks as well.
- **Snapshot management:** Azure Backup takes snapshots for some Azure native workloads (VMs and Azure Files), manages these snapshots and allows fast restores from them. This option drastically reduces the time to recover your data to the original storage.

## Vault considerations

Azure Backup uses vaults (Recovery Services and Backup vaults) to orchestrate, manage backups, and store backed-up data. Effective vault design helps organizations establish a structure to organize and manage the backup assets in Azure to support your business priorities. Consider the following guidelines when creating a vault.

## Single or multiple vaults

To use a single vault or multiple vaults to organize and manage your backup, see the following guidelines:

- **Protect resources across multiple regions globally:** If your organization has global operations across North America, Europe, and Asia, and your resources are deployed in East-US, UK West, and East Asia. One of the requirements of Azure Backup is that the vaults are required to be present in the same region as the resource to be backed-up. Therefore, you should create three separate vaults for each region to protect your resources.
- **Protect resources across various Business Units and Departments:** Consider that your business operations are divided into three separate Business Units (BU), and each business unit has its own set of departments (five departments - Finance, Sales, HR, R & D, and Marketing). Your business needs may require each department to manage and access their own backups and restores; also, enable them to track their individual usage and cost expense. For such scenarios, we recommend you to create one vault for each department in a BU. This way, you'll have 15 Vaults across your organization.
- **Protect different workloads:** If you plan to protect different types of workloads (such as 150 VMs, 40 SQL databases, and 70 PostgreSQL databases), then we recommend you create separate vaults for each type of workload (for this example, you need to create three vaults for each workload - VMs, SQL databases, and PostgreSQL databases). This helps you to separate access boundaries for the users by allowing you to grant access (using Azure role-based access control – Azure RBAC) to the relevant stakeholders.
- **Protect resources running in multiple environments:** If your operations require you to work on multiple environments, such as production, non-production, and developer, then we recommend you create separate vaults for each.
- **Protect large number (1000+) of Azure VMs:** Consider that you have 1500 VMs to back up. Azure Backup allows only 1000 Azure VMs to be backed-up in one vault. For this scenario, you can create two different vaults and distribute the resources as 1000 and 500 VMs to respective vaults or in any combination considering the upper limit.
- **Protect large number (2000+) of diverse workloads:** While managing your backups at scale, you'll protect the Azure VMs, along with other workloads, such as SQL and SAP HANA database running on those Azure VMs. For example, you've 1300 Azure VMs and 2500 SQL databases to protect. The vault limits allow you to back up 2000 workloads (with a restriction of 1000 VMs) in each vault. Therefore, mathematically you can back up 2000 workloads in one vault (1000 VMs + 1000

SQL databases) and rest 1800 workloads in a separate vault (300 VMs + 1500 SQL databases).

However, this type of segregation isn't recommended as you won't be able to define access boundaries and the workloads won't be isolated from each other. So, to distribute the workloads correctly, create four vaults. Two vaults to back up the VMs (1000 VMs + 300 VMs) and the other two vaults to back up the SQL databases (2000 databases + 500 databases).

- You can manage them with:
  - Backup center allows you to have a single pane to manage all Backup tasks. [Learn more here](#).
  - If you need consistent policy across vaults, then you can use Azure Policy to propagate backup policy across multiple vaults. You can write a custom [Azure Policy definition](#) that uses the '`deployifnotexists`' effect to propagate a backup policy across multiple vaults. You can also [assign](#) this Azure Policy definition to a particular scope (subscription or RG), so that it deploys a 'backup policy' resource to all Recovery Services vaults in the scope of the Azure Policy assignment. The settings of the backup policy (such as backup frequency, retention, and so on) should be specified by the user as parameters in the Azure Policy assignment.
- As your organizational footprint grows, you might want to move workloads across subscriptions for the following reasons: align by backup policy, consolidate vaults, trade-off on lower redundancy to save on cost (move from GRS to LRS). Azure Backup supports moving a Recovery Services vault across Azure subscriptions, or to another resource group within the same subscription. [Learn more here](#).

## Review default settings

Review the default settings for Storage Replication type and Security settings to meet your requirements before configuring backups in the vault.

- *Storage Replication type* by default is set to Geo-redundant (GRS). Once you configure the backup, the option to modify is disabled. Follow [these](#) steps to review and modify the settings.
  - Non-critical workloads like non-prod and dev are suitable for LRS storage replication.
  - Zone redundant storage (ZRS) is a good storage option for a high Data Durability along with Data Residency.

- Geo-Redundant Storage (GRS) is recommended for mission-critical workloads, such as the ones running in production environment, to prevent permanent data loss, and protect it in case of complete regional outage or a disaster in which the primary region isn't recoverable.
- *Soft delete* by default is Enabled on newly created vaults to protect backup data from accidental or malicious deletes. Follow [these](#) steps to review and modify the settings.
- *Cross Region Restore* allows you to restore Azure VMs in a secondary region, which is an Azure paired region. This option allows you to conduct drills to meet audit or compliance requirements, and to restore the VM or its disk if there's a disaster in the primary region. CRR is an opt-in feature for any GRS vault. [Learn more here](#).
- Before finalizing your vault design, review the [vault support matrixes](#) to understand the factors that might influence or limit your design choices.

## Backup Policy considerations

Azure Backup Policy has two components: *Schedule* (when to take backup) and *Retention* (how long to retain backup). You can define the policy based on the type of data that's being backed up, RTO/RPO requirements, operational or regulatory compliance needs and workload type (for example, VM, database, files). [Learn more](#)

Consider the following guidelines when creating Backup Policy:

### Schedule considerations

While scheduling your backup policy, consider the following points:

- For mission-critical resources, try scheduling the most frequently available automated backups per day to have a smaller RPO. [Learn more](#)  
If you need to take multiple backups per day for Azure VM via the extension, see the workarounds in the [next section](#).
- For a resource that requires the same schedule start time, frequency, and retention settings, you need to group them under a single backup policy.
- We recommend you to keep the backup scheduled start time during non-peak production application time. For example, it's better to schedule the daily automated backup during night, around 2-3 AM, rather than scheduling it in the day time when the usage of the resources high.

- To distribute the backup traffic, we recommend you back up different VMs at different times of the day. For example, to back up 500 VMs with the same retention settings, we recommend you to create 5 different policies associating them with 100 VMs each and scheduling them few hours apart.

## Retention considerations

- Short-term retention can be "daily". Retention for "Weekly", "monthly" or "yearly" backup points is referred to as Long-term retention.
- Long-term retention:
  - Planned (compliance requirements) - if you know in advance that data is required years from the current time, then use Long-term retention. Azure Backup supports back up of Long-Term Retention points in the archive tier, along with Snapshots and the Standard tier. [Learn more](#) about supported workloads for Archive tier and retention configuration.
  - Unplanned (on-demand requirement) - if you don't know in advance, then use you can use on-demand with specific custom retention settings (these custom retention settings aren't impacted by policy settings).
- On-demand backup with custom retention - if you need to take a backup not scheduled via backup policy, then you can use an on-demand backup. This can be useful for taking backups that don't fit your scheduled backup or for taking granular backup (for example, multiple IaaS VM backups per day since scheduled backup permits only one backup per day). It's important to note that the retention policy defined in scheduled policy doesn't apply to on-demand backups.

## Optimize Backup Policy

- As your business requirements change, you might need to extend or reduce retention duration. When you do so, you can expect the following:
  - If retention is extended, existing recovery points are marked and kept in accordance with the new policy.
  - If retention is reduced, recovery points are marked for pruning in the next clean-up job, and subsequently deleted.
  - The latest retention rules apply for all retention points (excluding on-demand retention points). So if the retention period is extended (for example to 100 days), then when the backup is taken, followed by retention reduction (for example from 100 days to seven days), all backup data will be retained according to the last specified retention period (that is, 7 days).

- Azure Backup provides you with the flexibility to *stop protecting and manage your backups*:
  - *Stop protection and retain backup data.* If you're retiring or decommissioning your data source (VM, application), but need to retain data for audit or compliance purposes, then you can use this option to stop all future backup jobs from protecting your data source and retain the recovery points that have been backed up. You can then restore or resume VM protection.
  - *Stop protection and delete backup data.* This option will stop all future backup jobs from protecting your VM and delete all the recovery points. You won't be able to restore the VM nor use Resume backup option.
  - If you resume protection (of a data source that has been stopped with retain data), then the retention rules will apply. Any expired recovery points will be removed (at the scheduled time).
- Before completing your policy design, it's important to be aware of the following factors that might influence your design choices.
  - A backup policy is scoped to a vault.
  - There's a limit on the number of items per policy (for example, 100 VMs). To scale, you can create duplicate policies with the same or different schedules.
  - You can't selectively delete specific recovery points.
  - You can't completely disable the scheduled backup and keep the data source in a protected state. The least frequent backup you can configure with the policy is to have one weekly scheduled backup. An alternative would be to stop protection with retain data and enable protection each time you want to take a backup, take an on-demand backup, and then turn off protection but retain the backup data. [Learn more here](#).

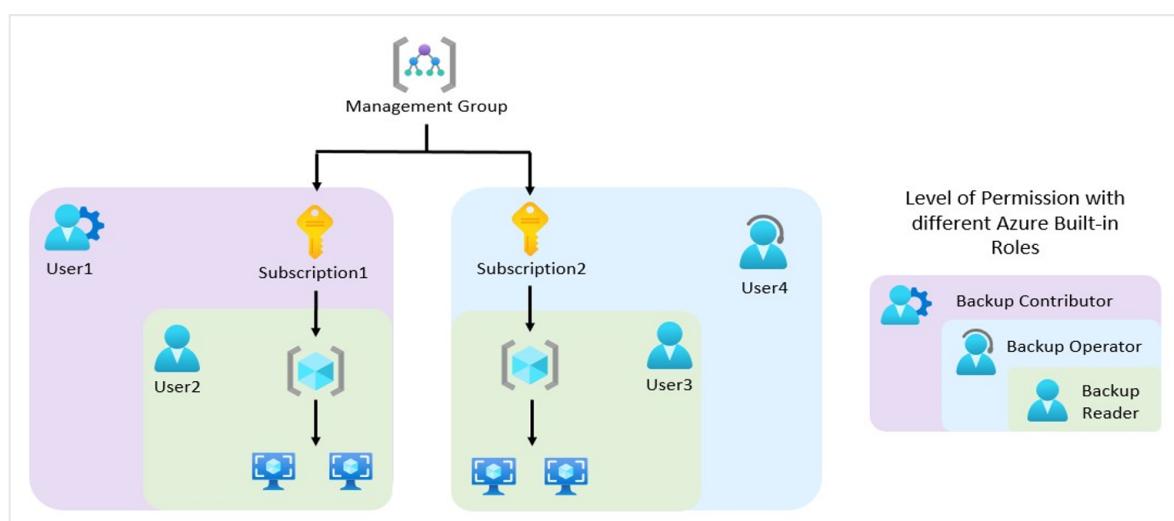
## Security considerations

To help you protect your backup data and meet the security needs of your business, Azure Backup provides confidentiality, integrity, and availability assurances against deliberate attacks and abuse of your valuable data and systems. Consider the following security guidelines for your Azure Backup solution:

## Authentication and authorization using Azure role-based access control (Azure RBAC)

- Azure role-based access control (Azure RBAC) enables fine-grained access management, segregation of duties within your team and granting only the amount of access to users necessary to perform their jobs. [Learn more here](#).
- If you've multiple workloads to back up (such as Azure VMs, SQL databases, and PostgreSQL databases) and you've multiple stakeholders to manage those backups, it is important to segregate their responsibilities so that user has access to only those resources they're responsible for. Azure role-based access control (Azure RBAC) enables granular access management, segregation of duties within your team, and granting only the types of access to users necessary to perform their jobs. [Learn more](#)
- You can also segregate the duties by providing minimum required access to perform a particular task. For example, a person responsible for monitoring the workloads shouldn't have access to modify the backup policy or delete the backup items. Azure Backup provides three built-in roles to control backup management operations: Backup contributors, operators, and readers. [Learn more here](#). For information about the minimum Azure role required for each backup operation for Azure VMs, SQL/SAP HANA databases, and Azure File Share, see [this guide](#).
- Azure role-based access control (Azure RBAC) also provides the flexibility to build [Custom Roles](#) based on your individual requirements. If you're unsure about the types of roles recommended for specific operation, you can utilize the built-in roles provided by Azure role-based access control (Azure RBAC) and get started.

The following diagram explains about how different Azure built-in roles work:



- In the above diagram, *User2* and *User3* are Backup Readers. Therefore, they have the permission to only monitor the backups and view the backup services.
- In terms of the scope of the access,

- *User2* can access only the Resources of *Subscription1*, and *User3* can access only the Resources of *Subscription2*.
- *User4* is a Backup Operator. It has the permission to enable backup, trigger on-demand backup, trigger restores, along with the capabilities of a Backup Reader. However, in this scenario, its scope is limited only to *Subscription2*.
- *User1* is a Backup Contributor. It has the permission to create vaults, create/modify/delete backup policies, and stop backups, along with the capabilities of a Backup Operator. However, in this scenario, its scope is limited only to *Subscription1*.
- Storage accounts used by Recovery Services vaults are isolated and can't be accessed by users for any malicious purposes. The access is only allowed through Azure Backup management operations, such as restore.

## Encryption of data in transit and at rest

Encryption protects your data and helps you to meet your organizational security and compliance commitments.

- Within Azure, data in transit between Azure storage and the vault is protected by HTTPS. This data remains within the Azure network.
- Backup data is automatically encrypted using Microsoft-managed keys. Alternatively, you can use your own keys, also known as [customer managed keys](#). Also, using CMK encryption for backup doesn't incur additional costs. However, the use of Azure Key Vault, where your key is stored, incur costs, which are a reasonable expense in return for the higher data security.
- Azure Backup supports backup and restore of Azure VMs that have their OS/data disks encrypted with Azure Disk Encryption (ADE). [Learn more](#)

## Protection of backup data from unintentional deletes with soft-delete

You may encounter scenarios where you've mission-critical backup data in a vault, and it gets deleted accidentally or erroneously. Also, a malicious actor may delete your production backup items. It's often costly and time-intensive to rebuild those resources and can even cause crucial data loss. Azure Backup provides safeguard against accidental and malicious deletion with the [Soft-Delete](#) feature by allowing you to recover those resources after they are deleted.

With soft-delete, if a user deletes the backup (of a VM, SQL Server database, Azure file share, SAP HANA database), the backup data is retained for 14 additional days, allowing the recovery of that backup item with no data loss. The additional 14 days retention of backup data in the soft delete state doesn't incur any cost. [Learn more](#)

## Multi-User Authorization (MUA)

**How would you protect your data if your administrator goes rogue and compromises your system?**

Any administrator that has the privileged access to your backup data has the potential to cause irreparable damage to the system. A rogue admin can delete all your business-critical data or even turn off all the security measures that may leave your system vulnerable to cyber-attacks.

Azure Backup provides you with the [Multi-User Authorization \(MUA\)](#) feature to protect you from such rogue administrator attacks. Multi-user authorization helps protect against a rogue administrator performing destructive operations (that is, disabling soft-delete), by ensuring that every privileged/destructive operation is done only after getting approval from a security administrator.

## Ransomware Protection

- Direct access to Azure Backup data to encrypt by malicious actor is ruled out, as all operations on backup data can only be performed through Recovery-Services vault or Backup Vault, which can be secured by Azure role-based access control (Azure RBAC) and MUA.
- By enabling soft-delete on backup data (which is enabled by default) will hold deleted data for 14 days (at free of cost). Disabling soft-delete can be protected using MUA.
- Use longer retention (weeks, months, years) to ensure clean backups (not encrypted by ransomware) don't expire prematurely, and there're strategies in place for early detection and mitigation of such attacks on source data.

## Monitoring and alerts of suspicious activity

You may encounter scenarios where someone tries to breach into your system and maliciously turn off the security mechanisms, such as disabling Soft Delete or attempts to perform destructive operations, such as deleting the backup resources.

Azure Backup provides security against such incidents by sending you critical alerts over your preferred notification channel (email, ITSM, Webhook, runbook, and sp pn) by creating an [Action Rule](#) on top of the alert. [Learn more](#)

## Security features to help protect hybrid backups

Azure Backup service uses the Microsoft Azure Recovery Services (MARS) agent to back up and restore files, folders, and the volume or system state from an on-premises computer to Azure. MARS now provides security features: a passphrase to encrypt before upload and decrypt after download from Azure Backup, deleted backup data is retained for an additional 14 days from the date of deletion, and critical operation (ex. changing a passphrase) can be performed only by users who have valid Azure credentials. [Learn more here](#).

## Network considerations

Azure Backup requires movement of data from your workload to the Recovery Services vault. Azure Backup provides several capabilities to protect backup data from being exposed inadvertently (such as a man-in-the-middle attack on the network). Consider the following guidelines:

### Internet connectivity

- **Azure VM backup:** All the required communication and data transfer between storage and Azure Backup service happens within the Azure network without needing to access your virtual network. So backup of Azure VMs placed inside secured networks don't require you to allow access to any IPs or FQDNs.
- **SAP HANA databases on Azure VM, SQL Server databases on Azure VM:** Requires connectivity to the Azure Backup service, Azure Storage, and Azure Active Directory. This can be achieved by using private endpoints or by allowing access to the required public IP addresses or FQDNs. Not allowing proper connectivity to the required Azure services may lead to failure in operations like database discovery, configuring backup, performing backups, and restoring data. For complete network guidance while using NSG tags, Azure firewall, and HTTP Proxy, refer to these [SQL](#) and [SAP HANA](#) articles.
- **Hybrid:** The MARS (Microsoft Azure Recovery Services) agent requires network access for all critical operations - install, configure, backup, and restore. The MARS agent can connect to the Azure Backup service over [Azure ExpressRoute](#) by using

public peering (available for old circuits) and Microsoft peering, using [private endpoints](#) or via [proxy/firewall with appropriate access controls](#).

## Private Endpoints for secure access

While protecting your critical data with Azure Backup, you wouldn't want your resources to be accessible from the public internet. Especially, if you're a bank or a financial institution, you would have stringent compliance and security requirements to protect your High Business Impact (HBI) data. Even in the healthcare industry, there are strict compliance rules.

To fulfill all these needs, use [Azure Private Endpoint](#), which is a network interface that connects you privately and securely to a service powered by Azure Private Link. We recommend you to use private endpoints for secure backup and restore without the need to add to an allowlist of any IPs/FQDNs for Azure Backup or Azure Storage from your virtual networks.

[Learn more](#) about how to create and use private endpoints for Azure Backup inside your virtual networks.

- When you enable private endpoints for the vault, they're only used for backup and restore of SQL and SAP HANA workloads in an Azure VM, MARS agent, DPM/MABS backups. You can use the vault for the backup of other workloads as well (they won't require private endpoints though). In addition to the backup of SQL and SAP HANA workloads, backup using the MARS agent and DPM/MABS Server, private endpoints are also used to perform file recovery in the case of Azure VM backup. [Learn more here](#).
- Azure Active Directory doesn't currently support private endpoints. So, IPs and FQDNs required for Azure Active Directory will need to be allowed outbound access from the secured network when performing backup of databases in Azure VMs and backup using the MARS agent. You can also use NSG tags and Azure Firewall tags for allowing access to Azure AD, as applicable. Learn more about the [prerequisites here](#).

## Governance considerations

Governance in Azure is primarily implemented with [Azure Policy](#) and [Azure Cost Management](#). [Azure Policy](#) allows you to create, assign, and manage policy definitions to enforce rules for your resources. This feature keeps those resources in compliance with your corporate standards. [Azure Cost Management](#) allows you to track cloud usage and expenditures for your Azure resources and other cloud providers. Also, the

following tools such as [Azure Price Calculator](#) and [Azure Advisor](#) play an important role in the cost management process.

## Auto-configure newly provisioned backup infrastructure with Azure Policy at Scale

- Whenever new infrastructure is provisioned and new VMs are created, as a backup admin, you need to ensure their protection. You can easily configure backups for one or two VMs. But it becomes complex when you need to configure hundreds or even thousands of VMs at scale. To simplify the process of configuring backups, Azure Backup provides you with a set of built-in Azure Policies to govern your backup estate.
- **Auto-enable backup on VMs using Policy (Central backup team model):** If your organization has a central backup team that manages backups across application teams, you can use this policy to configure backup to an existing central Recovery Services vault in the same subscription and location as that of the VMs. You can choose to include/exclude VMs that contain a certain tag from the policy scope. [Learn more](#).
- **Auto-enable backup on VMs using Policy (where backup owned by application teams):** If you organize applications in dedicated resource groups and want to have them backed-up by the same vault, use this policy to automatically manage this action. You can choose to include/exclude VMs that contain a certain tag from the policy scope. [Learn more](#).
- **Monitoring Policy:** To generate the Backup Reports for your resources, enable the diagnostic settings when you create a new vault. Often, adding a diagnostic setting manually per vault can be a cumbersome task. So, you can utilize an Azure built-in policy that configures the diagnostics settings at scale to all vaults in each subscription or resource group, with Log Analytics as the destination.
- **Audit-only Policy:** Azure Backup also provides you with an Audit-only policy that identifies the VMs with no backup configuration.

## Azure Backup cost considerations

The Azure Backup service offers the flexibility to effectively manage your costs; also, meet your BCDR (business continuity and disaster recovery) business requirement. Consider the following guidelines:

- Use the pricing calculator to evaluate and optimize cost by adjusting various levers. [Learn more](#)
- Optimize backup policy,
  - Optimize schedule and retention settings based on workload archetypes (such as mission-critical, non-critical).
  - Optimize retention settings for Instant Restore.
  - Choose the right backup type to meet requirements, while taking supported backup types (full, incremental, log, differential) by the workload in Azure Backup.
- **Reduce the backup storage cost with Selectively backup disks:** Exclude disk (preview feature) provides an efficient and cost-effective choice to selectively backup critical data. For example, you can back up only one disk when you don't want to back up all disks attached to a VM. This is also useful when you have multiple backup solutions. For example, to back up your databases or data with a workload backup solution (SQL Server database in Azure VM backup), use Azure VM level backup for selected disks.
- **Speed up your Restores and minimize RTO using the Instant Restore feature:** Azure Backup takes snapshots of Azure VMs and stores them along with the disks to boost recovery point creation and to speed up restore operations. This is called Instant Restore. This feature allows a restore operation from these snapshots by cutting down the restore times. It reduces the time needed to transform and copy data back from the vault. Therefore, it'll incur storage costs for the snapshots taken during this period. Learn more about [Azure Backup Instant Recovery capability](#).
- **Choose correct replication type:** Azure Backup vault's Storage Replication type is set to Geo-redundant (GRS), by default. This option can't be changed after you start protecting items. Geo-redundant storage (GRS) provides a higher level of data durability than Locally redundant storage (LRS), allows an opt-in to use Cross Region Restore, and costs more. Review the trade-offs between lower costs and higher data durability and choose the best option for your scenario. [Learn more](#)
- **Use Archive Tier for Long-Term Retention (LTR) and save costs:** Consider the scenario where you've older backup data that you rarely access, but is required to be stored for a long period (for example, 99 years) for compliance reasons. Storing such huge data in a Standard Tier is costly and isn't economical. To help you optimize your storage costs, Azure Backup provides you with [Archive Tier](#), which is an access tier especially designed for Long-Term Retention (LTR) of the backup data.

- If you're protecting both the workload running inside a VM and the VM itself, ensure if this dual protection is needed.

## Monitoring and Alerting considerations

As a backup user or administrator, you should be able to monitor all backup solutions and get notified on important scenarios. This section details the monitoring and notification capabilities provided by the Azure Backup service.

### Monitor

- Azure Backup provides **built-in job monitoring** for operations such as configuring backup, back up, restore, delete backup, and so on. This is scoped to the vault, and ideal for monitoring a single vault. [Learn more here](#).
- If you need to monitor operational activities at scale, then **Backup Explorer** provides an aggregated view of your entire backup estate, enabling detailed drill-down analysis and troubleshooting. It's a built-in Azure Monitor workbook that gives a single, central location to help you monitor operational activities across the entire backup estate on Azure, spanning tenants, locations, subscriptions, resource groups, and vaults. [Learn more here](#).
  - Use it to identify resources that aren't configured for backup, and ensure that you don't ever miss protecting critical data in your growing estate.
  - The dashboard provides operational activities for the last seven days (maximum). If you need to retain this data, then you can export as an Excel file and retain them.
  - If you're an Azure Lighthouse user, you can view information across multiple tenants, enabling boundary-less monitoring.
- If you need to retain and view the operational activities for long-term, then use **Reports**. A common requirement for backup admins is to obtain insights on backups based on data that spans an extended period of time. Use cases for such a solution include:
  - Allocating and forecasting of cloud storage consumed.
  - Auditing of backups and restores.
  - Identifying key trends at different levels of granularity.
- In addition,
  - You can send data (for example, jobs, policies, and so on) to the **Log Analytics** workspace. This will enable the features of Azure Monitor Logs to enable correlation of data with other monitoring data collected by Azure Monitor,

consolidate log entries from multiple Azure subscriptions and tenants into one location for analysis together, use log queries to perform complex analysis and gain deep insights on Log entries. [Learn more here](#).

- You can send data to an Azure event hub to send entries outside of Azure, for example to a third-party SIEM (Security Information and Event Management) or other log analytics solution. [Learn more here](#).
- You can send data to an Azure Storage account if you want to retain your log data longer than 90 days for audit, static analysis, or back up. If you only need to retain your events for 90 days or less, you don't need to set up archives to a storage account, since Activity Log events are kept in the Azure platform for 90 days. [Learn more](#).

## Alerts

In a scenario where your backup/restore job failed due to some unknown issue. To assign an engineer to debug it, you would want to be notified about the failure as soon as possible. There could also be a scenario where someone maliciously performs a destructive operation, such as deleting backup items or turning off soft-delete, and you would require an alert message for such incident.

You can configure such critical alerts and route them to any preferred notification channel (email, ITSM, webhook, runbook, and so on). Azure Backup integrates with multiple Azure services to meet different alerting and notification requirements:

- **Azure Monitor Logs (Log Analytics):** You can configure your [vaults to send data to a Log Analytics workspace](#), write custom queries on the workspace, and configure alerts to be generated based on the query output. You can view the query results in tables and charts; also, export them to Power BI or Grafana. (Log Analytics is also a key component of the reporting/auditing capability described in the later sections).
- **Azure Monitor Alerts:** For certain default scenarios, such as backup failure, restore failure, backup data deletion, and so on, Azure Backup sends alerts by default that are surfaced using Azure Monitor, without the need for a user to set up a Log Analytics workspace.
- **Azure Backup provides an in-built alert** notification mechanism via e-mail for failures, warnings, and critical operations. You can specify individual email addresses or distribution lists to be notified when an alert is generated. You can also choose whether to get notified for each individual alert or to group them in an hourly digest and then get notified.

- These alerts are defined by the service and provide support for limited scenarios - backup/restore failures, Stop protection with retain data/Stop protection with delete data, and so on. [Learn more here](#).
- If a destructive operation such as stop protection with delete data is performed, an alert is raised and an email is sent to subscription owners, admins, and co-admins even if notifications are **not** configured for the Recovery Services vault.
- Certain workloads can generate high frequency of failures (for example, SQL Server every 15 minutes). To prevent getting overwhelmed with alerts raised for each failure occurrence, the alerts are consolidated. [Learn more here](#).
- The in-built alerts can't be customized and are restricted to emails defined in the Azure portal.
- If you need to **create custom alerts** (for example, alerts of successful jobs) then use Log Analytics. In Azure Monitor, you can create your own alerts in a Log Analytics workspace. Hybrid workloads (DPM/MABS) can also send data to LA and use LA to provide common alerts across workloads supported by Azure Backup.
- You can also get notifications through built-in Recovery Services vault **activity logs**. However, it supports limited scenarios and isn't suitable for operations such as scheduled backup, which aligns better with resource logs than with activity logs. To learn more about these limitations and how you can use Log Analytics workspace for monitoring and alerting at scale for all your workloads that are protected by Azure Backup, refer to this [article](#).

## Automatic Retry of Failed Backup Jobs

Many of the failure errors or the outage scenarios are transient in nature, and you can remediate by setting up the right Azure role-based access control (Azure RBAC) permissions or re-trigger the backup/restore job. As the solution to such failures is simple, that you don't need to invest time waiting for an engineer to manually trigger the job or to assign the relevant permission. Therefore, the smarter way to handle this scenario is to automate the retry of the failed jobs. This will highly minimize the time taken to recover from failures. You can achieve this by retrieving relevant backup data via Azure Resource Graph (ARG) and combine it with corrective [PowerShell/CLI procedure](#).

Watch the following video to learn how to re-trigger backup for all failed jobs (across vaults, subscriptions, tenants) using ARG and PowerShell.

<https://www.youtube-nocookie.com/embed/8dioCgHNb5w>

## Route Alerts to your preferred notification channel

While transient errors can be corrected, some persistent errors might require in-depth analysis, and retriggering the jobs may not be the viable solution. You may have your own monitoring/ticketing mechanisms to ensure such failures are properly tracked and fixed. To handle such scenarios, you can choose to route the alerts to your preferred notification channel (email, ITSM, Webhook, runbook, and so on) by creating an Action Rule on the alert.

Watch the following video to learn how to leverage Azure Monitor to configure various notification mechanisms for critical alerts.

<https://www.youtube-nocookie.com/embed/oYulJKEPOYY>

## Next steps

Read the following articles as starting points for using Azure Backup:

- [Azure Backup overview](#)
- [Frequently Asked Questions](#)

# Azure to Azure disaster recovery architecture

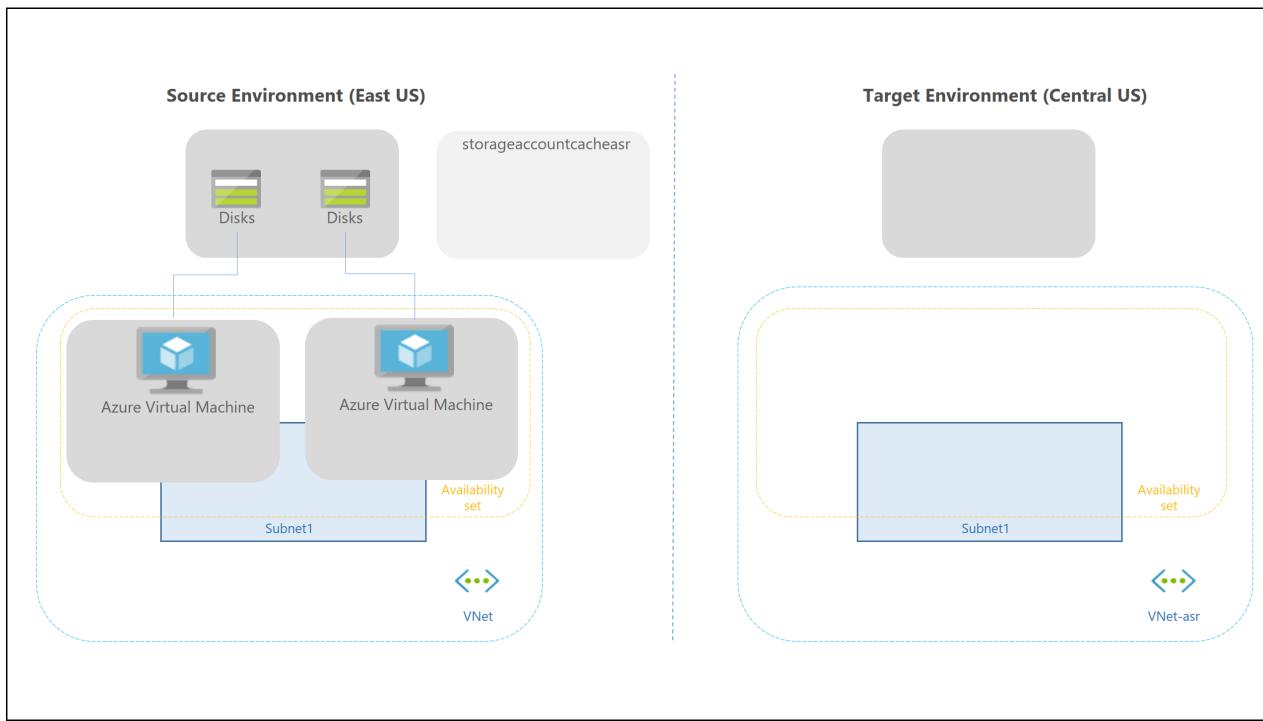
Article • 03/27/2023

This article describes the architecture, components, and processes used when you deploy disaster recovery for Azure virtual machines (VMs) using the [Azure Site Recovery](#) service. With disaster recovery set up, Azure VMs continuously replicate to a different target region. If an outage occurs, you can fail over VMs to the secondary region, and access them from there. When everything's running normally again, you can fail back and continue working in the primary location.

## Architectural components

The components involved in disaster recovery for Azure VMs are summarized in the following table.

Component	Requirements
VMs in source region	One of more Azure VMs in a <a href="#">supported source region</a> . VMs can be running any <a href="#">supported operating system</a> .
Source VM storage	Azure VMs can be managed, or have non-managed disks spread across storage accounts. <a href="#">Learn about</a> supported Azure storage.
Source VM networks	VMs can be located in one or more subnets in a virtual network (VNet) in the source region. <a href="#">Learn more</a> about networking requirements.
Cache storage account	You need a cache storage account in the source network. During replication, VM changes are stored in the cache before being sent to target storage. Cache storage accounts must be Standard.  Using a cache ensures minimal impact on production applications that are running on a VM.  <a href="#">Learn more</a> about cache storage requirements.
Target resources	Target resources are used during replication, and when a failover occurs. Site Recovery can set up target resource by default, or you can create/customize them.  In the target region, check that you're able to create VMs, and that your subscription has enough resources to support VM sizes that will be needed in the target region.



## Target resources

When you enable replication for a VM, Site Recovery gives you the option of creating target resources automatically.

Target resource	Default setting
Target subscription	Same as the source subscription.
Target resource group	<p>The resource group to which VMs belong after failover. It can be in any Azure region except the source region.</p> <p>Site Recovery creates a new resource group in the target region, with an "asr" suffix.</p>
Target VNet	<p>The virtual network (VNet) in which replicated VMs are located after failover. A network mapping is created between source and target virtual networks, and vice versa.</p> <p>Site Recovery creates a new VNet and subnet, with the "asr" suffix.</p>
Target storage account	<p>If the VM doesn't use a managed disk, this is the storage account to which data is replicated. Site Recovery creates a new storage account in the target region, to mirror the source storage account.</p>
Replica managed disks	<p>If the VM uses a managed disk, this is the managed disks to which data is replicated. Site Recovery creates replica managed disks in the storage region to mirror the source.</p>

Target resource	Default setting
Target availability sets	Availability set in which replicating VMs are located after failover. Site Recovery creates an availability set in the target region with the suffix "asr", for VMs that are located in an availability set in the source location. If an availability set exists, it's used and a new one isn't created.
Target availability zones	If the target region supports availability zones, Site Recovery assigns the same zone number as that used in the source region.

## Managing target resources

You can manage target resources as follows:

- You can modify target settings as you enable replication. Please note that the default SKU for the target region VM is the same as the SKU of the source VM (or the next best available SKU in comparison to the source VM SKU). The dropdown list only shows relevant SKUs of the same family as the source VM (Gen 1 or Gen 2).
- You can modify target settings after replication is already working. Similar to other resources such as the target resource group, target name, and others, the target region VM SKU can also be updated after replication is in progress. A resource which cannot be updated is the availability type (single instance, set or zone). To change this setting, you need to disable replication, modify the setting, and then reenable.

## Replication policy

When you enable Azure VM replication, Site Recovery creates a new replication policy with the default settings summarized in the table, by default.

Policy setting	Details	Default
Recovery point retention	Specifies how long Site Recovery keeps recovery points.	1 day
App-consistent snapshot frequency	How often Site Recovery takes an app-consistent snapshot.	0 hours (Disabled)

## Managing replication policies

You can manage and modify the settings of default replication policies as follows:

- You can modify the settings as you enable replication.
- You can create a replication policy at any time, and then apply it when you enable replication.

 **Note**

High recovery point retention period may have an implication on the storage cost since more recovery points may need to be saved.

## Multi-VM consistency

If you want VMs to replicate together, and have shared crash-consistent and app-consistent recovery points at failover, you can gather them together into a replication group. Multi-VM consistency impacts workload performance, and should only be used for VMs running workloads that need consistency across all machines.

## Snapshots and recovery points

Recovery points are created from snapshots of VM disks taken at a specific point in time. When you fail over a VM, you use a recovery point to restore the VM in the target location.

When failing over, we generally want to ensure that the VM starts with no corruption or data loss, and that the VM data is consistent for the operating system, and for apps that run on the VM. This depends on the type of snapshots taken.

Site Recovery takes snapshots as follows:

1. Site Recovery takes crash-consistent snapshots of data by default, and app-consistent snapshots if you specify a frequency for them.
2. Recovery points are created from the snapshots, and stored in accordance with retention settings in the replication policy.

## Consistency

The following table explains different types of consistency.

## Crash-consistent

Description	Details	Recommendation
-------------	---------	----------------

Description	Details	Recommendation
A crash consistent snapshot captures data that was on the disk when the snapshot was taken. It doesn't include anything in memory.	Site Recovery creates crash-consistent recovery points every five minutes by default. This setting can't be modified.	Today, most apps can recover well from crash-consistent points.
It contains the equivalent of the on-disk data that would be present if the VM crashed or the power cord was pulled from the server at the instant that the snapshot was taken.		Crash-consistent recovery points are usually sufficient for the replication of operating systems, and apps such as DHCP servers and print servers.
A crash-consistent doesn't guarantee data consistency for the operating system, or for apps on the VM.		

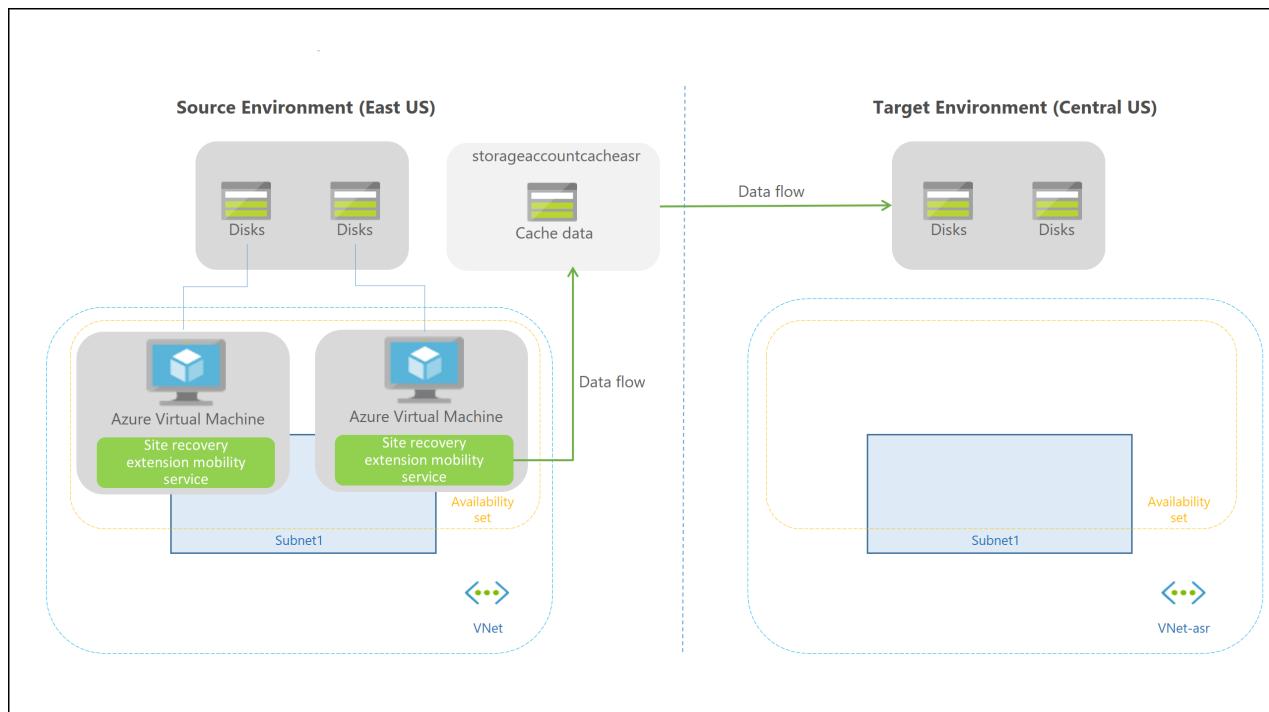
## App-consistent

Description	Details	Recommendation
<p>App-consistent recovery points are created from app-consistent snapshots.</p> <p>An app-consistent snapshot contains all the information in a crash-consistent snapshot, plus all the data in memory and transactions in progress.</p>	<p>App-consistent snapshots use the Volume Shadow Copy Service (VSS):</p> <ol style="list-style-type: none"> <li>1) Azure Site Recovery uses Copy Only backup (VSS_BT_COPY) method which does not change Microsoft SQL's transaction log backup time and sequence number</li> <li>2) When a snapshot is initiated, VSS performs a copy-on-write (COW) operation on the volume.</li> <li>3) Before it performs the COW, VSS informs every app on the machine that it needs to flush its memory-resident data to disk.</li> <li>4) VSS then allows the backup/disaster recovery app (in this case Site Recovery) to read the snapshot data and proceed.</li> </ol>	<p>App-consistent snapshots are taken in accordance with the frequency you specify. This frequency should always be less than you set for retaining recovery points. For example, if you retain recovery points using the default setting of 24 hours, you should set the frequency at less than 24 hours.</p> <p>They're more complex and take longer to complete than crash-consistent snapshots.</p> <p>They affect the performance of apps running on a VM enabled for replication.</p>

## Replication process

When you enable replication for an Azure VM, the following happens:

1. The Site Recovery Mobility service extension is automatically installed on the VM.
2. The extension registers the VM with Site Recovery.
3. Continuous replication begins for the VM. Disk writes are immediately transferred to the cache storage account in the source location.
4. Site Recovery processes the data in the cache, and sends it to the target storage account, or to the replica managed disks.
5. After the data is processed, crash-consistent recovery points are generated every five minutes. App-consistent recovery points are generated according to the setting specified in the replication policy.



## Replication process

# Connectivity requirements

The Azure VMs you replicate need outbound connectivity. Site Recovery never needs inbound connectivity to the VM.

## Outbound connectivity (URLs)

If outbound access for VMs is controlled with URLs, allow these URLs.

Name	Commercial	Government	Description
------	------------	------------	-------------

Name	Commercial	Government	Description
Storage	<code>*.blob.core.windows.net</code>	<code>*.blob.core.usgovcloudapi.net</code>	Allows data to be written from the VM to the cache storage account in the source region.
Azure Active Directory	<code>login.microsoftonline.com</code>	<code>login.microsoftonline.us</code>	Provides authorization and authentication to Site Recovery service URLs.
Replication	<code>*.hypervrecoverymanager.windowsazure.com</code>	<code>*.hypervrecoverymanager.windowsazure.us</code>	Allows the VM to communicate with the Site Recovery service.
Service Bus	<code>*.servicebus.windows.net</code>	<code>*.servicebus.usgovcloudapi.net</code>	Allows the VM to write Site Recovery monitoring and diagnostics data.
Key Vault	<code>*.vault.azure.net</code>	<code>*.vault.usgovcloudapi.net</code>	Allows access to enable replication for ADE-enabled virtual machines via portal
Azure Automation	<code>*.automation.ext.azure.com</code>	<code>*.azure-automation.us</code>	Allows enabling auto-upgrade of mobility agent for a replicated item via portal

## Outbound connectivity for IP address ranges

To control outbound connectivity for VMs using IP addresses, allow these addresses. Please note that details of network connectivity requirements can be found in [networking white paper](#)

## Source region rules

Rule	Details	Service tag
Allow HTTPS outbound: port 443	Allow ranges that correspond to storage accounts in the source region	Storage.<region-name>
Allow HTTPS outbound: port 443	Allow ranges that correspond to Azure Active Directory (Azure AD)	AzureActiveDirectory
Allow HTTPS outbound: port 443	Allow ranges that correspond to Events Hub in the target region.	EventHub.<region-name>
Allow HTTPS outbound: port 443	Allow ranges that correspond to Azure Site Recovery	AzureSiteRecovery
Allow HTTPS outbound: port 443	Allow ranges that correspond to Azure Key Vault (This is required only for enabling replication of ADE-enabled virtual machines via portal)	AzureKeyVault
Allow HTTPS outbound: port 443	Allow ranges that correspond to Azure Automation Controller (This is required only for enabling auto-upgrade of mobility agent for a replicated item via portal)	GuestAndHybridManagement

## Target region rules

Rule	Details	Service tag
Allow HTTPS outbound: port 443	Allow ranges that correspond to storage accounts in the target region	Storage.<region-name>
Allow HTTPS outbound: port 443	Allow ranges that correspond to Azure AD	AzureActiveDirectory
Allow HTTPS outbound: port 443	Allow ranges that correspond to Events Hub in the source region.	EventHub.<region-name>
Allow HTTPS outbound: port 443	Allow ranges that correspond to Azure Site Recovery	AzureSiteRecovery
Allow HTTPS outbound: port 443	Allow ranges that correspond to Azure Key Vault (This is required only for enabling replication of ADE-enabled virtual machines via portal)	AzureKeyVault
Allow HTTPS outbound: port 443	Allow ranges that correspond to Azure Automation Controller (This is required only for enabling auto-upgrade of mobility agent for a replicated item via portal)	GuestAndHybridManagement

## Control access with NSG rules

If you control VM connectivity by filtering network traffic to and from Azure networks/subnets using [NSG rules](#), note the following requirements:

- NSG rules for the source Azure region should allow outbound access for replication traffic.
- We recommend you create rules in a test environment before you put them into production.
- Use [service tags](#) instead of allowing individual IP addresses.
  - Service tags represent a group of IP address prefixes gathered together to minimize complexity when creating security rules.
  - Microsoft automatically updates service tags over time.

Learn more about [outbound connectivity](#) for Site Recovery, and [controlling connectivity with NSGs](#).

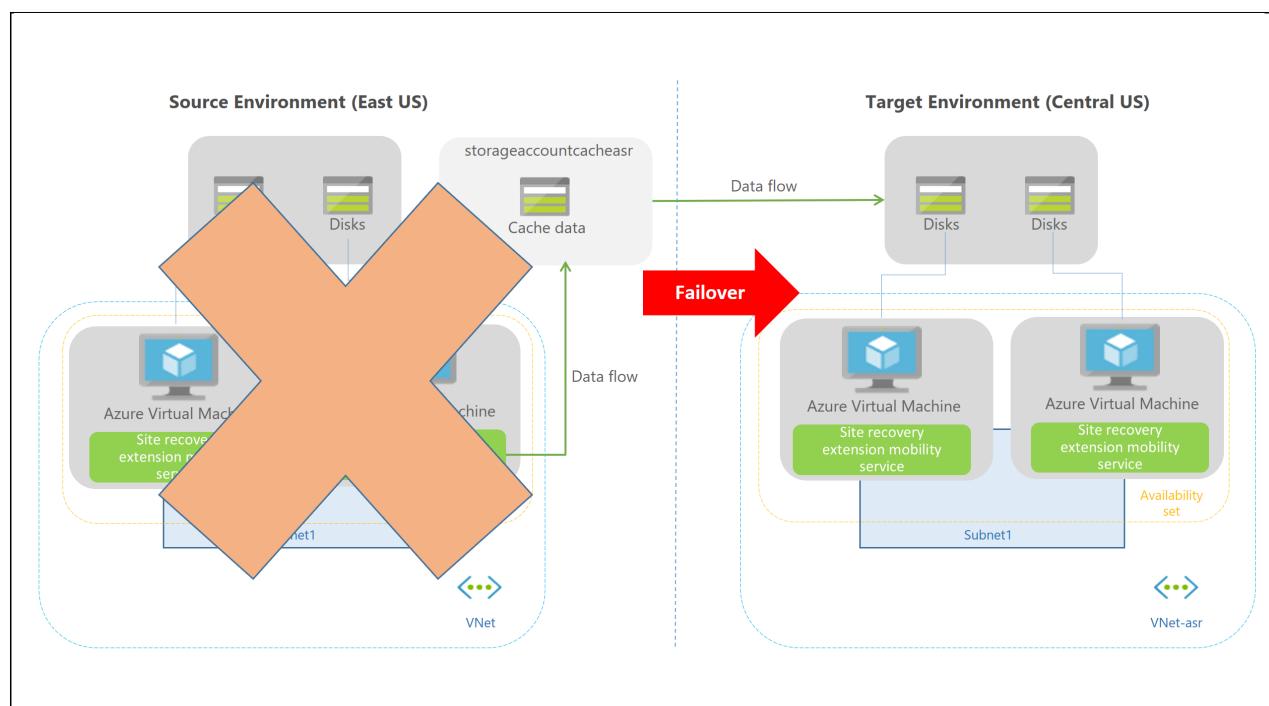
## Connectivity for multi-VM consistency

If you enable multi-VM consistency, machines in the replication group communicate with each other over port 20004.

- Ensure that there is no firewall appliance blocking the internal communication between the VMs over port 20004.
- If you want Linux VMs to be part of a replication group, ensure the outbound traffic on port 20004 is manually opened as per the guidance of the specific Linux version.

## Failover process

When you initiate a failover, the VMs are created in the target resource group, target virtual network, target subnet, and in the target availability set. During a failover, you can use any recovery point.



## Next steps

[Quickly replicate](#) an Azure VM to a secondary region.

# Support matrix for Azure VM disaster recovery between Azure regions

Article • 12/19/2023

This article summarizes support and prerequisites for disaster recovery of Azure VMs from one Azure region to another, using the [Azure Site Recovery](#) service.

## Deployment method support

Expand table

Deployment	Support
Azure portal	Supported.
PowerShell	Supported. <a href="#">Learn more</a>
REST API	Supported.
CLI	Not currently supported.

## Resource move/migrate support

Expand table

Resource action	Details
Move vaults across resource groups	Not supported.
Move compute/storage/network resources across resource groups	Not supported.  If you move a VM or associated components such as storage/network after the VM is replicating, you need to disable and then re-enable replication for the VM.
Replicate Azure VMs from one subscription to another for disaster recovery	Supported within the same Microsoft Entra tenant.
Migrate VMs across regions within supported geographical clusters (within and across subscriptions)	Supported within the same Microsoft Entra tenant.

Resource action	Details
Migrate VMs within the same region	Not supported.
Azure Dedicated Hosts	Not supported.

## Region support

Azure Site Recovery allows you to perform global disaster recovery. You can replicate and recover VMs between any two Azure regions in the world. If you have concerns around data sovereignty, you may choose to limit replication within your specific geographic cluster. The various geographic clusters are as follows:

[\[+\] Expand table](#)

Geographic cluster	Azure regions
America	Canada East, Canada Central, South Central US, West Central US, East US, East US 2, West US, West US 2, West US 3, Central US, North Central US
Europe	UK West, UK South, North Europe, West Europe, South Africa West, South Africa North, Norway East, France Central, Switzerland North, Germany West Central, UAE North (UAE is treated as part of the Europe geo cluster)
Asia	South India, Central India, West India, Southeast Asia, East Asia, Japan East, Japan West, Korea Central, Korea South, Qatar Central
JIO	JIO India West  Replication can't be done between JIO and non-JIO regions for Virtual Machines present in JIO subscriptions. This is because JIO subscriptions can have resources only in JIO regions.
Australia	Australia East, Australia Southeast, Australia Central, Australia Central 2
Azure Government	US GOV Virginia, US GOV Iowa, US GOV Arizona, US GOV Texas, US DOD East, US DOD Central
Germany	Germany Central, Germany Northeast
China	China East, China North, China North2, China East2
Brazil	Brazil South
Restricted Regions reserved for in-	Switzerland West reserved for Switzerland North, France South reserved for France Central, Norway West for Norway East customers, JIO India

Geographic cluster	Azure regions
country/region disaster recovery	<p>Central for JIO India West customers, Brazil Southeast for Brazil South customers, South Africa West for South Africa North customers, Germany North for Germany West Central customers, UAE Central for UAE North customers.</p> <p>To use restricted regions as your primary or recovery region, get yourselves allowlisted by raising a request <a href="#">here</a> for both source and target subscriptions.</p>

 **Note**

- For **Brazil South**, you can replicate and fail over to these regions: Brazil Southeast, South Central US, West Central US, East US, East US 2, West US, West US 2, and North Central US.
- Brazil South can only be used as a source region from which VMs can replicate using Site Recovery. It can't act as a target region. Note that if you fail over from Brazil South as a source region to a target, fallback to Brazil South from the target region is supported. Brazil Southeast can only be used as a target region.
- If the region in which you want to create a vault doesn't show, make sure your subscription has access to create resources in that region.
- If you can't see a region within a geographic cluster when you enable replication, make sure your subscription has permissions to create VMs in that region.

## Cache storage

This table summarizes support for the cache storage account used by Site Recovery during replication.

 [Expand table](#)

Setting	Support	Details
General purpose V2 storage accounts (Hot and Cool tier)	Supported	Usage of GPv2 is recommended because GPv1 doesn't support ZRS (Zonal Redundant Storage).

Setting	Support	Details
Premium storage	Supported	Use Premium Block Blob storage accounts to get High Churn support. For more information, see <a href="#">Azure VM Disaster Recovery - High Churn Support</a> .
Region	Same region as virtual machine	Cache storage account should be in the same region as the virtual machine being protected.
Subscription	Can be different from source virtual machines	Cache storage account need not be in the same subscription as the source virtual machine(s).
Azure Storage firewalls for virtual networks	Supported	<p>If you're using firewall enabled cache storage account or target storage account, ensure you '<a href="#">Allow trusted Microsoft services</a>'.</p> <p>Also, ensure that you allow access to at least one subnet of source Vnet.</p> <p>Note: Don't restrict virtual network access to your storage accounts used for Site Recovery. You should allow access from 'All networks'.</p>
Soft delete	Not supported	Soft delete isn't supported because once it is enabled on cache storage account, it increases cost. Azure Site Recovery performs frequent creates/deletes of log files while replicating causing costs to increase.
Encryption at rest (CMK)	Supported	Storage account encryption can be configured with customer managed keys (CMK)
Managed identity	Not supported	The cached storage account must allow shared key access and Shared Access Signatures (SAS) signed by the shared key.

The following table lists the limits in terms of number of disks that can replicate to a single storage account.

[\[+\] Expand table](#)

Storage account type	Churn = 4 MBps per disk	Churn = 8 MBps per disk
V1 storage account	300 disks	150 disks
V2 storage account	750 disks	375 disks

As average churn on the disks increases, the number of disks that a storage account can support decreases. The above table may be used as a guide for making decisions on

number of storage accounts that need to be provisioned.

Note that the above limits are specific to Azure-to-Azure and Zone-to-Zone DR scenarios.

## Replicated machine operating systems

Site Recovery supports replication of Azure VMs running the operating systems listed in this section. Note that if an already-replicating machine is later upgraded (or downgraded) to a different major kernel, you need to disable replication and re-enable replication after the upgrade.

### Windows

Expand table

Operating system	Details
Windows Server 2022	Supported.
Windows Server 2019	Supported for Server Core, Server with Desktop Experience.
Windows Server 2016	Supported Server Core, Server with Desktop Experience.
Windows Server 2012 R2	Supported.
Windows Server 2012	Supported.
Windows Server 2008 R2 with SP1/SP2	From version <a href="#">9.30</a> of the Mobility service extension for Azure VMs, you need to install a Windows <a href="#">servicing stack update (SSU)</a> and <a href="#">SHA-2 update</a> on machines running Windows Server 2008 R2 SP1/SP2. SHA-1 isn't supported from September 2019, and if SHA-2 code signing isn't enabled the agent extension won't install/upgrade as expected. Learn more about <a href="#">SHA-2 upgrade and requirements</a> .
Windows 11 (x64)	Supported (From Mobility Agent version 9.56 onwards).

Operating system	Details
Windows 10 (x64)	Supported.
Windows 8.1 (x64)	Supported.
Windows 8 (x64)	Supported.
Windows 7 (x64) with SP1 onwards	From version <a href="#">9.30</a> of the Mobility service extension for Azure VMs, you need to install a Windows <a href="#">servicing stack update (SSU)</a> and <a href="#">SHA-2 update</a> on machines running Windows 7 with SP1. SHA-1 isn't supported from September 2019, and if SHA-2 code signing isn't enabled the agent extension won't install/upgrade as expected. Learn more about <a href="#">SHA-2 upgrade and requirements</a> .

## Linux

[\[ \]](#) [Expand table](#)

Operating system	Details
Red Hat Enterprise Linux	6.7, 6.8, 6.9, 6.10, 7.0, 7.1, 7.2, 7.3, 7.4, 7.5, 7.6, <a href="#">7.7</a> , <a href="#">7.8</a> , <a href="#">7.9</a> , <a href="#">8.0</a> , 8.1, <a href="#">8.2</a> , <a href="#">8.3</a> , <a href="#">8.4</a> (4.18.0-305.30.1.el8_4.x86_64 or higher), <a href="#">8.5</a> (4.18.0-348.5.1.el8_5.x86_64 or higher), <a href="#">8.6</a> , 8.7, 8.8, 8.9, 9.0. <b>Note:</b> Support for Red Hat Enterprise Linux 9.1 is removed from support matrix as issues were observed while using Azure Site Recovery with Red Hat Enterprise Linux 9.1.
CentOS	6.5, 6.6, 6.7, 6.8, 6.9, 6.10 7.0, 7.1, 7.2, 7.3, 7.4, 7.5, 7.6, 7.7, <a href="#">7.8</a> , <a href="#">7.9 pre-GA version</a> , 7.9 GA version is supported from 9.37 hot fix patch** 8.0, 8.1, <a href="#">8.2</a> , <a href="#">8.3</a> , 8.4 (4.18.0-305.30.1.el8_4.x86_64 or later), 8.5 (4.18.0-348.5.1.el8_5.x86_64 or later), 8.6, 8.7.
Ubuntu 14.04 LTS Server	Includes support for all 14.04.x versions; <a href="#">Supported kernel versions</a> ;
Ubuntu 16.04 LTS Server	Includes support for all 16.04.x versions; <a href="#">Supported kernel version</a> Ubuntu servers using password-based authentication and sign-in, and the cloud-init package to configure cloud VMs, might have password-based sign-in disabled on failover (depending on the cloudinit configuration). Password-based sign in can be re-enabled on the virtual machine by resetting the password from

Operating system	Details
	the Support > Troubleshooting > Settings menu (of the failed over VM in the Azure portal).
Ubuntu 18.04 LTS Server	<p>Includes support for all 18.04.x versions; <a href="#">Supported kernel version</a></p> <p>Ubuntu servers using password-based authentication and sign-in, and the cloud-init package to configure cloud VMs, might have password-based sign-in disabled on failover (depending on the cloudinit configuration). Password-based sign in can be re-enabled on the virtual machine by resetting the password from the Support &gt; Troubleshooting &gt; Settings menu (of the failed over VM in the Azure portal).</p>
Ubuntu 20.04 LTS server	<p>Includes support for all 20.04.x versions; <a href="#">Supported kernel version</a></p>
Ubuntu 22.04 LTS server	<p>Includes support for all 22.04.x versions; <a href="#">Supported kernel version</a></p>
Debian 7	<p>Includes support for all 7. x versions <a href="#">Supported kernel versions</a></p>
Debian 8	<p>Includes support for all 8. x versions <a href="#">Supported kernel versions</a></p>
Debian 9	<p>Includes support for 9.1 to 9.13. Debian 9.0 isn't supported. <a href="#">Supported kernel versions</a></p>
Debian 10	<p><a href="#">Supported kernel versions</a></p>
Debian 11	<p><a href="#">Supported kernel versions</a></p>
SUSE Linux Enterprise Server 12	<p>SP1, SP2, SP3, SP4, SP5 (<a href="#">Supported kernel versions</a>)</p>
SUSE Linux Enterprise Server 15	<p>15, SP1, SP2, SP3, SP4, SP5 (<a href="#">Supported kernel versions</a>)</p>
SUSE Linux Enterprise Server 11	<p>SP3</p> <p>Upgrade of replicating machines from SP3 to SP4 isn't supported. If a replicated machine has been upgraded, you need to disable replication and re-enable replication after the upgrade.</p>
SUSE Linux Enterprise Server 11	<p>SP4</p>
Oracle Linux	<p>6.4, 6.5, 6.6, 6.7, 6.8, 6.9, 6.10, 7.0, 7.1, 7.2, 7.3, 7.4, 7.5, 7.6, <a href="#">7.7 ↗</a>, <a href="#">7.8 ↗</a>, <a href="#">7.9 ↗</a>, <a href="#">8.0 ↗</a>, <a href="#">8.1 ↗</a>, <a href="#">8.2 ↗</a>, <a href="#">8.3 ↗</a> (running the Red Hat compatible kernel or Unbreakable</p>

Operating system	Details
	<p>Enterprise Kernel Release 3, 4, 5, and 6 (UEK3, UEK4, UEK5, UEK6), <a href="#">8.4</a>, 8.5, 8.6, 8.7, 8.8, 8.9, 9.0</p> <p><b>Notes:</b> Support for Oracle Linux <a href="#">9.1</a> is removed from support matrix as issues were observed while using Azure Site Recovery with Oracle Linux 9.1.</p> <p>8.1 (running on all UEK kernels and RedHat kernel &lt;= 3.10.0-1062.* are supported in <a href="#">9.35</a>. Support for rest of the RedHat kernels is available in <a href="#">9.36</a>).</p>
Rocky Linux	<a href="#">See supported versions.</a>

 **Note**

For Linux versions, Azure Site Recovery doesn't support custom OS kernels. Only the stock kernels that are part of the distribution minor version release/update are supported.

 **Note**

To support latest Linux kernels within 15 days of release, Azure Site Recovery rolls out hot fix patch on top of latest mobility agent version. This fix is rolled out in between two major version releases. To update to latest version of mobility agent (including hot fix patch), follow steps mentioned in [this article](#). This patch is currently rolled out for mobility agents used in Azure to Azure DR scenario.

## Supported Ubuntu kernel versions for Azure virtual machines

 [Expand table](#)

Release	Mobility service version	Kernel version
14.04 LTS	<a href="#">9.57</a>	No new 14.04 LTS kernels supported in this release.
14.04 LTS	<a href="#">9.56</a>	No new 14.04 LTS kernels supported in this release.
14.04 LTS	<a href="#">9.55</a>	No new 14.04 LTS kernels supported in this release.
14.04 LTS	<a href="#">9.54</a>	No new 14.04 LTS kernels supported in this release.
14.04 LTS	<a href="#">9.53</a>	No new 14.04 LTS kernels supported in this release.

Release	Mobility service version	Kernel version
16.04 LTS	<a href="#">9.57 ↗</a>	No new 16.04 LTS kernels supported in this release.
16.04 LTS	<a href="#">9.56 ↗</a>	No new 16.04 LTS kernels supported in this release.
16.04 LTS	<a href="#">9.55 ↗</a>	No new 16.04 LTS kernels supported in this release.
16.04 LTS	<a href="#">9.54 ↗</a>	No new 16.04 LTS kernels supported in this release.
16.04 LTS	<a href="#">9.53 ↗</a>	No new 16.04 LTS kernels supported in this release.
<hr/>		
18.04 LTS	<a href="#">9.57 ↗</a>	No new 18.04 LTS kernels supported in this release.
18.04 LTS	<a href="#">9.56 ↗</a>	No new 18.04 LTS kernels supported in this release.
18.04 LTS	<a href="#">9.55 ↗</a>	4.15.0-1166-azure 4.15.0-1167-azure 4.15.0-212-generic 4.15.0-213-generic 5.4.0-1108-azure 5.4.0-1109-azure 5.4.0-149-generic 5.4.0-150-generic
18.04 LTS	<a href="#">9.54 ↗</a>	4.15.0-208-generic 4.15.0-209-generic 5.4.0-1105-azure 5.4.0-1106-azure 5.4.0-146-generic 4.15.0-1163-azure 4.15.0-1164-azure 4.15.0-1165-azure 4.15.0-210-generic 4.15.0-211-generic 5.4.0-1107-azure 5.4.0-147-generic 5.4.0-147-generic 5.4.0-148-generic 4.15.0-212-generic 4.15.0-1166-azure 5.4.0-149-generic 5.4.0-150-generic 5.4.0-1108-azure 5.4.0-1109-azure
18.04 LTS	<a href="#">9.53 ↗</a>	5.4.0-137-generic 5.4.0-1101-azure 4.15.0-1161-azure 4.15.0-204-generic

Release	Mobility service version	Kernel version
		5.4.0-1103-azure 5.4.0-139-generic 4.15.0-206-generic 5.4.0-1104-azure 5.4.0-144-generic 4.15.0-1162-azure
20.04 LTS	<a href="#">9.57 ↗</a>	5.15.0-1052-azure 5.15.0-1053-azure 5.15.0-89-generic 5.15.0-91-generic 5.4.0-1120-azure 5.4.0-1121-azure 5.4.0-167-generic 5.4.0-169-generic
20.04 LTS	<a href="#">9.56 ↗</a>	5.15.0-1049-azure 5.15.0-1050-azure 5.15.0-1051-azure 5.15.0-86-generic 5.15.0-87-generic 5.15.0-88-generic 5.4.0-1117-azure 5.4.0-1118-azure 5.4.0-1119-azure 5.4.0-164-generic 5.4.0-165-generic 5.4.0-166-generic
20.04 LTS	<a href="#">9.55 ↗</a>	5.15.0-1039-azure 5.15.0-1040-azure 5.15.0-1041-azure 5.15.0-73-generic 5.15.0-75-generic 5.15.0-76-generic 5.4.0-1108-azure 5.4.0-1109-azure 5.4.0-1110-azure 5.4.0-1111-azure 5.4.0-149-generic 5.4.0-150-generic 5.4.0-152-generic 5.4.0-153-generic 5.4.0-155-generic 5.4.0-1112-azure 5.15.0-78-generic 5.15.0-1042-azure

Release	Mobility service version	Kernel version
		5.15.0-79-generic 5.4.0-156-generic 5.15.0-1047-azure 5.15.0-84-generic 5.4.0-1116-azure 5.4.0-163-generic 5.15.0-1043-azure 5.15.0-1045-azure 5.15.0-1046-azure 5.15.0-82-generic 5.15.0-83-generic
20.04 LTS	<a href="#">9.54 ↗</a>	5.15.0-1035-azure 5.15.0-1036-azure 5.15.0-69-generic 5.4.0-1105-azure 5.4.0-1106-azure 5.4.0-146-generic 5.4.0-147-generic 5.15.0-1037-azure 5.15.0-1038-azure 5.15.0-70-generic 5.15.0-71-generic 5.15.0-72-generic 5.4.0-1107-azure 5.4.0-148-generic 5.4.0-149-generic 5.4.0-150-generic 5.4.0-1108-azure 5.4.0-1109-azure 5.15.0-73-generic 5.15.0-1039-azure
20.04 LTS	<a href="#">9.53 ↗</a>	5.4.0-1101-azure 5.15.0-1033-azure 5.15.0-60-generic 5.4.0-1103-azure 5.4.0-139-generic 5.15.0-1034-azure 5.15.0-67-generic 5.4.0-1104-azure 5.4.0-144-generic
22.04 LTS	<a href="#">9.57 ↗</a>	5.15.0-1052-azure 5.15.0-1053-azure 5.15.0-76-generic

Release	Mobility service version	Kernel version
		5.15.0-89-generic 5.15.0-91-generic
22.04 LTS	<a href="#">9.56 ↗</a>	5.15.0-1049-azure 5.15.0-1050-azure 5.15.0-1051-azure 5.15.0-86-generic 5.15.0-87-generic 5.15.0-88-generic
22.04 LTS	<a href="#">9.55 ↗</a>	5.15.0-1039-azure 5.15.0-1040-azure 5.15.0-1041-azure 5.15.0-73-generic 5.15.0-75-generic 5.15.0-76-generic 5.15.0-78-generic 5.15.0-1042-azure 5.15.0-1044-azure 5.15.0-79-generic 5.15.0-1047-azure 5.15.0-84-generic 5.15.0-1045-azure 5.15.0-1046-azure 5.15.0-82-generic 5.15.0-83-generic
22.04 LTS	<a href="#">9.54 ↗</a>	5.15.0-1035-azure 5.15.0-1036-azure 5.15.0-69-generic 5.15.0-70-generic 5.15.0-1037-azure 5.15.0-1038-azure 5.15.0-71-generic 5.15.0-72-generic 5.15.0-73-generic 5.15.0-1039-azure
22.04 LTS	<a href="#">9.53 ↗</a>	5.15.0-1003-azure 5.15.0-1005-azure 5.15.0-1007-azure 5.15.0-1008-azure 5.15.0-1010-azure 5.15.0-1012-azure 5.15.0-1013-azure 5.15.0-1014-azure 5.15.0-1017-azure 5.15.0-1019-azure

Release	Mobility service version	Kernel version
		5.15.0-1020-azure
		5.15.0-1021-azure
		5.15.0-1022-azure
		5.15.0-1023-azure
		5.15.0-1024-azure
		5.15.0-1029-azure
		5.15.0-1030-azure
		5.15.0-1031-azure
		5.15.0-25-generic
		5.15.0-27-generic
		5.15.0-30-generic
		5.15.0-33-generic
		5.15.0-35-generic
		5.15.0-37-generic
		5.15.0-39-generic
		5.15.0-40-generic
		5.15.0-41-generic
		5.15.0-43-generic
		5.15.0-46-generic
		5.15.0-47-generic
		5.15.0-48-generic
		5.15.0-50-generic
		5.15.0-52-generic
		5.15.0-53-generic
		5.15.0-56-generic
		5.15.0-57-generic
		5.15.0-58-generic
		5.15.0-1033-azure
		5.15.0-60-generic
		5.15.0-1034-azure
		5.15.0-67-generic

**ⓘ Note**

To support latest Linux kernels within 15 days of release, Azure Site Recovery rolls out hot fix patch on top of latest mobility agent version. This fix is rolled out in between two major version releases. To update to latest version of mobility agent (including hot fix patch) follow steps mentioned in [this article](#). This patch is currently rolled out for mobility agents used in Azure to Azure DR scenario.

## Supported Debian kernel versions for Azure virtual machines

[Expand table](#)

Release	Mobility service version	Kernel version
Debian 7	<a href="#">9.57 ↗</a>	No new Debian 7 kernels supported in this release.
Debian 7	<a href="#">9.56 ↗</a>	No new Debian 7 kernels supported in this release.
Debian 7	<a href="#">9.55 ↗</a>	No new Debian 7 kernels supported in this release.
Debian 7	<a href="#">9.54 ↗</a>	No new Debian 7 kernels supported in this release.
Debian 7	<a href="#">9.53 ↗</a>	No new Debian 7 kernels supported in this release.
Debian 8	<a href="#">9.57 ↗</a>	No new Debian 8 kernels supported in this release.
Debian 8	<a href="#">9.56 ↗</a>	No new Debian 8 kernels supported in this release.
Debian 8	<a href="#">9.55 ↗</a>	No new Debian 8 kernels supported in this release.
Debian 8	<a href="#">9.54 ↗</a>	No new Debian 8 kernels supported in this release.
Debian 8	<a href="#">9.53 ↗</a>	No new Debian 8 kernels supported in this release.
Debian 9.1	<a href="#">9.57 ↗</a>	No new Debian 9.1 kernels supported in this release.
Debian 9.1	<a href="#">9.56 ↗</a>	No new Debian 9.1 kernels supported in this release.
Debian 9.1	<a href="#">9.55 ↗</a>	No new Debian 9.1 kernels supported in this release.
Debian 9.1	<a href="#">9.54 ↗</a>	No new Debian 9.1 kernels supported in this release.
Debian 9.1	<a href="#">9.53 ↗</a>	No new Debian 9.1 kernels supported in this release.
Debian 10	<a href="#">9.57</a>	No new Debian 10 kernels supported in this release.
Debian 10	<a href="#">9.56 ↗</a>	5.10.0-0.deb10.26-amd64 5.10.0-0.deb10.26-cloud-amd64
Debian 10	<a href="#">9.55 ↗</a>	5.10.0-0.deb10.23-amd64 5.10.0-0.deb10.23-cloud-amd64 4.19.0-25-amd64 4.19.0-25-cloud-amd64 5.10.0-0.deb10.24-amd64 5.10.0-0.deb10.24-cloud-amd64
Debian 10	<a href="#">9.54 ↗</a>	5.10.0-0.bpo.3-amd64 5.10.0-0.bpo.3-cloud-amd64 5.10.0-0.bpo.4-amd64 5.10.0-0.bpo.4-cloud-amd64 5.10.0-0.bpo.5-amd64 5.10.0-0.bpo.5-cloud-amd64

Release	Mobility service version	Kernel version
		4.19.0-24-amd64 4.19.0-24-cloud-amd64 5.10.0-0.deb10.22-amd64 5.10.0-0.deb10.22-cloud-amd64 5.10.0-0.deb10.23-amd64 5.10.0-0.deb10.23-cloud-amd64
Debian 10	<a href="#">9.53 ↗</a>	5.10.0-0.deb10.21-amd64 5.10.0-0.deb10.21-cloud-amd64
Debian 11	<a href="#">9.57 ↗</a>	No new Debian 11 kernels supported in this release.
Debian 11	<a href="#">9.56 ↗</a>	5.10.0-26-amd64 5.10.0-26-cloud-amd64
Debian 11	<a href="#">9.55 ↗</a>	5.10.0-24-amd64 5.10.0-24-cloud-amd64 5.10.0-25-amd64 5.10.0-25-cloud-amd64
Debian 11	<a href="#">9.54 ↗</a>	5.10.0-22-amd64 5.10.0-22-cloud-amd64 5.10.0-23-amd64 5.10.0-23-cloud-amd64
Debian 11	<a href="#">9.53 ↗</a>	5.10.0-21-amd64 5.10.0-21-cloud-amd64

 **Note**

To support latest Linux kernels within 15 days of release, Azure Site Recovery rolls out hot fix patch on top of latest mobility agent version. This fix is rolled out in between two major version releases. To update to latest version of mobility agent (including hot fix patch) follow steps mentioned in [this article](#). This patch is currently rolled out for mobility agents used in Azure to Azure DR scenario.

## Supported SUSE Linux Enterprise Server 12 kernel versions for Azure virtual machines

 [Expand table](#)

Release	Mobility service version	Kernel version
SUSE Linux Enterprise Server 12 (SP1, SP2, SP3, SP4, SP5)	<a href="#">9.57</a>	All <a href="#">stock SUSE 12 SP1,SP2,SP3,SP4,SP5 kernels</a> are supported.  4.12.14-16.155-azure:5
SUSE Linux Enterprise Server 12 (SP1, SP2, SP3, SP4, SP5)	<a href="#">9.56</a>	All <a href="#">stock SUSE 12 SP1,SP2,SP3,SP4,SP5 kernels</a> are supported.  4.12.14-16.152-azure:5
SUSE Linux Enterprise Server 12 (SP1, SP2, SP3, SP4, SP5)	<a href="#">9.55</a>	All <a href="#">stock SUSE 12 SP1,SP2,SP3,SP4,SP5 kernels</a> are supported.  4.12.14-16.136-azure:5 4.12.14-16.139-azure:5 4.12.14-16.146-azure:5 4.12.14-16.149-azure:5
SUSE Linux Enterprise Server 12 (SP1, SP2, SP3, SP4, SP5)	<a href="#">9.54</a>	All <a href="#">stock SUSE 12 SP1,SP2,SP3,SP4,SP5 kernels</a> are supported.  4.12.14-16.130-azure:5 4.12.14-16.133-azure:5
SUSE Linux Enterprise Server 12 (SP1, SP2, SP3, SP4, SP5)	<a href="#">9.53</a>	All <a href="#">stock SUSE 12 SP1,SP2,SP3,SP4,SP5 kernels</a> are supported.  4.12.14-16.124-azure:5 4.12.14-16.127-azure:5

## Supported SUSE Linux Enterprise Server 15 kernel versions for Azure virtual machines

[\[ \]](#) Expand table

Release	Mobility service version	Kernel version
SUSE Linux Enterprise Server 15 (SP1, SP2, SP3, SP4, SP5)	<a href="#">9.57</a>	By default, all <a href="#">stock SUSE 15, SP1, SP2, SP3, SP4, SP5 kernels</a> are supported.  5.14.21-150400.14.72-azure:4 5.14.21-150500.33.23-azure:5 5.14.21-150500.33.26-azure:5

Release	Mobility service version	Kernel version
SUSE Linux Enterprise Server 15 (SP1, SP2, SP3, SP4, SP5)	<a href="#">9.56 ↗</a>	By default, all <a href="#">stock SUSE 15, SP1, SP2, SP3, SP4, SP5 kernels ↗</a> are supported.  5.14.21-150400.14.69-azure:4 5.14.21-150500.31-azure:5 5.14.21-150500.33.11-azure:5 5.14.21-150500.33.14-azure:5 5.14.21-150500.33.17-azure:5 5.14.21-150500.33.20-azure:5 5.14.21-150500.33.3-azure:5 5.14.21-150500.33.6-azure:5
SUSE Linux Enterprise Server 15 (SP1, SP2, SP3, SP4)	<a href="#">9.55 ↗</a>	By default, all <a href="#">stock SUSE 15, SP1, SP2, SP3, SP4 kernels ↗</a> are supported.  5.14.21-150400.14.52-azure:4 4.12.14-16.139-azure:5 5.14.21-150400.14.55-azure:4 5.14.21-150400.14.60-azure:4 5.14.21-150400.14.63-azure:4 5.14.21-150400.14.66-azure:4
SUSE Linux Enterprise Server 15 (SP1, SP2, SP3, SP4)	<a href="#">9.54 ↗</a>	By default, all <a href="#">stock SUSE 15, SP1, SP2, SP3, SP4 kernels ↗</a> are supported.  5.14.21-150400.14.40-azure:4 5.14.21-150400.14.43-azure:4 5.14.21-150400.14.46-azure:4 5.14.21-150400.14.49-azure:4
SUSE Linux Enterprise Server 15 (SP1, SP2, SP3, SP4)	<a href="#">9.53 ↗</a>	By default, all <a href="#">stock SUSE 15, SP1, SP2, SP3, SP4 kernels ↗</a> are supported.  5.14.21-150400.14.31-azure:4 5.14.21-150400.14.34-azure:4 5.14.21-150400.14.37-azure:4

## Supported Rocky Linux kernel versions for Azure virtual machines

[Expand table](#)

Release	Mobility service version	Kernel version
Rocky Linux	<a href="#">9.57 ↗</a>	Rocky Linux 8.8 Rocky Linux 8.9

Release	Mobility service version	Kernel version
Rocky Linux	<a href="#">9.56</a>	Rocky Linux 8.7 Rocky Linux 9.0

**ⓘ Note**

Support for Rocky Linux 9.1 is removed from the support matrix as issues were observed while using it with Azure Site Recovery.

**ⓘ Important**

To support latest Linux kernels within 15 days of release, Azure Site Recovery rolls out hot fix patch on top of latest mobility agent version. This fix is rolled out in between two major version releases. To update to latest version of mobility agent (including hot fix patch) follow steps mentioned in [this article](#). This patch is currently rolled out for mobility agents used in Azure to Azure DR scenario.

## Replicated machines - Linux file system/guest storage

- File systems: ext3, ext4, XFS, BTRFS
- Volume manager: LVM2

**ⓘ Note**

Multipath software isn't supported.

## Replicated machines - compute settings

[\[+\] Expand table](#)

Setting	Support	Details
Size	Any Azure VM size with at least two CPU cores and 1-GB RAM	Verify <a href="#">Azure virtual machine sizes</a> .

Setting	Support	Details
RAM	Azure Site Recovery driver consumes 6% of RAM.	
Availability sets	Supported	If you enable replication for an Azure VM with the default options, an availability set is created automatically, based on the source region settings. You can modify these settings.
Availability zones	Supported	
Dedicated Hosts	Not supported	
Hybrid Use Benefit (HUB)	Supported	If the source VM has a HUB license enabled, a test failover or failed over VM also uses the HUB license.
Virtual Machine Scale Set Flex	Availability scenario - supported. Scalability scenario - not supported.	
Azure gallery images - Microsoft published	Supported	Supported if the VM runs on a supported operating system.
Azure Gallery images - Third party published	Supported	Supported if the VM runs on a supported operating system.
Custom images - Third party published	Supported	Supported if the VM runs on a supported operating system.
VMs migrated using Site Recovery	Supported	If a VMware VM or physical machine was migrated to Azure using Site Recovery, you need to uninstall the older version of Mobility service running on the machine, and restart the machine before replicating it to another Azure region.
Azure RBAC policies	Not supported	Azure role-based access control (Azure RBAC) policies on VMs aren't replicated to the failover VM in target region.
Extensions	Not supported	Extensions aren't replicated to the failover VM in target region. It needs to be installed manually after failover.

Setting	Support	Details
Proximity Placement Groups	Supported	Virtual machines located inside a Proximity Placement Group can be protected using Site Recovery.
Tags	Supported	User-generated tags applied on source virtual machines are carried over to target virtual machines post-test failover or failover. Tags on the VM(s) are replicated once every 24 hours for as long as the VM(s) is/are present in the target region.

## Replicated machines - disk actions

Expand table

Action	Details
Resize disk on replicated VM	<p>Resizing up on the source VM is supported. Resizing down on the source VM isn't supported. Resizing should be performed before failover. No need to disable/re-enable replication.</p> <p>If you change the source VM after failover, the changes aren't captured.</p> <p>If you change the disk size on the Azure VM after failover, changes aren't captured by Site Recovery, and fallback will be to the original VM size.</p> <p>If resizing to <math>&gt;= 4</math> TB, note Azure guidance on disk caching <a href="#">here</a>.</p>
Add a disk to a replicated VM	Supported
Offline changes to protected disks	Disconnecting disks and making offline modifications to them require triggering a full resync.
Disk caching	Disk Caching isn't supported for disks 4 TiB and larger. If multiple disks are attached to your VM, each disk that is smaller than 4 TiB will support caching. Changing the cache setting of an Azure disk detaches and re-attaches the target disk. If it's the operating system disk, the VM is restarted. Stop all applications/services that might be affected by this disruption before changing the disk cache setting. Not following those recommendations could lead to data corruption.

# Replicated machines - storage

This table summarized support for the Azure VM OS disk, data disk, and temporary disk.

- It's important to observe the VM disk limits and targets for [managed disks](#) to avoid any performance issues.
- If you deploy with the default settings, Site Recovery automatically creates disks and storage accounts based on the source settings.
- If you customize, ensure you follow the guidelines.

[\[+\] Expand table](#)

Component	Support	Details
Disk renaming	Supported	
OS disk maximum size	4096 GB	<a href="#">Learn more</a> about VM disks.
Temporary disk	Not supported	The temporary disk is always excluded from replication.  Don't store any persistent data on the temporary disk. <a href="#">Learn more</a> .
Data disk maximum size	32 TB for managed disks  4 TB for unmanaged disks	
Data disk minimum size	No restriction for unmanaged disks. 1 GB for managed disks	
Data disk maximum number	Up to 64, in accordance with support for a specific Azure VM size	<a href="#">Learn more</a> about VM sizes.
Data disk maximum size per storage account (for unmanaged disks)	35 TB	This is an upper limit for cumulative size of page blobs created in a premium Storage Account
Data disk change rate	Maximum of 20 MBps per disk for premium storage. Maximum of 2 MBps per disk for Standard storage.	If the average data change rate on the disk is continuously higher than the maximum, replication won't catch up.  However, if the maximum is exceeded

Component	Support	Details
		sporadically, replication can catch up, but you might see slightly delayed recovery points.
Data disk - standard storage account	Supported	
Data disk - premium storage account	Supported	If a VM has disks spread across premium and standard storage accounts, you can select a different target storage account for each disk, to ensure you have the same storage configuration in the target region.
Managed disk - standard	Supported in Azure regions in which Azure Site Recovery is supported.	
Managed disk - premium	Supported in Azure regions in which Azure Site Recovery is supported.	
Disk subscription limits	Up to 3000 protected disks per Subscription	Ensure that the Source or Target subscription doesn't have more than 3000 Azure Site Recovery-protected Disks (Both Data and OS).
Standard SSD	Supported	
Redundancy	LRS, ZRS, and GRS are supported.	
Cool and hot storage	Not supported	VM disks aren't supported on cool and hot storage
Storage Spaces	Supported	
NVMe storage interface	Not supported	
Encryption at host	Not Supported	The VM will get protected, but the failed over VM won't have Encryption at host enabled. <a href="#">See detailed information</a> to create a VM with end-to-end encryption using Encryption at host.
Encryption at rest (SSE)	Supported	SSE is the default setting on storage accounts.

Component	Support	Details
Encryption at rest (CMK)	Supported	Both Software and HSM keys are supported for managed disks
Double Encryption at rest	Supported	Learn more on supported regions for <a href="#">Windows</a> and <a href="#">Linux</a>
FIPS encryption	Not supported	
Azure Disk Encryption (ADE) for Windows OS	Supported for VMs with managed disks.	VMs using unmanaged disks aren't supported.  HSM-protected keys aren't supported.  Encryption of individual volumes on a single disk isn't supported.
Azure Disk Encryption (ADE) for Linux OS	Supported for VMs with managed disks.	VMs using unmanaged disks aren't supported.  HSM-protected keys aren't supported.  Encryption of individual volumes on a single disk isn't supported.  Known issue with enabling replication. <a href="#">Learn more</a> .
SAS key rotation	Not Supported	If the SAS key for storage accounts is rotated, customer needs to disable and re-enable replication.
Host Caching	Supported	
Hot add	Supported	Enabling replication for a data disk that you add to a replicated Azure VM is supported for VMs that use managed disks.  Only one disk can be hot added to an Azure VM at a time. Parallel addition of multiple disks isn't supported.
Hot remove disk	Not supported	If you remove data disk on the VM, you need to disable replication and enable replication again for the VM.
Exclude disk	Support. You must use <a href="#">PowerShell</a> to configure.	Temporary disks are excluded by default.

Component	Support	Details
Storage Spaces Direct	Supported for crash consistent recovery points. Application consistent recovery points aren't supported.	
Scale-out File Server	Supported for crash consistent recovery points. Application consistent recovery points aren't supported.	
DRBD	Disks that are part of a DRBD setup aren't supported.	
LRS	Supported	
GRS	Supported	
RA-GRS	Supported	
ZRS	Supported	
Cool and Hot Storage	Not supported	Virtual machine disks aren't supported on cool and hot storage
Azure Storage firewalls for virtual networks	Supported	If you want to restrict virtual network access to storage accounts, enable <a href="#">Allow trusted Microsoft services</a> .
General purpose V2 storage accounts (Both Hot and Cool tier)	Supported	Transaction costs increase substantially compared to General purpose V1 storage accounts
Generation 2 (UEFI boot)	Supported	
NVMe disks	Not supported	
Azure Shared Disks	Not supported	
Ultra Disks	Not supported	
Secure transfer option	Supported	
Write accelerator enabled disks	Not supported	

Component	Support	Details
Tags	Supported	User-generated tags are replicated every 24 hours.
Soft delete	Not supported	Soft delete isn't supported because once it's enabled on a storage account, it increases cost. Azure Site Recovery performs very frequent creates/deletes of log files while replicating causing costs to increase.
iSCSI disks	Not supported	Azure Site Recovery may be used to migrate or failover iSCSI disks into Azure. However, iSCSI disks aren't supported for Azure to Azure replication and failover/failback.

### ⓘ Important

To avoid performance issues, make sure that you follow VM disk scalability and performance targets for **managed disks**. If you use default settings, Site Recovery creates the required disks and storage accounts, based on the source configuration. If you customize and select your own settings, follow the disk scalability and performance targets for your source VMs.

## Limits and data change rates

The following table summarizes Site Recovery limits.

- These limits are based on our tests, but obviously don't cover all possible application I/O combinations.
- Actual results can vary based on your app I/O mix.
- There are two limits to consider, per disk data churn and per virtual machine data churn.
- The current limit for per virtual machine data churn is 54 MB/s, regardless of size.

ⓘ [Expand table](#)

Storage target	Average source disk I/O	Average source disk data churn	Total source disk data churn per day
Standard storage	8 KB	2 MB/s	168 GB per disk
Premium P10 or P15 disk	8 KB	2 MB/s	168 GB per disk

Storage target	Average source disk I/O	Average source disk data churn	Total source disk data churn per day
Premium P10 or P15 disk	16 KB	4 MB/s	336 GB per disk
Premium P10 or P15 disk	32 KB or greater	8 MB/s	672 GB per disk
Premium P20 or P30 or P40 or P50 disk	8 KB	5 MB/s	421 GB per disk
Premium P20 or P30 or P40 or P50 disk	16 KB or greater	20 MB/s	1684 GB per disk

 **Note**

High churn support is now available in Azure Site Recovery where churn limit per virtual machine has increased up to 100 MB/s. For more information, see [Azure VM Disaster Recovery - High Churn Support](#).

## Replicated machines - networking

 [Expand table](#)

Setting	Support	Details
NIC	Maximum number supported for a specific Azure VM size	<p>NICs are created when the VM is created during failover.</p> <p>The number of NICs on the failover VM depends on the number of NICs on the source VM when replication was enabled. If you add or remove a NIC after enabling replication, it doesn't impact the number of NICs on the replicated VM after failover.</p> <p>The order of NICs after failover isn't guaranteed to be the same as the original order.</p> <p>You can rename NICs in the target region based on your organization's naming conventions.</p>
Internet Load Balancer	Not supported	You can set up public/internet load balancers in the primary region. However, public/internet load balancers aren't supported by Azure Site Recovery in the DR region.

Setting	Support	Details
Internal Load balancer	Supported	Associate the preconfigured load balancer using an Azure Automation script in a recovery plan.
Public IP address	Supported	Associate an existing public IP address with the NIC. Or, create a public IP address and associate it with the NIC using an Azure Automation script in a recovery plan.
NSG on NIC	Supported	Associate the NSG with the NIC using an Azure Automation script in a recovery plan.
NSG on subnet	Supported	Associate the NSG with the subnet using an Azure Automation script in a recovery plan.
Reserved (static) IP address	Supported	<p>If the NIC on the source VM has a static IP address, and the target subnet has the same IP address available, it's assigned to the failed over VM.</p> <p>If the target subnet doesn't have the same IP address available, one of the available IP addresses in the subnet is reserved for the VM.</p> <p>You can also specify a fixed IP address and subnet in <b>Replicated items &gt; Settings &gt; Network &gt; Network interfaces</b>.</p>
Dynamic IP address	Supported	<p>If the NIC on the source has dynamic IP addressing, the NIC on the failed over VM is also dynamic by default.</p> <p>You can modify this to a fixed IP address if required.</p>
Multiple IP addresses	Supported	When you fail over a VM that has a NIC with multiple IP addresses, only the primary IP address of the NIC in the source region is kept by default. To failover Secondary IP Configurations, go to the <b>Network</b> blade and configure them.
Traffic Manager	Supported	You can preconfigure Traffic Manager so that traffic is routed to the endpoint in the source region on a regular basis, and to the endpoint in the target region in case of failover.
Azure DNS	Supported	
Custom DNS	Supported	
Unauthenticated proxy	Supported	<a href="#">Learn more</a>

Setting	Support	Details
Authenticated Proxy	Not supported	If the VM is using an authenticated proxy for outbound connectivity, it can't be replicated using Azure Site Recovery.
VPN site-to-site connection to on-premises (with or without ExpressRoute)	Supported	Ensure that the UDRs and NSGs are configured in such a way that the Site Recovery traffic isn't routed to on-premises. <a href="#">Learn more</a>
VNET to VNET connection	Supported	<a href="#">Learn more</a>
Virtual Network Service Endpoints	Supported	If you are restricting the virtual network access to storage accounts, ensure that the trusted Microsoft services are allowed access to the storage account.
Accelerated networking	Supported	Accelerated networking can be enabled on the recovery VM only if it is enabled on the source VM also. <a href="#">Learn more</a> .
Palo Alto Network Appliance	Not supported	With third-party appliances, there are often restrictions imposed by the provider inside the Virtual Machine. Azure Site Recovery needs agent, extensions, and outbound connectivity to be available. But the appliance doesn't let any outbound activity to be configured inside the Virtual Machine.
IPv6	Not supported	Mixed configurations that include both IPv4 and IPv6 are supported. However, Azure Site Recovery will use any free IPv4 address available, if there are no free IPv4 addresses in the subnet, then the configuration is not supported.
Private link access to Site Recovery service	Supported	<a href="#">Learn more</a>
Tags	Supported	User-generated tags on NICs are replicated every 24 hours.

## Next steps

- Read [networking guidance](#) for replicating Azure VMs.
- Deploy disaster recovery by [replicating Azure VMs](#).

# Integrate ExpressRoute with disaster recovery for Azure VMs

Article • 12/14/2023

This article describes how to integrate Azure ExpressRoute with [Azure Site Recovery](#), when you set up disaster recovery for Azure VMs to a secondary Azure region.

Site Recovery enables disaster recovery of Azure VMs by replicating Azure VM data to Azure.

- If Azure VMs use [Azure managed disks](#), VM data is replicated to a replicated managed disk in the secondary region.
- If Azure VMs don't use managed disks, VM data is replicated to an Azure storage account.
- Replication endpoints are public, but replication traffic for Azure VMs doesn't cross the internet.

ExpressRoute enables you to extend on-premises networks into the Microsoft Azure cloud over a private connection, facilitated by a connectivity provider. If you have ExpressRoute configured, it integrates with Site Recovery as follows:

- **During replication between Azure regions:** Replication traffic for Azure VM disaster recovery is within Azure only, and ExpressRoute isn't needed or used for replication. However, if you're connecting from an on-premises site to the Azure VMs in the primary Azure site, there are many issues to be aware of when you're setting up disaster recovery for those Azure VMs.
- **Failover between Azure regions:** When outages occur, you fail over Azure VMs from the primary to secondary Azure region. After failing over to a secondary region, there are many steps to take in order to access the Azure VMs in the secondary region using ExpressRoute.

## Before you begin

Before you begin, make sure you understand the following concepts:

- ExpressRoute [circuits](#)
- ExpressRoute [routing domains](#)
- ExpressRoute [locations](#).
- Azure VM [replication architecture](#)
- How to [set up replication](#) for Azure VMs.

- How to [fail over](#) Azure VMs.

## General recommendations

For best practice, and to ensure efficient Recovery Time Objectives (RTOs) for disaster recovery, we recommend you do the following when you set up Site Recovery to integrate with ExpressRoute:

- Provision networking components before failover to a secondary region:
  - When you enable replication for Azure VMs, Site Recovery can automatically deploy networking resources such as networks, subnets, and gateways in the target Azure region, based on source network settings.
  - Site Recovery can't automatically set up networking resources such as VNet gateways.
  - We recommend you provision these additional networking resources before failover. A small downtime is associated with this deployment, and it can impact the overall recovery time, if you didn't account for it during deployment planning.
- Run regular disaster recovery drills:
  - A drill validates your replication strategy without data loss or downtime, and doesn't affect your production environment. It helps avoid last-minute configuration issues that can adversely impact RTO.
  - When you run a test failover for the drill, we recommend that you use a separate Azure VM network, instead of the default network that's set up when you enable replication.
- Use different IP address spaces if you have a single ExpressRoute circuit.
  - We recommend that you use a different IP address space for the target virtual network. This avoids issues when establishing connections during regional outages.
  - If you can't use a separate address space, be sure to run the disaster recovery drill test failover on a separate test network with different IP addresses. You can't connect two VNets with overlapping IP address space to the same ExpressRoute circuit.

## Replicate Azure VMs when using ExpressRoute

If you want to set up replication for Azure VMs in a primary site, and you're connecting to these VMs from your on-premises site over ExpressRoute, here's what you need to do:

1. [Enable replication](#) for each Azure VM.

## 2. Optionally let Site Recovery set up networking:

- When you configure and enable replication, Site Recovery sets up networks, subnets, and gateway subnets in the target Azure region, to match those in the source region. Site Recovery also maps between the source and target virtual networks.
- If you don't want Site Recovery to do this automatically, create the target-side network resources before you enable replication.

## 3. Create other networking elements:

- Site Recovery doesn't create route tables, VNet gateways, VNet gateway connections, VNet peering, or other networking resources and connections in the secondary region.
- You need to create these additional networking elements in the secondary region, anytime before running a failover from the primary region.
- You can use [recovery plans](#) and automation scripts to set up and connect these networking resources.

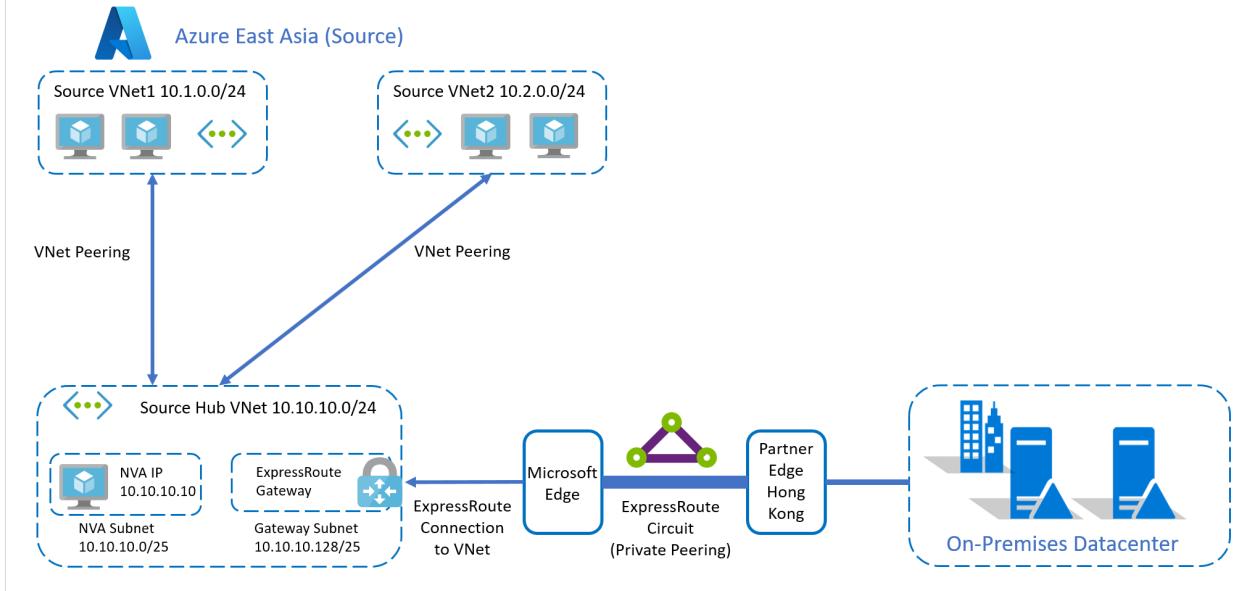
## 4. If you have a network virtual appliance (NVA) deployed to control the flow of network traffic:

- Azure's default system route for Azure VM replication is 0.0.0.0/0.
- Typically, NVA deployments also define a default route (0.0.0.0/0) that forces outbound Internet traffic to flow through the NVA. The default route is used when no other specific route configuration can be found.
- If so, the NVA might be overloaded if all replication traffic passes through the NVA.
- The same limitation also applies when using default routes for routing all Azure VM traffic to on-premises deployments.
- In this scenario, we recommend that you [create a network service endpoint](#) in your virtual network for the Microsoft.Storage service, so that the replication traffic doesn't leave Azure boundary.

# Replication example

Typically enterprise deployments have workloads split across multiple Azure VNets, with a central connectivity hub for external connectivity to the internet and to on-premises sites. A hub and spoke topology is typically used together with ExpressRoute.

## On-premises to Azure connectivity – Before failover



- **Region.** Apps are deployed in the Azure East Asia region.
- **Spoke vNets.** Apps are deployed in two spoke vNets:
  - **Source vNet1:** 10.1.0.0/24.
  - **Source vNet2:** 10.2.0.0/24.
  - Each spoke virtual network is connected to **Hub vNet**.
- **Hub vNet.** There's a hub vNet **Source Hub vNet:** 10.10.10.0/24.
  - This hub vNet acts as the gatekeeper.
  - All communications across subnets go through this hub.
    - **Hub vNet subnets.** The hub vNet has two subnets:
      - **NVA subnet:** 10.10.10.0/25. This subnet contains an NVA (10.10.10.10).
      - **Gateway subnet:** 10.10.10.128/25. This subnet contains an ExpressRoute gateway connected to an ExpressRoute connection that routes to the on-premises site via a private peering routing domain.
- The on-premises datacenter has an ExpressRoute circuit connection through a partner edge in Hong Kong Special Administrative Region.
- All routing is controlled through Azure route tables (UDR).
- All outbound traffic between vNets, or to the on-premises datacenter is routed through the NVA.

## Hub and spoke peering settings

### Spoke to hub

[\[ \]](#) Expand table

Direction	Setting	State
Spoke to hub	Allow virtual network address	Enabled
Spoke to hub	Allow forwarded traffic	Enabled
Spoke to hub	Allow gateway transit	Disabled
Spoke to hub	Use remote gateways	Enabled

### Configuration

Allow virtual network access i

Allow forwarded traffic i

Allow gateway transit i

Use remote gateways i

[Expand table](#)

## Hub to spoke

Direction	Setting	State
Hub to spoke	Allow virtual network address	Enabled
Hub to spoke	Allow forwarded traffic	Enabled
Hub to spoke	Allow gateway transit	Enabled
Hub to spoke	Use remove gateways	Disabled

### Configuration

Allow virtual network access i

Allow forwarded traffic i

Allow gateway transit i

Use remote gateways i

## Example steps

In our example, the following should happen when enabling replication for Azure VMs in the source network:

1. You [enable replication](#) for a VM.
2. Site Recovery creates replica vNets, subnets, and gateway subnets in the target region.
3. Site Recovery creates mappings between the source networks and the replica target networks it creates.
4. You manually create virtual network gateways, virtual network gateway connections, virtual network peering, or any other networking resources or connections.

## Fail over Azure VMs when using ExpressRoute

After you fail Azure VMs over to the target Azure region using Site Recovery, you can access them using ExpressRoute [private peering](#).

- You need to connect ExpressRoute to the target vNet with a new connection. The existing ExpressRoute connection isn't automatically transferred.
- The way in which you set up your ExpressRoute connection to the target vNet depends on your ExpressRoute topology.

## Access with two circuits

### Two circuits with two peering locations

This configuration helps protect ExpressRoute circuits against regional disaster. If your primary peering location goes down, connections can continue from the other location.

- The circuit connected to the production environment is usually the primary. The secondary circuit typically has lower bandwidth, which can be increased if a disaster occurs.
- After failover, you can establish connections from the secondary ExpressRoute circuit to the target vNet. Alternatively, you can have connections set up and ready in case of disaster, to reduce overall recovery time.
- With simultaneous connections to both primary and target vNets, make sure that your on-premises routing only uses the secondary circuit and connection after failover.
- The source and target vNets can receive new IP addresses, or keep the same ones, after failover. In both cases, the secondary connections can be established prior to failover.

## Two circuits with single peering location

This configuration helps protect against failure of the primary ExpressRoute circuit, but not if the single ExpressRoute peering location goes down, impacting both circuits.

- You can have simultaneous connections from the on-premises datacenter to source vNet with the primary circuit, and to the target vNet with the secondary circuit.
- With simultaneous connections to primary and target, make sure that on-premises routing only uses the secondary circuit and connection after failover.
- You can't connect both circuits to the same vNet when circuits are created at the same peering location.

## Access with a single circuit

In this configuration there's only one Expressroute circuit. Although the circuit has a redundant connection in case one goes down, a single route circuit won't provide resilience if your peering region goes down. Note that:

- If the target Azure region isn't in the same location as the source, you need to enable ExpressRoute Premium if you're using a single ExpressRoute circuit. Learn about [ExpressRoute locations](#) and [ExpressRoute pricing](#).
- You can't connect source and target vNets simultaneously to the circuit if the same IP address space is used on the target region. In this scenario:
  - Disconnect the source side connection, and then establish the target side connection. This connection change can be scripted as part of a Site Recovery recovery plan. Note that:
    - In a regional failure, if the primary region is inaccessible, the disconnect operation could fail. This could impact connection creation to the target region.
    - If you created the connection in the target region, and primary region recovers later, you might experience packet drops if two simultaneous connections attempt to connect to the same address space.
    - To prevent this, terminate the primary connection immediately.
    - After VM failback to the primary region, the primary connection can again be established, after you disconnect the secondary connection.
- If a different address space is used on the target vNet, you can simultaneously connect to the source and target vNets from the same ExpressRoute circuit.

## Failover example

In our example, we're using the following topology:

- Two different ExpressRoute circuits in two different peering locations.
- Retain private IP addresses for the Azure VMs after failover.
- The target recovery region is Azure SouthEast Asia.
- A secondary ExpressRoute circuit connection is established through a partner edge in Singapore.

For a simple topology that uses a single ExpressRoute circuit, with same IP address after failover, [review this article](#).

## Example steps

To automate recovery in this example, here's what you need to do:

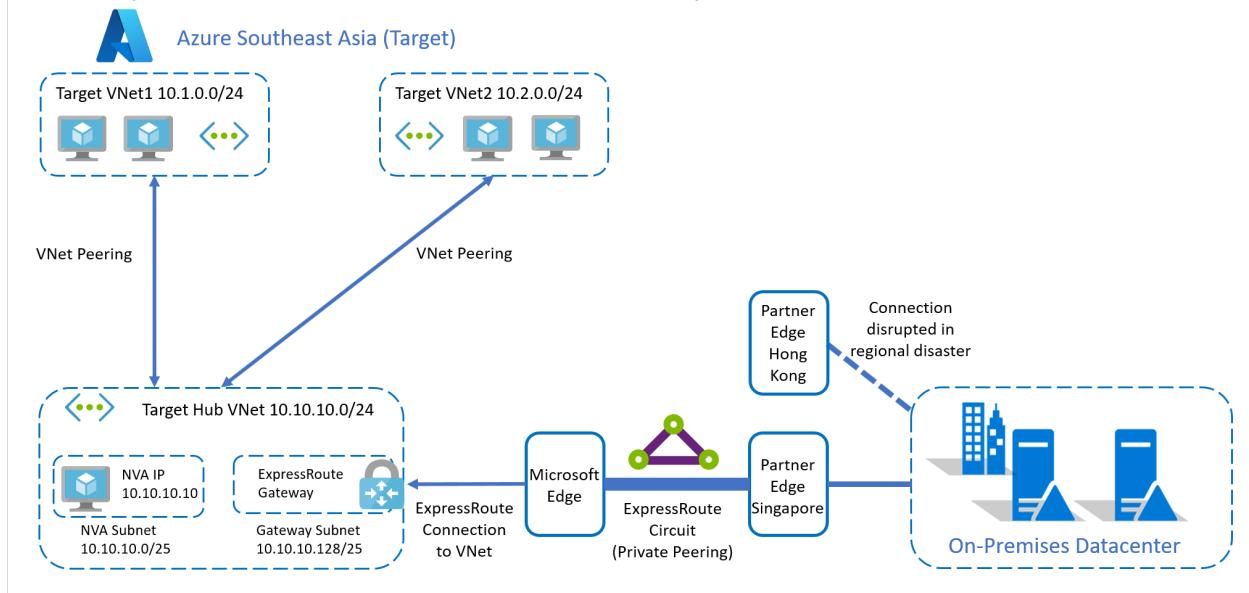
1. Follow the steps to set up replication.
2. [Fail over the Azure VMs](#), with these additional steps during or after the failover.
  - a. Create the Azure ExpressRoute Gateway in the target region hub VNet. This is need to connect the target hub vNet to the ExpressRoute circuit.
  - b. Create the connection from the target hub vNet to the target ExpressRoute circuit.
  - c. Set up the VNet peerings between the target region's hub and spoke virtual networks. The peering properties on the target region are the same as those on the source region.
  - d. Set up the UDRs in the hub VNet, and the two spoke VNets.
    - The properties of the target side UDRs are the same as those on the source side when using the same IP addresses.
    - With different target IP addresses, the UDRs should be modified accordingly.

The above steps can be scripted as part of a [recovery plan](#). Depending on the application connectivity and recovery time requirements, the above steps can also be completed prior to starting the failover.

## After recovery

After recovering the VMs and completing connectivity, the recovery environment is as follows.

## On-premises to Azure connectivity – After failover



## Next steps

Learn more about using [recovery plans](#) to automate app failover.

# Recover from a region-wide service disruption

Article • 03/20/2023

Azure is divided physically and logically into units called regions. A region consists of one or more data centers in close proximity. Many regions and services also support [availability zones](#), which can be used to provide more resiliency against outages in a single data center. Consider using regions with availability zones to improve the availability of your solution.

Under rare circumstances, it is possible that facilities in an entire availability zone or region can become inaccessible, for example, due to network failures. Or, facilities can be lost entirely, for example, due to a natural disaster. Azure has capabilities for creating applications that are distributed across zones and regions. Such distribution helps to minimize the possibility that a failure in one zone or region could affect other zones or regions.

## Cloud services

### Resource management

You can distribute compute instances across regions by creating a separate cloud service in each target region, and then publishing the deployment package to each cloud service. However, distributing traffic across cloud services in different regions must be implemented by the application developer or with a traffic management service.

Determining the number of spare role instances to deploy in advance for disaster recovery is an important aspect of capacity planning. Having a full-scale secondary deployment ensures that capacity is already available when needed; however, this effectively doubles the cost. A common pattern is to have a small, secondary deployment, just large enough to run critical services. This small secondary deployment is a good idea, both to reserve capacity, and for testing the configuration of the secondary environment.

#### Note

The subscription quota is not a capacity guarantee. The quota is simply a credit limit. To guarantee capacity, the required number of roles must be defined in the

service model, and the roles must be deployed.

## Load Balancing

To load balance traffic across regions requires a traffic management solution. Azure provides [Azure Traffic Manager](#). You can also take advantage of third-party services that provide similar traffic management capabilities.

## Strategies

Many alternative strategies are available for implementing distributed compute across regions. These must be tailored to the specific business requirements and circumstances of the application. At a high level, the approaches can be divided into the following categories:

- **Redeploy on disaster:** In this approach, the application is redeployed from scratch at the time of disaster. This is appropriate for non-critical applications that don't require a guaranteed recovery time.
- **Warm Spare (Active/Passive):** A secondary hosted service is created in an alternate region, and roles are deployed to guarantee minimal capacity; however, the roles don't receive production traffic. This approach is useful for applications that have not been designed to distribute traffic across regions.
- **Hot Spare (Active/Active):** The application is designed to receive production load in multiple regions. The cloud services in each region might be configured for higher capacity than required for disaster recovery purposes. Alternatively, the cloud services might scale out as necessary at the time of a disaster and fail over. This approach requires substantial investment in application design, but it has significant benefits. These include low and guaranteed recovery time, continuous testing of all recovery locations, and efficient usage of capacity.

A complete discussion of distributed design is outside the scope of this document. For more information, see [Disaster Recovery and High Availability for Azure Applications](#).

## Virtual machines

Recovery of infrastructure as a service (IaaS) virtual machines (VMs) is similar to platform as a service (PaaS) compute recovery in many respects. There are important differences, however, due to the fact that an IaaS VM consists of both the VM and the VM disk.

- **Use Azure Backup to create cross region backups that are application consistent.** [Azure Backup](#) enables customers to create application consistent backups across multiple VM disks, and support replication of backups across regions. You can do this by choosing to geo-replicate the backup vault at the time of creation. Replication of the backup vault must be configured at the time of creation. It can't be set later. If a region is lost, Microsoft will make the backups available to customers. Customers will be able to restore to any of their configured restore points.
- **Separate the data disk from the operating system disk.** An important consideration for IaaS VMs is that you cannot change the operating system disk without re-creating the VM. This is not a problem if your recovery strategy is to redeploy after disaster. However, it might be a problem if you are using the Warm Spare approach to reserve capacity. To implement this properly, you must have the correct operating system disk deployed to both the primary and secondary locations, and the application data must be stored on a separate drive. If possible, use a standard operating system configuration that can be provided on both locations. After a failover, you must then attach the data drive to your existing IaaS VMs in the secondary DC. Use AzCopy to copy snapshots of the data disk(s) to a remote site.
- **Be aware of potential consistency issues after a geo-failover of multiple VM Disks.** VM Disks are implemented as Azure Storage blobs, and have the same geo-replication characteristic. Unless [Azure Backup](#) is used, there are no guarantees of consistency across disks, because geo-replication is asynchronous and replicates independently. Individual VM disks are guaranteed to be in a crash consistent state after a geo-failover, but not consistent across disks. This could cause problems in some cases (for example, in the case of disk striping).
- **Use Azure Site Recovery to replicate applications across regions.** With [Azure Site Recovery](#), customers don't need to worry about separating data disks from operating system disks or about potential consistency issues. Azure Site Recovery replicates workloads running on physical and virtual machines from a primary site (either on-premises or in Azure) to a secondary location (in Azure). When an outage occurs at the customer's primary site, a failover can be triggered to quickly return the customer to an operational state. After the primary location is restored, customers can then fail back. They can easily replicate using recovery points with application-consistent snapshots. These snapshots capture disk data, all in-memory data, and all in-process transactions. Azure Site Recovery allows customers to keep recovery time objectives (RTO) and recovery point objectives (RPO) within organizational limits. Customers can also easily run disaster recovery drills without affecting applications in production. Using recovery plans, customers

can sequence the failover and recovery of multitier applications running on multiple VMs. They can group machines together in a recovery plan, and optionally add scripts and manual actions. Recovery plans can be integrated with Azure Automation runbooks.

## Storage

### Recovery by using geo-redundant storage of blob, table, queue, and VM disk storage

In Azure, blobs, tables, queues, and VM disks are all geo-replicated by default. This is referred to as geo-redundant storage (GRS). GRS replicates storage data to a paired datacenter located hundreds of miles apart within a specific geographic region. GRS is designed to provide additional durability in case there is a major datacenter disaster. Microsoft controls when failover occurs, and failover is limited to major disasters in which the original primary location is deemed unrecoverable in a reasonable amount of time. Under some scenarios, this can be several days. Data is typically replicated within a few minutes, although synchronization interval is not yet covered by a service level agreement.

If a geo-failover occurs, there will be no change to how the account is accessed (the URL and account key will not change). The storage account will, however, be in a different region after failover. This could impact applications that require regional affinity with their storage account. Even for services and applications that do not require a storage account in the same datacenter, the cross-datacenter latency and bandwidth charges might be compelling reasons to move traffic to the failover region temporarily. This could factor into an overall disaster recovery strategy.

In addition to automatic failover provided by GRS, Azure has introduced a service that gives you read access to the copy of your data in the secondary storage location. This is called read-access geo-redundant storage (RA-GRS).

For more information about both GRS and RA-GRS storage, see [Azure Storage replication](#).

### Geo-replication region mappings

It is important to know where your data is geo-replicated, in order to know where to deploy the other instances of your data that require regional affinity with your storage. For more information, see [Azure Paired Regions](#).

## Geo-replication pricing

Geo-replication is included in current pricing for Azure Storage. This is called geo-redundant storage (GRS). If you do not want your data geo-replicated, you can disable geo-replication for your account. This is called locally redundant storage (LRS), and it is charged at a discounted price compared to GRS.

## Determining if a geo-failover has occurred

If a geo-failover occurs, this will be posted to the [Azure Service Health Dashboard](#). Applications can implement an automated means of detecting this, however, by monitoring the geo-region for their storage account. This can be used to trigger other recovery operations, such as activation of compute resources in the geo-region where their storage moved to. You can perform a query for this from the service management API, by using [Get Storage Account Properties](#). The relevant properties are:

Console

```
<GeoPrimaryRegion>primary-region</GeoPrimaryRegion>
<StatusOfPrimary>[Available|Unavailable]</StatusOfPrimary>
<LastGeoFailoverTime>DateTime</LastGeoFailoverTime>
<GeoSecondaryRegion>secondary-region</GeoSecondaryRegion>
<StatusOfSecondary>[Available|Unavailable]</StatusOfSecondary>
```

## Database

### SQL Database

Azure SQL Database provides two types of recovery: geo-restore and active geo-replication.

#### Geo-restore

[Geo-restore](#) is also available with Basic, Standard, and Premium databases. It provides the default recovery option when the database is unavailable because of an incident in the region where your database is hosted. Similar to point-in-time restore, geo-restore relies on database backups in geo-redundant Azure storage. It restores from the geo-replicated backup copy, and therefore is resilient to the storage outages in the primary region. For more information, see [Restore an Azure SQL Database or failover to a secondary](#).

## Active geo-replication

[Active geo-replication](#) is available for all database tiers. It's designed for applications that have more aggressive recovery requirements than geo-restore can offer. Using active geo-replication, you can create up to four readable secondaries on servers in different regions. You can initiate failover to any of the secondaries. In addition, active geo-replication can be used to support the application upgrade or relocation scenarios, as well as load balancing for read-only workloads. For details, see [Configure active geo-replication for Azure SQL Database and initiate failover](#). Refer to [Designing globally available services using Azure SQL Database](#) and [Managing rolling upgrades of cloud applications by using SQL Database active geo-replication](#) for details on how to design and implement applications and applications upgrade without downtime.

## SQL Server on Azure Virtual Machines

A variety of options are available for recovery and high availability for SQL Server 2012 (and later) running in Azure Virtual Machines. For more information, see [High availability and disaster recovery for SQL Server in Azure Virtual Machines](#).

## Other Azure platform services

When attempting to run your cloud service in multiple Azure regions, you must consider the implications for each of your dependencies. In the following sections, the service-specific guidance assumes that you must use the same Azure service in an alternate Azure datacenter. This involves both configuration and data-replication tasks.

### Note

In some cases, these steps can help to mitigate a service-specific outage rather than an entire datacenter event. From the application perspective, a service-specific outage might be just as limiting and would require temporarily migrating the service to an alternate Azure region.

## Service Bus

Azure Service Bus uses a unique namespace that does not span Azure regions. So the first requirement is to set up the necessary service bus namespaces in the alternate region. However, there are also considerations for the durability of the queued messages. There are several strategies for replicating messages across Azure regions.

For the details on these replication strategies and other disaster recovery strategies, see [Best practices for insulating applications against Service Bus outages and disasters](#).

## App Service

To migrate an Azure App Service application, such as Web Apps or Mobile Apps, to a secondary Azure region, you must have a backup of the website available for publishing. If the outage does not involve the entire Azure datacenter, it might be possible to use FTP to download a recent backup of the site content. Then create a new app in the alternate region, unless you have previously done this to reserve capacity. Publish the site to the new region, and make any necessary configuration changes. These changes could include database connection strings or other region-specific settings. If necessary, add the site's SSL certificate and change the DNS CNAME record so that the custom domain name points to the redeployed Azure Web App URL.

## HDInsight

The data associated with HDInsight is stored by default in Azure Blob Storage. HDInsight requires that a Hadoop cluster processing MapReduce jobs must be colocated in the same region as the storage account that contains the data being analyzed. Provided you use the geo-replication feature available to Azure Storage, you can access your data in the secondary region where the data was replicated if for some reason the primary region is no longer available. You can create a new Hadoop cluster in the region where the data has been replicated and continue processing it.

## SQL Reporting

At this time, recovering from the loss of an Azure region requires multiple SQL Reporting instances in different Azure regions. These SQL Reporting instances should access the same data, and that data should have its own recovery plan in the event of a disaster. You can also maintain external backup copies of the RDL file for each report.

## Media Services

Azure Media Services has a different recovery approach for encoding and streaming. Typically, streaming is more critical during a regional outage. To prepare for this, you should have a Media Services account in two different Azure regions. The encoded content should be located in both regions. During a failure, you can redirect the streaming traffic to the alternate region. Encoding can be performed in any Azure

region. If encoding is time-sensitive, for example during live event processing, you must be prepared to submit jobs to an alternate datacenter during failures.

## Virtual network

Configuration files provide the quickest way to set up a virtual network in an alternate Azure region. After configuring the virtual network in the primary Azure region, [export the virtual network settings](#) for the current network to a network configuration file. If an outage occurs in the primary region, [restore the virtual network](#) from the stored configuration file. Then configure other cloud services, virtual machines, or cross-premises settings to work with the new virtual network.

There are VNET related resources which we need to take into account (Ex. NSG, DNS, Route Tables). The method described in [Infrastructure as a code](#) is a way to generate the same environment every time, and you can deploy in a new region.

## Checklists for disaster recovery

### Cloud Services checklist

1. Review the Cloud Services section of this document.
2. Create a cross-region disaster recovery strategy.
3. Understand trade-offs in reserving capacity in alternate regions.
4. Use traffic routing tools, such as Azure Traffic Manager.

### Virtual Machines checklist

1. Review the Virtual Machines section of this document.
2. Use [Azure Backup](#) to create application consistent backups across regions.

### Storage checklist

1. Review the Storage section of this document.
2. Do not disable geo-replication of storage resources.
3. Understand alternate region for geo-replication if a failover occurs.
4. Create custom backup strategies for user-controlled failover strategies.

### SQL Database checklist

1. Review the SQL Database section of this document.

2. Use [Geo-restore](#) or [geo-replication](#) as appropriate.

## SQL Server on Virtual Machines checklist

1. Review the SQL Server on Virtual Machines section of this document.
2. Use cross-region AlwaysOn Availability Groups or database mirroring.
3. Alternately use backup and restore to blob storage.

## Service Bus checklist

1. Review the Service Bus section of this document.
2. Configure a Service Bus namespace in an alternate region.
3. Consider custom replication strategies for messages across regions.

## App Service checklist

1. Review the App Service section of this document.
2. Maintain site backups outside of the primary region.
3. If outage is partial, attempt to retrieve current site with FTP.
4. Plan to deploy the site to new or existing site in an alternate region.
5. Plan configuration changes for both application and DNS CNAME records.

## HDInsight checklist

1. Review the HDInsight section of this document.
2. Create a new Hadoop cluster in the region with replicated data.

## SQL Reporting checklist

1. Review the SQL Reporting section of this document.
2. Maintain an alternate SQL Reporting instance in a different region.
3. Maintain a separate plan to replicate the target to that region.

## Media Services checklist

1. Review the Media Services section of this document.
2. Create a Media Services account in an alternate region.
3. Encode the same content in both regions to support streaming failover.
4. Submit encoding jobs to an alternate region if a service disruption occurs.

## **Virtual Network checklist**

1. Review the Virtual Network section of this document.
2. Use exported virtual network settings to re-create it in another region.

# Moving Azure VMs to another Azure region

Article • 12/14/2023

This article provides an overview of the reasons and steps involved in moving Azure VMs to another Azure region using [Azure Site Recovery](#).

## Reasons to move Azure VMs

You might move VMs for the following reasons:

- You already deployed in one region, and a new region support was added which is closer to the end users of your application or service. In this scenario, you'd want to move your VMs as is to the new region to reduce latency. Use the same approach if you want to consolidate subscriptions or if there are governance or organization rules that require you to move.
- Your VM was deployed as a single-instance VM or as part of an availability set. If you want to increase the availability SLAs, you can move your VMs into an Availability Zone.

## Move VMs with Resource Mover

You can now move VMs to another region with [Azure Resource Mover](#). Resource Mover is in public preview and provides:

- A single hub for moving resources across regions.
- Reduced move time and complexity. Everything you need is in a single location.
- A simple and consistent experience for moving different types of Azure resources.
- An easy way to identify dependencies across resources you want to move. This helps you to move related resources together, so that everything works as expected in the target region, after the move.
- Automatic cleanup of resources in the source region, if you want to delete them after the move.
- Testing. You can try out a move, and then discard it if you don't want to do a full move.

## Move VMs with Site Recovery

Moving VMs with Site Recovery involves the following steps:

1. Verify prerequisites.
2. Prepare the source VMs.
3. Prepare the target region.
4. Copy data to the target region. Use Azure Site Recovery replication technology to copy data from the source VM to the target region.
5. Test the configuration. After the replication is complete, test the configuration by performing a test failover to a non-production network.
6. Perform the move.
7. Discard the resources in the source region.

 **Note**

Details about these steps are provided in the following sections.

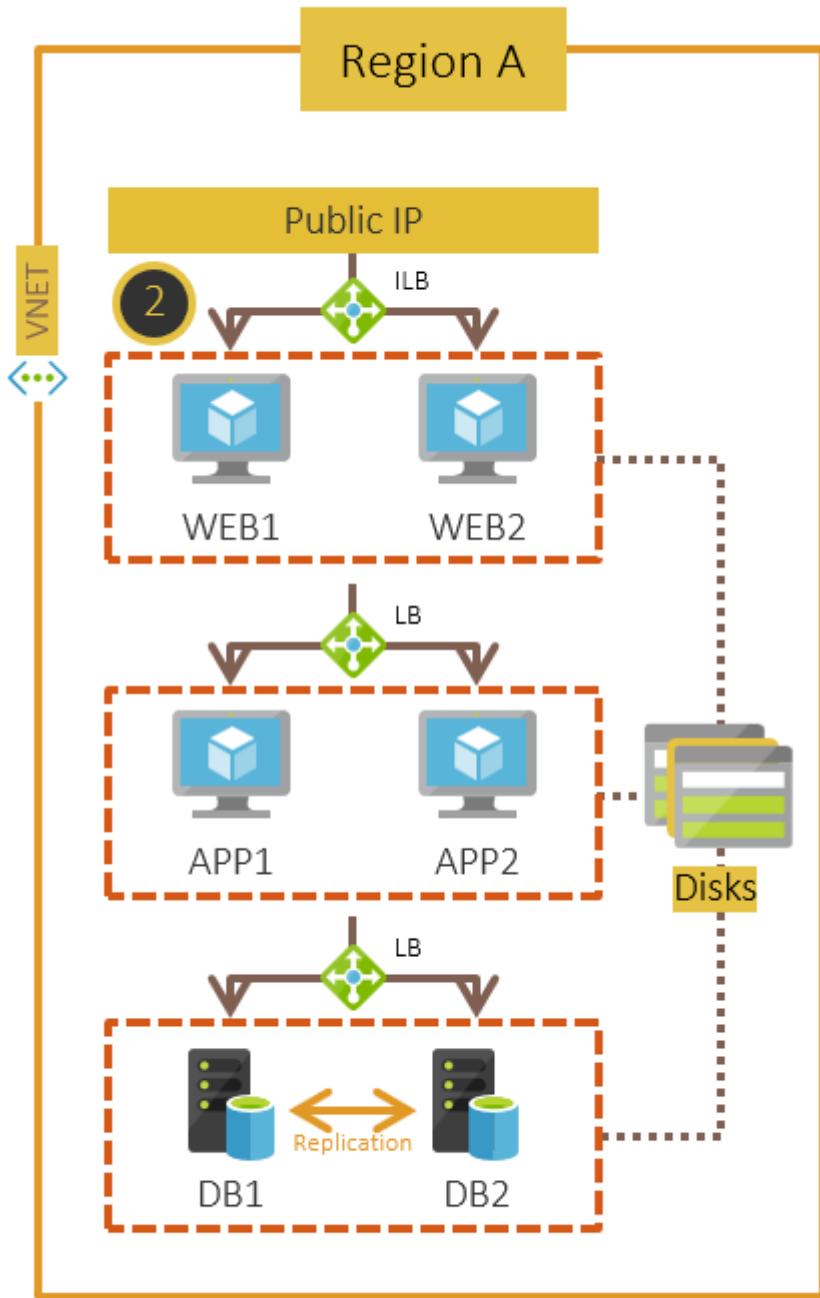
 **Important**

Currently, Azure Site Recovery supports moving VMs from one region to another but doesn't support moving within a region.

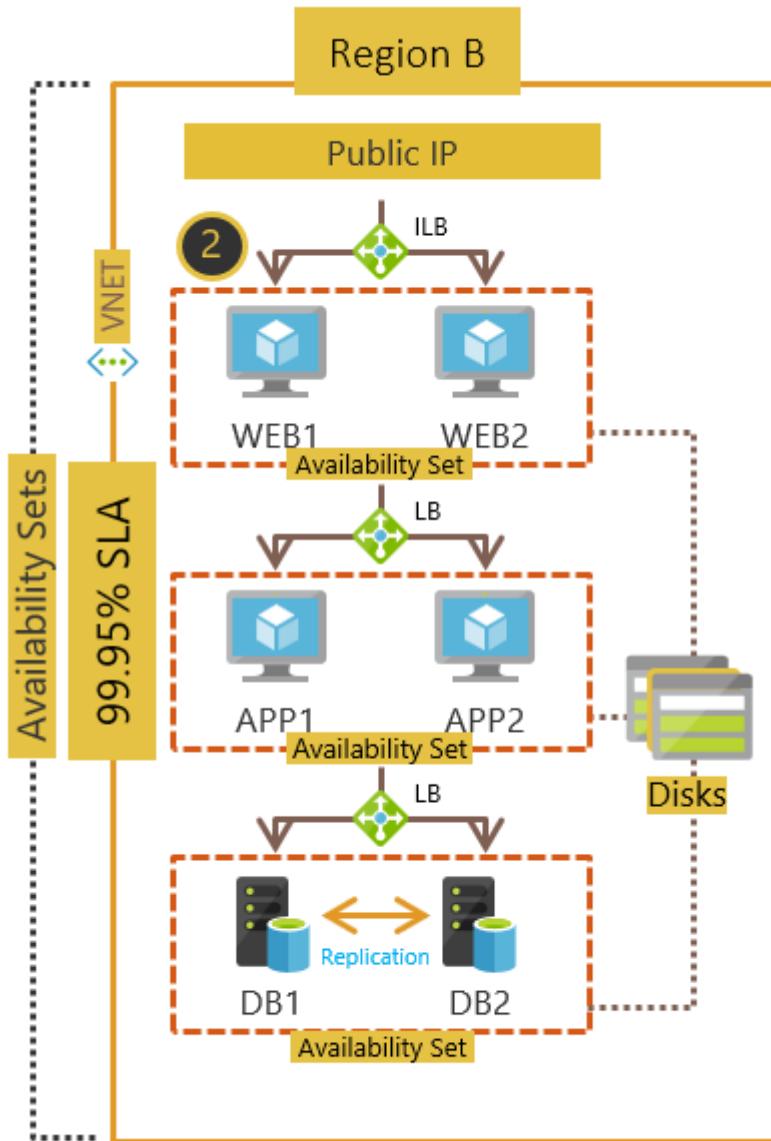
## Typical architectures for a multi-tier deployment

This section describes the most common deployment architectures for a multi-tier application in Azure. The example is a three-tiered application with a public IP. Each of the tiers (web, application, and database) has two VMs each, and they are connected by an Azure load balancer to the other tiers. The database tier has SQL Server Always On replication between the VMs for high availability.

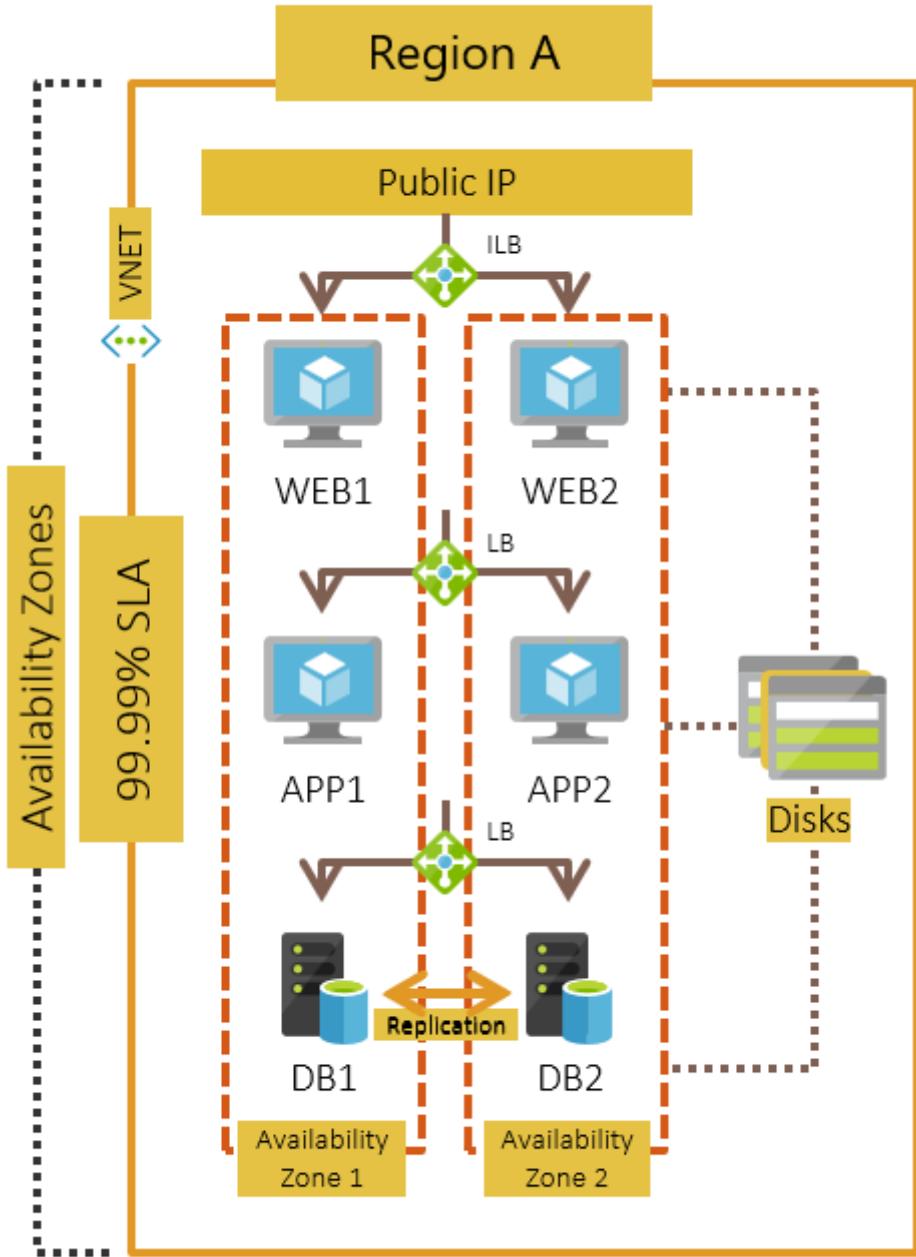
- **Single-instance VMs deployed across various tiers:** Each VM in a tier is configured as a single-instance VM and is connected by load balancers to the other tiers. This configuration is the simplest to adopt.



- **VMs in each tier deployed across availability sets:** Each VM in a tier is configured in an availability set. [Availability sets](#) ensure that the VMs you deploy on Azure are distributed across multiple isolated hardware nodes in a cluster. This ensures that if a hardware or software failure within Azure happens, only a subset of your VMs are affected, and your overall solution remains available and operational.



- **VMs in each tier deployed across Availability Zones:** Each VM in a tier is configured across [Availability Zones](#). An Availability Zone in an Azure region is a combination of a fault domain and an update domain. For example, if you create three or more VMs across three zones in an Azure region, your VMs are effectively distributed across three fault domains and three update domains. The Azure platform recognizes this distribution across update domains to make sure that VMs in different zones are not updated at the same time.



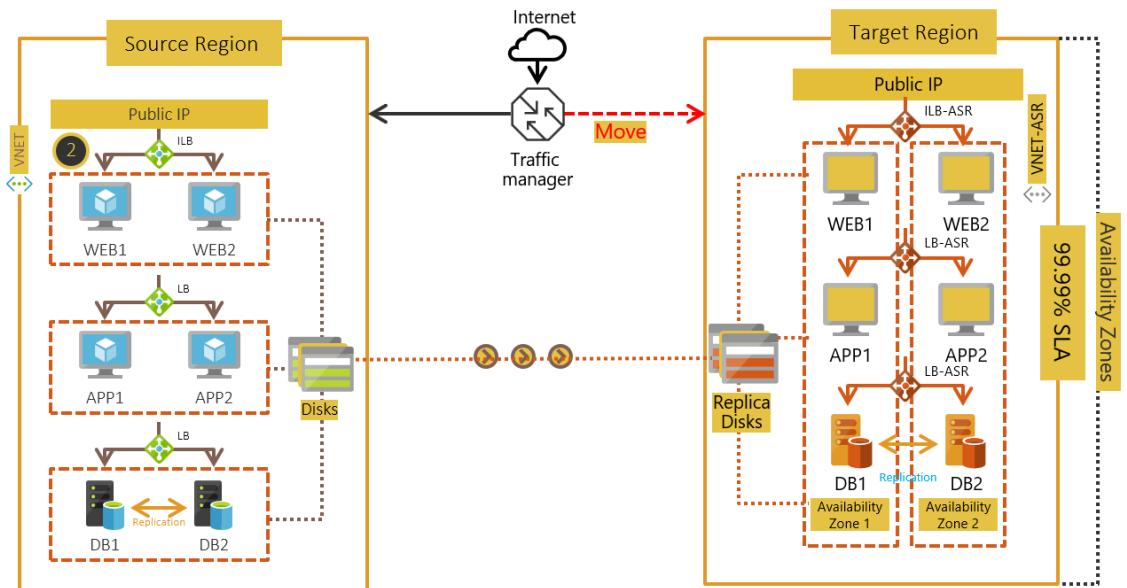
## Move VMs as is to a target region

Based on the [architectures](#) mentioned earlier, here's what the deployments will look like after you perform the move as is to the target region.

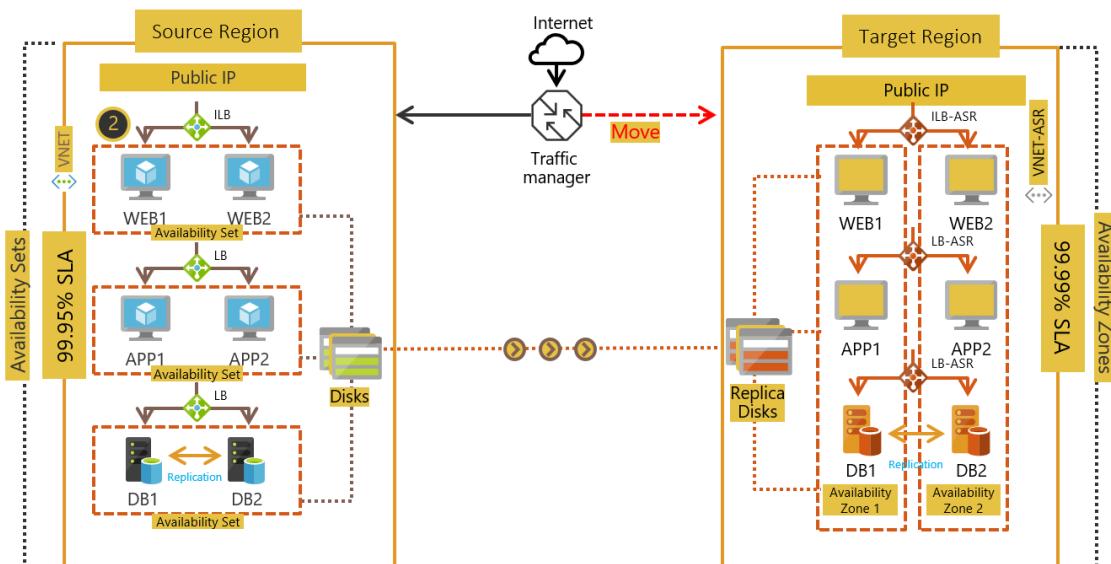
- Single-instance VMs deployed across various tiers
- VMs in each tier deployed across availability sets
- VMs in each tier deployed across Availability Zones

## Move VMs to increase availability

- Single-instance VMs deployed across various tiers



- **VMs in each tier deployed across availability sets:** You can configure your VMs in an availability set into separate Availability Zones when you enable replication for your VM by using Azure Site Recovery. The SLA for availability will be 99.99% after you complete the move operation.



## Next steps

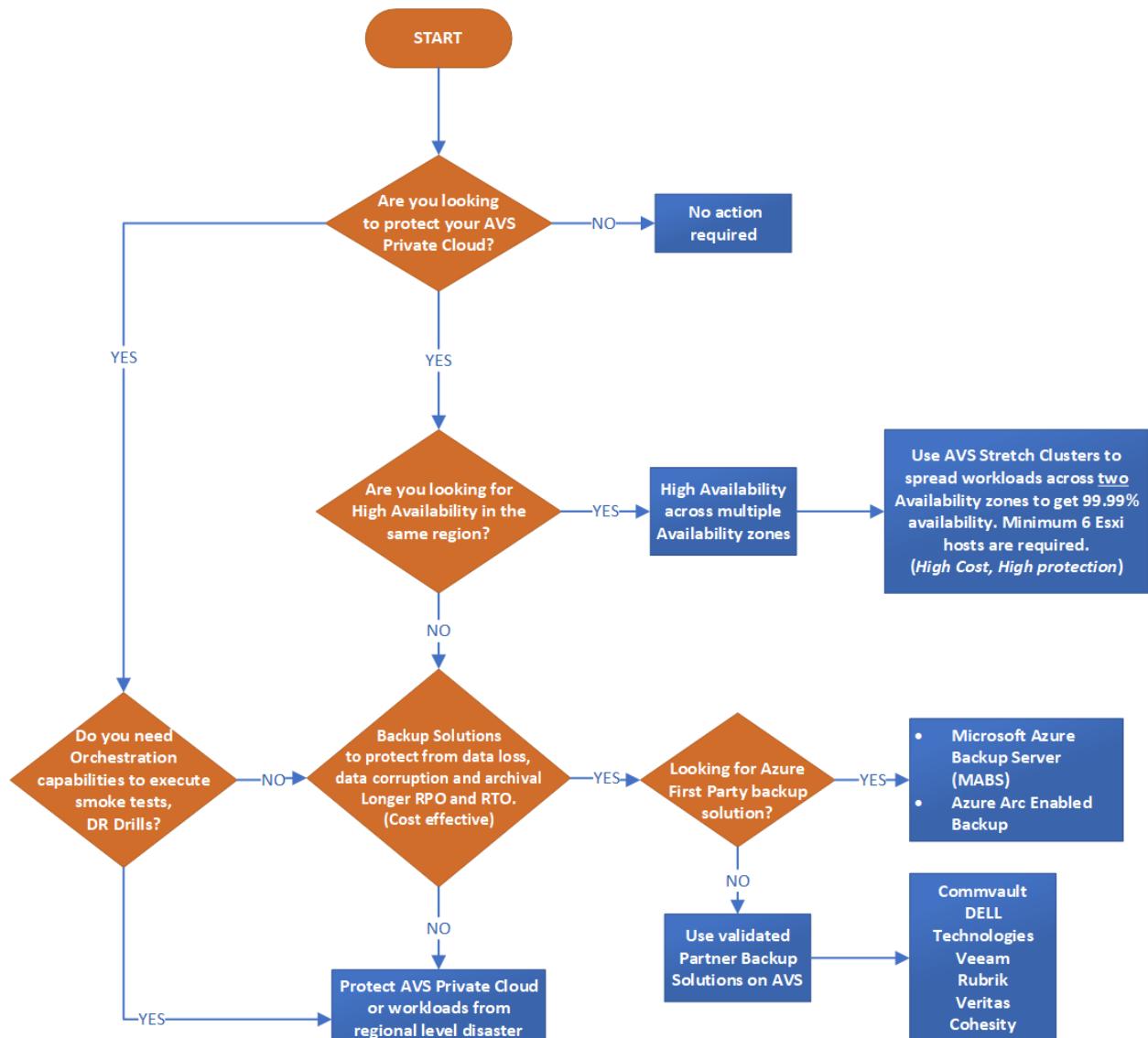
- Move Azure VMs to another region
- Move Azure VMs into Availability Zones

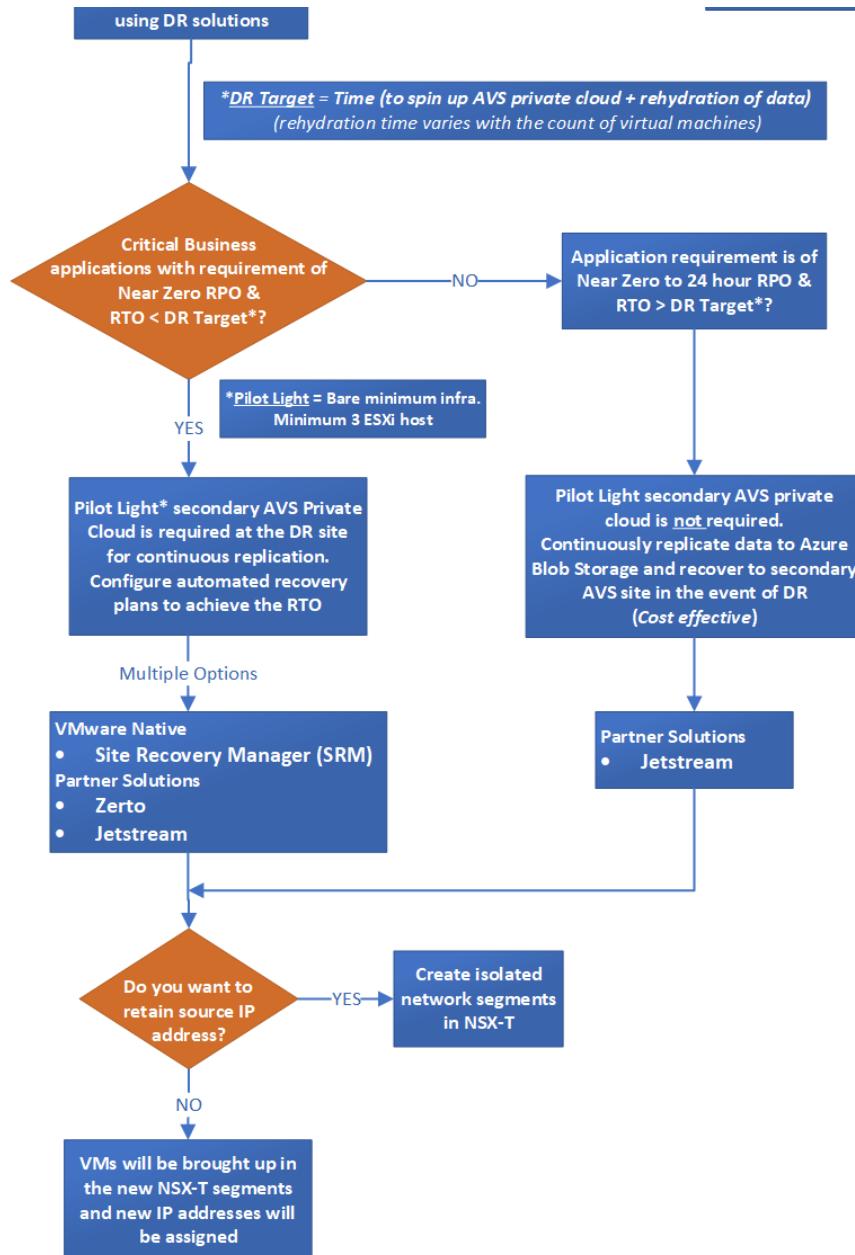
# Business continuity and disaster recovery for Azure VMware Solution

Article • 02/02/2023

This enterprise-scale scenario helps improve business continuity and disaster recovery (BCDR). [Azure VMware Solution](#) provides [private clouds](#) that contain VMware vSphere clusters built from dedicated bare-metal Azure infrastructure. The solution provides a minimum of three ESXi hosts, up to a maximum of 16 hosts per cluster. All provisioned private clouds have VMware vCenter Server, VMware vSAN, VMware vSphere, and VMware NSX-T Data Center. To learn about the service level agreement (SLA) for Azure VMware Solution, see [SLA for Azure VMware Solution](#).

Whether you have an on-premises or Azure VMware Solution, you should consider various BCDR factors to prepare for a disaster. A robust BCDR plan aims to protect a company from data loss, financial loss, and downtime if there's a disruptive event. The following decision tree shows various BCDR options available for Azure VMware Solution.





### ① Note

A pilot light environment is set up with a minimal configuration, with only core components to support a critical set of applications. However, it can scale out and spawn more hosts to take the bulk of the load if a failover occurs. For disaster recovery of compute and memory intensive Azure VMware solution workloads, the same amount of storage is required at the secondary site.

## Business continuity design considerations

- VMware vSAN storage policies on Azure VMware Solution are implemented with storage availability in mind. When the cluster has between three and five hosts, the number of host failures that can be tolerated without data loss equals one. When the cluster has between 6 and 16 hosts, the number of host failures to tolerate

before data loss can occur equals two. VMware vSAN storage policies can be applied on a per-VM basis. While these policies are the default, you can amend the policy to suit custom requirements. For more information, see [Azure VMware Solution storage concepts](#).

- vSphere high availability is enabled by default on Azure VMware Solution. The high availability admittance policy reserves compute and memory capacity for a single node. This reservation ensures sufficient capacity to restart workloads in another node in an Azure VMware Solution cluster.
- High availability with stretch cluster: With Azure VMware Solution, ESXi hosts deployed in a standard vSphere cluster traditionally reside in a single Azure availability zone and are protected by vSphere high availability. However, workloads aren't protected against an availability zone failure. To protect against failure, a single vSAN cluster can span two separate availability zones, called a vSAN stretched cluster. For more information, see [Deploy vSAN stretched clusters](#).
- Choose a validated backup solution for the VMware vSphere virtual machines (VMs), such as [Microsoft Azure Backup Server](#) or a [partner backup solution](#).
- For information about supported features in partner backup solutions, refer to the respective partner documentation.

 **Note**

vCenter Server and NSX-T Data Center configurations for private clouds are backed up hourly, and backups are kept for three days.

- Azure VMware Solution components such as vCenter Server, NSX-T Manager, or HCX Manager are managed services for which backup is managed by Azure. To restore from a backup, [create an Azure Support request](#).

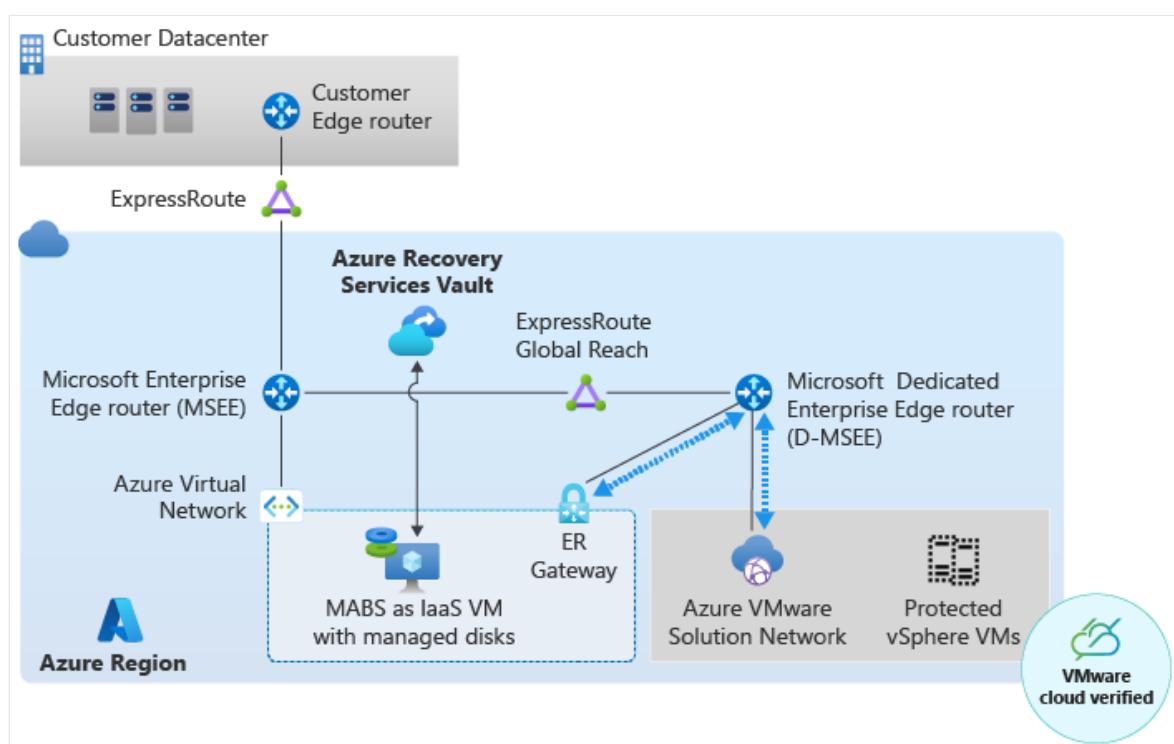
## Business continuity design recommendations

- Use Azure Backup Server to back up the Azure VMware Solution private cloud. For more information, see [Back up VMware vSphere VMs with Azure Backup](#). Supported deployment topologies include [MARS Agent](#) and [Data Protection Manager](#). Each deployment topology has its own support matrix, constraints, and limitations.
- Deploy the Azure Backup Server in the same Azure region as the Azure VMware Solution private cloud. This deployment method reduces traffic costs, eases

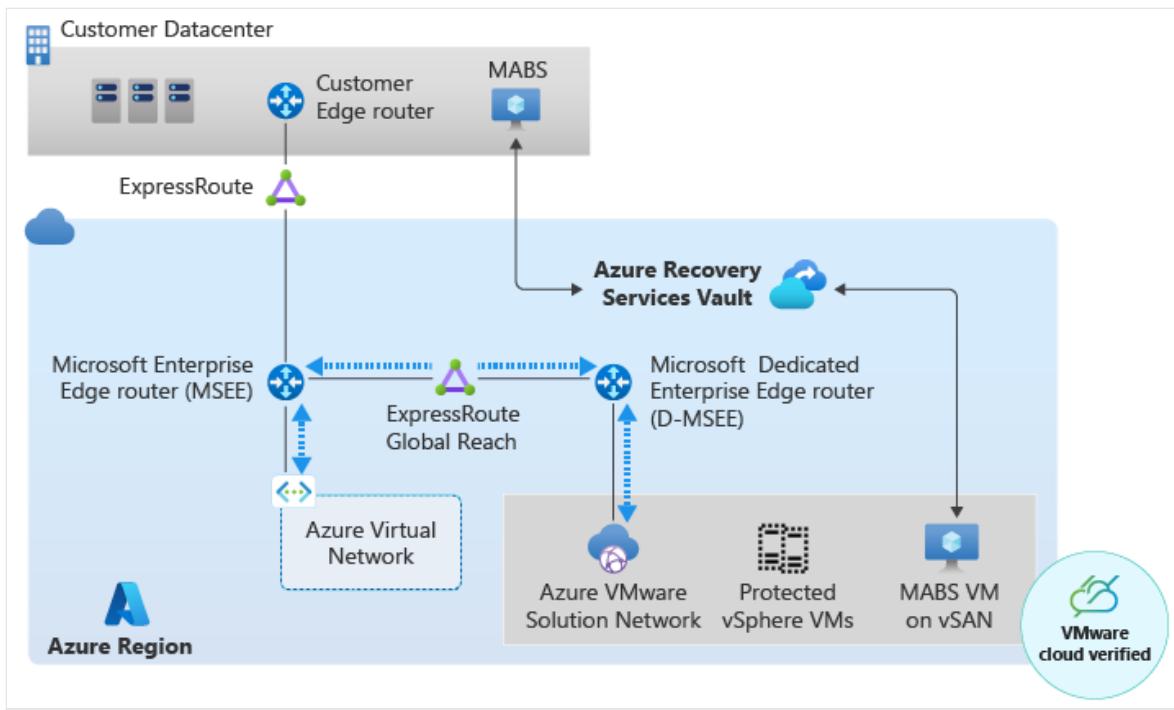
administration, and keeps the primary/secondary topology. See the [Azure regions decision guide](#) for Azure region deployment best practices.

- Azure Backup can be deployed as an Azure infrastructure as a service (IaaS) VM or within the Azure VMware Solution private cloud. It's highly recommended to deploy it outside of the Azure VMware Solution private cloud. Deploy Backup in an Azure virtual network and ensure this virtual network is connected to the same ExpressRoute that's connected to the Azure VMware Solution private cloud. Running Backup Server outside of Azure VMware Solution private cloud helps to reduce vSAN consumption, since vSAN is a *limited capacity* resource within the Azure VMware Solution private cloud.

*Azure Backup Server deployed as an Azure IaaS VM.*



*Azure Backup Server deployed as an Azure VMware Solution VM.*



- Use the [application performance requirements checklist](#) to arrive at the right capacity and disk type, such as HDD, SSD, or Ultra. Consider the Azure IaaS VM SKU that supports the [disk type and capacity](#) for backup operations.
- Use [Azure Backup Server capacity planner](#) to determine number of servers, storage, and IOPS requirements for each of them. When providing "Total Size of the Workload (GB)\*" value in capacity planner, use median value between "used storage" and "allocated storage" of all VMs in vCenter you want to backup.
- Use [storage pools](#) with Azure Backup Server for enhanced disk IOPS/throughput. Use [tiered storage](#) on Backup Server for enhanced operations.
- Identify the number of parallel backup jobs and restore operations to run on Azure Backup server. Currently, 8 parallel backup jobs are supported. Measure the amount of time taken to backup and restore mission-critical workloads over multiple runs. Validate that backup and restore times meet RPO and RTO requirements for Azure Backup server. Ensure that AVS vSAN datastore has enough capacity to hold restored backup.
- Add necessary Antivirus exceptions for Azure Backup Server files and folders as documented [here](#) if any Antivirus/Antimalware software runs on Azure Backup Server. When using DPM protection agent on any Azure VMware Solution VM for application backup(e.g. SQL, Sharepoint, etc.), disable realtime monitoring of *dpmra.exe*.
- Configure appropriate NSG (Network Security Group) rules on subnet hosting Azure Backup Server to allow network communication from DPM protection agent running on protected VM in Azure VMware Solution. DPM protection agent

communicates with Azure Backup Server on any dynamic port [between 1024 and 65535](#).

- Currently, Azure Backup Server doesn't support cross-region restore for Azure VMware Solution private cloud. Refer to [partner backup solutions](#) and [disaster recovery section](#) when cross-region Azure VMware Solution recovery is required.

## Disaster recovery design considerations

- Align business requirements with recovery time objectives (RTO), capacity, and recovery point objectives (RPO) for applications. Plan and design accordingly to achieve these objectives using the most appropriate replication technology. For example, natively replicate SQL databases using SQL Always On availability group, or use a disaster recovery tool such as VMware Site Recovery Manager.
- Determine the target disaster recovery site for the protected Azure VMware Solution private cloud. This site influences which disaster recovery tooling is suitable for the environment. For example, if you want to recover Azure VMware Solution workloads to Azure native IaaS virtual machines, [Zerto](#) is the only solution.
- Determine which subset of Azure VMware Solution workloads requires protection if there's a disaster recovery event. Consider categorizing the workloads based on priority: P0 for business-critical workloads, and P1, P2, P3 for other workloads that are important but not as critical for the business to operate. The customer's business continuity plan defines the priority levels, which helps to control the costs associated with disaster recovery implementation.
- In most cases, non-production environments such as dev, test, or UAT don't need to fail over to a secondary site. You should run the pilot light at the secondary site with reduced capacity for production and critical workloads to save on costs. For more capacity, you can scale out to add ESXi hosts to the cluster during the disaster recovery event.
- For pilot light deployments especially, ensure that you've secured all the host quota needed in the secondary site so that you don't have to wait for the required capacity during full scale out. See [Request host quota for Azure VMware Solution](#).
- Set up functional domain roles, like Active Directory domain controllers, in the secondary environment.
- Solutions from partners like [JetStream](#) and Zerto are generally available and validated on Azure VMware Solution. They support most disaster recovery scenarios and can provide faster recovery with near-zero RPO.

- VMware Site Recovery Manager, Jetstream, and Zerto support migration from third-party locations into Azure VMware Solution.
- [VMware HCX](#) is also a cost-effective disaster recovery solution. However, it's not recommended for large production workloads due to manual orchestration.
- For disaster recovery between Azure VMware Solution private clouds in different Azure regions, you need to enable ExpressRoute Global Reach between both backend ExpressRoute circuits. These circuits create primary-to-secondary private cloud connectivity when required for solutions like VMware SRM and VMware HCX.
- For disaster recovery between Azure VMware Solution private clouds in the same Azure region, you need to enable [Azure VMware Solution Interconnect](#). It creates a routing link between the management and workload networks of the Azure VMware Solution private clouds for communication between the clouds. Ensure that routed IP address space in each private cloud is unique and doesn't overlap.
- When working with disaster recovery, you can use the same source IP address space in the primary Azure region and the secondary Azure region. However, it requires extra design and engineering efforts.
  - Retain the same IP addresses: The virtual machines at the secondary Azure VMware Solution site can be recovered using the same source IP address as the primary site. For this method, create isolated VLANs or NSX-T segments in the secondary site and ensure that none of these isolated VLANs or segments are connected to the environment. Modify your disaster recovery routes to reflect that the subnet has moved to the secondary site and the new IP addresses location. While this method works, it also creates engineering overhead when aiming for fully automated disaster recovery.
  - Use different IP addresses: You can also use different IP addresses for recovered VMs. If the VM is moved to a secondary site, the recovery plan within the VMware Site Recovery Manager details the custom IP map. Select this map for the change of IP address. VMs are brought up in the new NSX-T segments and new IP addresses are assigned. The tooling can differ for different disaster recovery solutions.
- **Important factors for partial and full disaster recovery scenarios:**
  - VMware Site Recovery Manager supports partial recovery, which recovers only a subset of virtual machines, and full disaster recovery. Between two Azure

VMware Solution sites in region 1 and region 2, all or some of the VMs can fail over.

- The requirement of source IP address retention for recovered VMs dictates whether partial versus full disaster recovery is possible.
- In order to maintain the source IP address while performing partial disaster recovery in Site Recovery Manager, the subnet gateway needs to move to the secondary site.

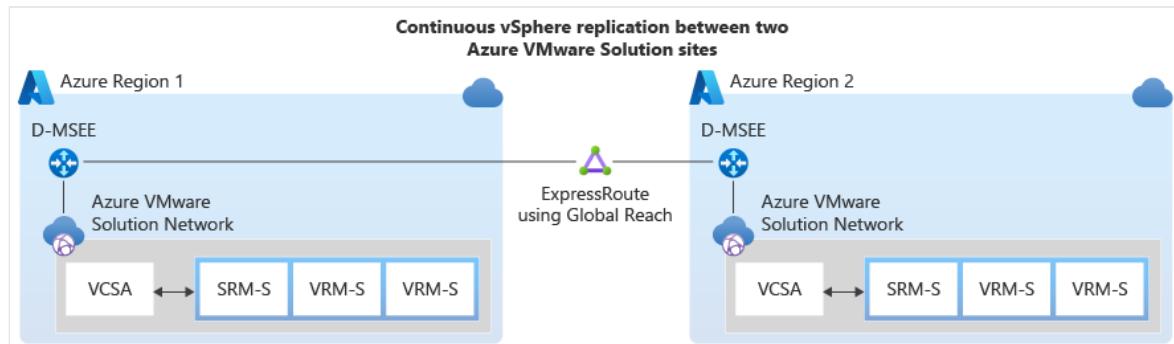
**! Note**

Active-standby disaster recovery doesn't require Layer 2 stretching.

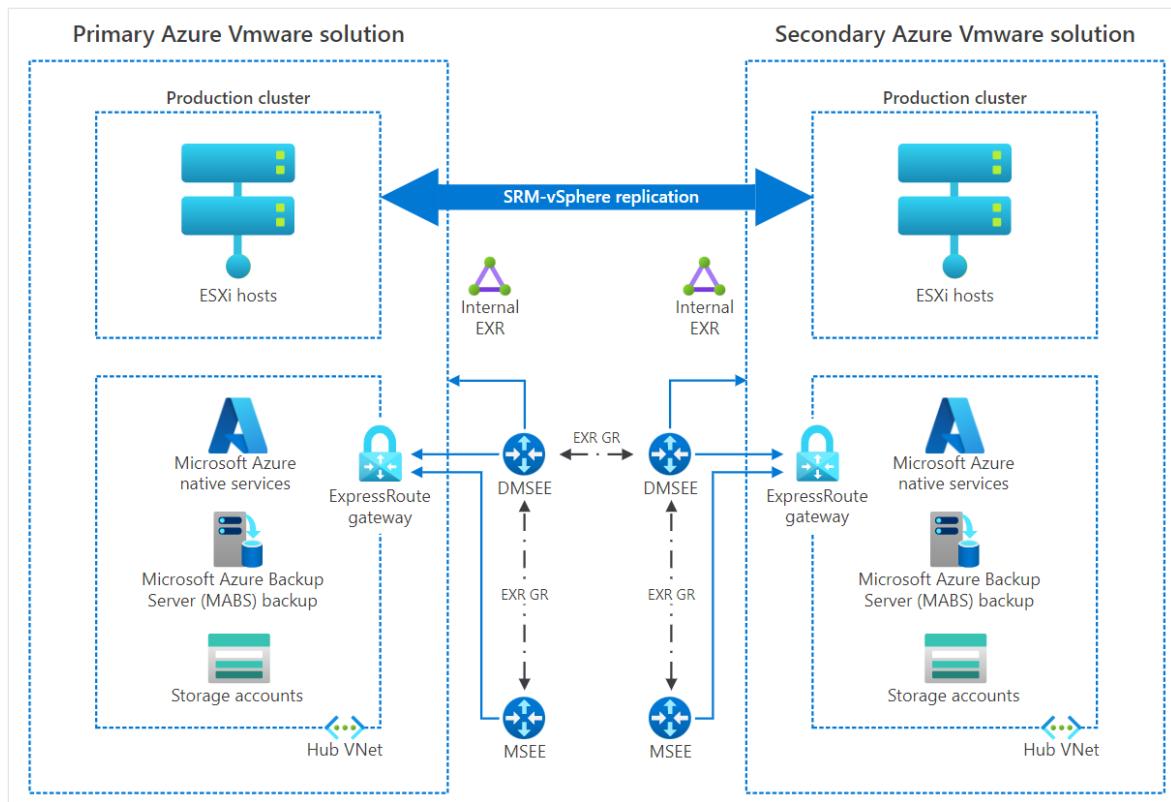
## Disaster recovery design recommendations

- Use [VMware Site Recovery Manager](#) when working with Azure VMware Solution in both primary and secondary sites. Primary and secondary sites are also known as protected and recovery sites, respectively.

*High-level overview of continuous vSphere replication.*



*Detailed example of continuous vSphere replication between primary and secondary sites.*



- For business-critical applications, Zerto and JetStream are available as disaster recovery solutions for Azure VMware Solution private cloud. JetStream and Zerto are built on the foundation of continuous data protection (CDP), using VMware vSphere API for I/O filtering (VAIO) framework, which enables minimal or close to no data loss. It also enables cost-effective disaster recovery by using minimal resources.
- Use Zerto if Azure IaaS virtual machines are the disaster recovery target for the Azure VMware Solution private cloud.
- Minimize manual input by using automated recovery plans within each of the respective disaster recovery solutions. These plans are helpful when working with either VMware Site Recovery Manager or partner solutions. A recovery plan gathers machines into recovery groups for failover. It then helps to define a systematic recovery process by creating independent units that can failover.
- Set up smoke tests or disaster recovery drills at least once a year to ensure recovery plans work as expected. The level of effort running these drills is determined by the orchestration capabilities of the chosen disaster recovery tool.
- Use [geopolitical regional pairs](#) as the secondary disaster recovery environment. Some of the benefits of regional pairs are prioritized region recovery, sequential updates, physical isolation, and data residency.
- Keep address spaces different to avoid overlapping IP addresses between the two sites. For example, you can use `192.168.0.0/16` for region 1 and `10.0.0.0/16` for

region 2.

- Use ExpressRoute Global Reach connectivity between the primary and secondary private clouds in different regions. See more networking considerations and recommendations in the [relevant design area](#).

## Next steps

Learn about considerations and recommendations for the initial deployment of Azure VMware Solution and guidance for operational automation.

[Platform automation for Azure VMware Solution](#)

# Design area: Management for Azure environments

Article • 03/21/2023

This design area establishes a foundation for operations management across your Azure, hybrid, or multicloud environments. You can enhance your learning later on using the operations guidance outlined in [Manage methodology](#) of the Cloud Adoption Framework.

## Design area review

**Involved roles or functions:** This design area is led by [central IT](#) or [cloud operations](#), specifically the [security architects within that team](#). The [cloud platform](#) and [cloud center of excellence](#) will likely be required to define and implement the technical requirements coming from this exercise. More advanced operations guardrails might also require support from [cloud governance](#).

**Scope:** The goal of this exercise is to understand operations management requirements and implement those requirements consistently across all workloads in your cloud platform. The primary scope of this exercise focuses on operations tooling. You'll use operations tooling to manage the collective portfolio of workloads with a set of common tools and processes. This initial set of operations tooling is also referred to as your operations baseline.

**Out of scope:** You can use the operations baseline defined in this exercise consistently across all workloads. The operations baseline can also be expanded with other tools and processes as outlined in the [Manage methodology](#) of the Cloud Adoption Framework. Doing so helps to improve operations for specific technology platforms or individual workloads.

You can also use the operations baseline with the Azure Well-Architected Framework and Microsoft Azure Well-Architected Review to improve the operations and architecture of individual workloads you deploy within your cloud environment. However, any advanced operations, tech platform operations, or workload operations are out of scope for this exercise.

## Design area overview

For stable, ongoing operations in the cloud, a management baseline is required to provide visibility, operations compliance, and protect and recover capabilities.

The management design area focuses on the considerations and recommendations for landing zone design decisions. Also, the [Manage methodology](#) of the Cloud Adoption Framework provides further in-depth guidance for holistic management processes and tools.

## Operations baseline

Use the following operations items to evaluate which operations management tooling you need to include in your operations baseline.

Scope	Context
<a href="#">Inventory &amp; visibility</a>	<p>As cloud environments are implemented and scaled out, management controls that span the environment become increasingly important.</p> <p>No matter the services that are running on top of the landing zone, the management of fundamental elements of the platform is necessary to ensure stable, ongoing operations.</p> <p>These management tools should scale as the environments do.</p> <p>They can include a mix of first-party and third-party tools, depending on your existing investments.</p>
<a href="#">Operational Compliance</a>	<p>Requirements for patching and managing configuration drift.</p> <p>Requirements for automatic or centralize resource optimization and sizing.</p> <p>Requirements for workloads that should only be optimized or resized by the assigned workload teams.</p> <p>Processes for ensuring completion of their regular optimization efforts.</p>
<a href="#">Protect &amp; Recover</a>	<p>Your organization needs to design suitable, platform-level capabilities that application workloads can depend on for a basic level of business continuity and disaster recovery.</p> <p>Specifically, these application workloads have requirements related to <a href="#">recover time objective (RTO)</a> and <a href="#">recovery point objective (RPO)</a>. Be sure that you capture disaster recovery (DR) requirements to identify and address needs for advanced operations.</p>

## Advanced operations

Use the following advanced operations items as discussion points within your cloud architecture and operations teams. These discussions let you explore and agree on the requirements and features to include in your management design.

Scope	Context
-------	---------

Scope	Context
<b>Platform management</b>	<p>When evaluating supported workloads, it's common for those workloads to have dependencies on shared platforms, like SAP, WVD, AVS, SQL, and so on. When technology platforms are used by multiple workloads, advanced operations can't be delegated to a single workload team. In these instances, centralized operations teams need a plan for the ongoing operations of those shared technology platforms. These responsibilities require extra tooling beyond the operations baseline that supports the overall cloud environment.</p>
<b>Workload management</b>	<p>Workloads built on top of the landing zone platform might have specific management requirements in addition to the tools and processes put in place for the platform services.</p> <p>These requirements should be considered in the context of the platform management to ensure that additions or exceptions are known and documented. It's also important to look at these requirements in the broader context. Often, what is thought to be a requirement for a single workload can become a common pattern. Consider these situations as part of the overall platform toolset to avoid unnecessary duplication of effort.</p> <p>For further information on considerations for workload-specific management, review the <a href="#">operational excellence</a> of the Azure Well-Architected Framework.</p>

# Management and monitoring for Azure VMware Solution enterprise-scale scenario

Article • 03/17/2023

Proper management and monitoring are critical to the success of [Azure VMware Solution](#). This enterprise-scale scenario outlines important recommendations for the design of your environment. More guidance is available in the [Azure enterprise-scale landing zone for management and monitoring](#).

As you plan your management and monitoring environment for Azure VMware Solution, it's critical to understand the [shared responsibility matrix](#). The matrix shows which components Microsoft is responsible for, and which ones that you're responsible for managing and monitoring. Microsoft takes care of the ongoing maintenance, security, and management of cloud resources, leaving your company in charge of the things that matter most, like guest OS provisioning, applications, and virtual machines.

## ⓘ Important

To support Azure VMware Solution, it's important to follow the recommendations below to configure service health alerts.

## Platform management and monitoring

Review the following *considerations* for platform management and monitoring of Azure VMware Solution.

## Azure tooling considerations

- Create alerts and dashboards for the metrics that are most important to your operations teams. See [Configure alerts for Azure VMware Solution](#) for available monitoring and alerting metrics. An example monitoring dashboard is [available on GitHub](#).
- vSAN storage is a limited resource that needs to be managed to maintain availability and performance. Familiarize yourself with [Azure VMware Solution storage concepts](#). Use vSAN storage for guest virtual machine (VM) workloads

only. Review the following design considerations to reduce unnecessary storage use on vSAN.

- [Configure content libraries on Azure Blob Storage](#) to move VM template storage off of vSAN.
- Store backups on an Azure VM, either with [Microsoft tooling](#) or with a [partner vendor](#).
- The Activity Log provides a record of operations performed within Azure. These operations include creation, updates, deletion, and special operations like listing credentials or keys. For example, Azure VMware Solution will emit a [List PrivateClouds AdminCredentials](#) whenever someone visits the *Identity* tab within the Azure portal or programmatically requests `cloudadmin` credentials. Alert rules can be configured to send notifications when specific activities are logged.
- Azure VMware Solution uses a local identity provider. After deployment, use a single administrative user account for the initial Azure VMware Solution configurations. Integrating Azure VMware Solution with [Active Directory](#) allows traceability of actions to users. Review guidance from the [identity portion of the landing zone](#).

## VMware tooling considerations

- Consider VMware solutions like vRealize Operations Manager and vRealize Network Insights to provide a detailed understanding of the Azure VMware Solution platform. Customers can see monitoring data like vCenter Server events and flow logs for the NSX-T Data Center distributed firewall.
- Metrics available in vRealize Operations are documented in [VMware's vRealize Operations documentation](#).
- *Pull* logging is currently supported by vRealize Log Insight for Azure VMware Solution. Only events, tasks, and alarms can be captured. Syslog pushing of unstructured data from hosts to vRealize isn't currently supported. SNMP Traps aren't supported.
- While Microsoft monitors the health of vSAN, it's possible to utilize vCenter Server to query and monitor the performance of vSAN. Performance metrics can be viewed from a VM or backend perspective, showing average latency, IOPS, throughput, and outstanding IO through vCenter.
- vCenter Server logs can be sent to Storage Accounts or Event Hubs using the Diagnostic Settings within the Private Cloud resource in Azure. Log settings aren't directly configurable within vCenter Server, only via the Private Cloud resource in Azure. More information is available in the [configuring VMware syslog documentation](#). The output is raw syslog, so consider retention and downstream processing before enabling.

- In-guest memory collection isn't supported by vRealize Operations using VMware tools. Active and consumed memory will continue to work.

## Guest workload management considerations

- Virtual machines within Azure VMware Solution are treated the same as on-premises VMware vSphere VMs by default. You can continue using existing VM-level monitoring within AVS via existing agents.
- Azure VMware Solution VMs won't show up in the Azure portal unless [Azure Arc for Servers](#) is deployed to them. Azure Arc for Servers allows for an agent-based approach to VM management & monitoring from the Azure control plane. You can apply Azure Policy guest configurations, protect servers with Microsoft Defender, and deploy the Azure Monitor agent to the guest VMs.

## Design recommendations

Review the following *recommendations* for platform management and monitoring of Azure VMware Solution.

## Azure tooling recommendations

- Configure [Azure Service Health](#) to send alerts for service issues, planned maintenance, and other events that could impact Azure VMware Solution and other services. These notifications are sent to Action Groups, which can be used to send email, SMS, push notifications, and voice calls to addresses of your choice. Actions can also trigger Azure and third-party systems, including Azure Functions, Logic Apps, Automation Runbooks, Event Hubs, and Webhooks.
- Monitor baseline performance of Azure VMware Solution infrastructure through [Azure Monitor Metrics](#). These metrics can be queried and filtered from the Azure portal, queried via REST API, or directed to Log Analytics, Azure Storage, Event Hubs, or [Partner Integrations](#).
- Configure the following [alerts in Azure Monitor](#) to provide warnings if the cluster nears dangerous values for disk, CPU, or RAM usage:

Metric	Alert
Disk - Percentage Datastore Disk Used (%)	>70% warning
Disk - Percentage Datastore Disk Used (%)	>75% critical

Metric	Alert
CPU - Percentage CPU (%)	>80% warning
Memory - Average Memory Usage (%)	>80% warning

- You can automate the creation of [Azure Monitor Alerts](#) and [Azure Service Health alerts](#).
- For service-level agreement (SLA) purposes, Azure VMware Solution requires slack space of 25 percent available on vSAN.
- For SLA purposes, Azure VMware Solution requires the number of failures to `tolerate = 1` for clusters that have between three and five hosts, and the number of failures to `tolerate = 2` for clusters with 6-to-16 hosts. The full SLA is documented in the following [service level agreement](#).
- In a hybrid environment, you can use [Connection Monitor](#) to monitor communication between on-premises and Azure resources.
- Configure two connection monitors in [Azure Network Watcher](#) to monitor connectivity.
  - [Configure Connection Monitor](#) to view the availability and performance of the network connections within, from, and to the Azure VMware Solution, including ExpressRoute Direct and ExpressRoute Global Reach connections.
- Send your logs to Log Analytics. For more information, see [Send Logs to Log Analytics](#).

## VMware tooling recommendations

- During workload migration, use the "monitor-as-on-premises" model to minimize changes during migration and provide vSphere Administrators with the experience they're accustomed to.
- Monitor [vSphere Health Status](#).
  - Create [vSphere events, alarms, and actions](#).
- Consider using vRealize Log Insight for [monitoring a NSX-T Data Center environment](#).

## Guest workload management recommendations

Review the following recommendations for guest management and for monitoring of workloads running in Azure VMware Solution.

- During workload migration, use the "monitor-as-on-premises" model to minimize changes during migration. After migration, consider using [Azure Arc for Servers](#) to enable management and monitoring of Azure VMware Solution-hosted workloads with Azure native solutions.
- The default storage policy uses thick provisioning. For efficient use of vSAN capacity, evaluate using thin provisioning for VMs. Each VM's disk configuration can vary. A VM can have thick or thin disks, or both, depending on the requirements for the workload.
- Configure guest monitoring for VMs by following the [hybrid guidance](#) for Windows and Linux. Configure both Windows and Linux this way for the following Azure integrations:

Integration	Description
<a href="#">Log Analytics</a>	Primary tool for aggregating, querying, and interactively analyzing logs generated by Azure resources.
<a href="#">Microsoft Defender for Cloud</a>	Unified infrastructure security management system that strengthens security posture by providing advanced threat protection across hybrid and Azure resources.
<a href="#">Microsoft Sentinel</a>	Cloud-native security information and event management solution. This Azure resource provides security analytics, alert detection, and automated threat response across on-premises and cloud environments.
<a href="#">Azure Update Management</a>	Manages operating system updates for Windows and Linux machines on-premises and in cloud environments.
<a href="#">Azure Monitor</a>	Comprehensive monitoring solution for collecting, analyzing, and acting upon telemetry from cloud and on-premises environments.

## Storage considerations

To help with storage-heavy workloads that need more storage capacity than vSAN provides based on the CPU and memory requirements, consider using [Azure NetApp Files](#) to extend your storage footprint into Azure native storage services.

Azure VMware Solution supports attaching Network File System (NFS) datastores as a persistent storage option. You can create NFS datastores with Azure NetApp Files volumes and attach them to clusters of your choice. By using NFS datastores backed by Azure NetApp Files, you can extend your storage instead of scaling the clusters. You can

also use Azure NetApp Files volumes to replicate data from on-premises or primary VMware environments to a secondary site.

For more information read [Azure NetApp Files datastores for Azure VMware Solution](#).

## Other considerations

- If you use a network virtual appliance, consider monitoring trace logs between on-premises and Azure resources. Ensure monitoring is in place between Azure and Azure VMware Solution.

## Next steps

Learn about design considerations for Azure VMware Solution business continuity and disaster recovery in an enterprise-scale scenario.

[Business continuity and disaster recovery for Azure VMware Solution](#)

# Backup cloud and on-premises workloads to cloud

Article • 04/24/2023

Azure Backup comprehensively protects your data assets in Azure through a simple, secure, and cost-effective solution that requires zero-infrastructure. It's Azure's built-in data protection solution for a wide range of workloads. It helps protect your mission critical workloads running in the cloud, and ensures your backups are always available and managed at scale across your entire backup estate.

## Intended audience

The primary target audience for this article is the IT and application administrators, and implementers of large and mid-sized organizations, who want to learn about the capabilities of Azure's built-in data protection technology, Azure Backup, and to implement solutions to protect your deployments efficiently. The article assumes you're familiar with core Azure technologies, data protection concepts and have experience working with a backup solution. The guidance covered in this article can make it easier to design your backup solution on Azure using established patterns and avoid known pitfalls.

## How this article is organized

While it's easy to start protecting infrastructure and applications on Azure, when you ensure that the underlying Azure resources are set up correctly and being used optimally you can accelerate your time to value. This article covers a brief overview of design considerations and guidance for optimally configuring your Azure Backup deployment. It examines the core components (for example, Recovery Services vault, Backup Policy) and concepts (for example, governance) and how to think of them and their capabilities with links to detailed product documentation.

## Get started

### Subscription design strategy

Apart from having a clear roadmap to navigate through the Cloud Adoption Journey, you must plan your cloud deployment's subscription design and account structure to match your organization's ownership, billing, and management capabilities. As the vault

is scoped to a subscription, your Subscription design will highly influence your Vault design. [Learn more](#) about different Subscription Design Strategies and guidance on when to use them.

## Document your Backup requirements

To get started with Azure Backup, plan your backup needs. Following are some of the questions you should ask yourself while formulating a perfect backup strategy.

### **What workload type do you wish to protect?**

To design your vaults, ensure if you require a centralized/ decentralized mode of operation.

### **What's the required backup granularity ?**

Determine if it should be application consistent, crash consistent, or log backup.

### **Do you've any compliance requirements?**

Ensure if you need to enforce security standards and separate access boundaries.

### **What's the required RPO, RTO?**

Determine the backup frequency and the speed of restore.

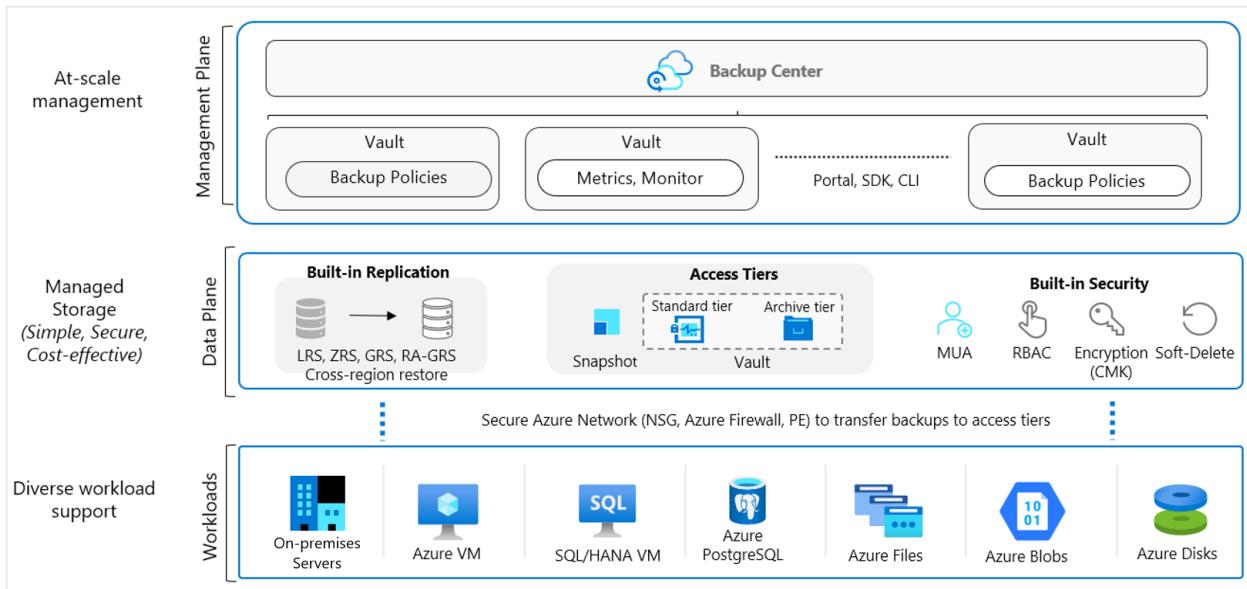
### **Do you've any Data Residency constraints?**

Determine the storage redundancy for the required Data Durability.

### **How long do you want to retain the backup data?**

Decide on the duration the backed-up data be retained in the storage.

## Architecture



## Workloads

Azure Backup enables data protection for various workloads (on-premises and cloud). It's a secure and reliable built-in data protection mechanism in Azure. It can seamlessly scale its protection across multiple workloads without any management overhead for you. There are multiple automation channels as well to enable this (via PowerShell, CLI, Azure Resource Manager templates, and REST APIs.)

- **Scalable, durable, and secure storage:** Azure Backup uses reliable Blob storage with in-built security and high availability features. You can choose LRS, GRS, or RA-GRS storages for your backup data.
- **Native workload integration:** Azure Backup provides native integration with Azure Workloads (VMs, SAP HANA, SQL in Azure VMs and even Azure Files) without requiring you to manage automation or infrastructure to deploy agents, write new scripts or provision storage.

[Learn more](#) about supported workloads.

## Data plane

- **Automated storage management:** Azure Backup automates provisioning and managing storage accounts for the backup data to ensure it scales as the backup data grows.
- **Malicious delete protection:** Protect against any accidental and malicious attempts for deleting your backups via soft delete of backups. The deleted backup data is stored for 14 days free of charge and allows it to be recovered from this state.

- **Secure encrypted backups:** Azure Backup ensures your backup data is stored in a secure manner, leveraging built-in security capabilities of the Azure platform, such as Azure role-based access control (Azure RBAC) and Encryption.
- **Backup data lifecycle management:** Azure Backup automatically cleans up older backup data to comply with the retention policies. You can also tier your data from operational storage to vault storage.
- **Protected critical operations:** Multi-user authorization (MUA) for Azure Backup allows you to add an additional layer of protection to critical operations on your Recovery Services vaults.

## Management plane

- **Access control:** Vaults (Recovery Services and Backup vaults) provide the management capabilities and are accessible via the Azure portal, Backup Center, Vault dashboards, SDK, CLI, and even REST APIs. It's also an Azure role-based access control (Azure RBAC) boundary, providing you with the option to restrict access to backups only to authorized Backup Admins.
- **Policy management:** Azure Backup Policies within each vault define when the backups should be triggered and the duration they need to be retained. You can also manage these policies and apply them across multiple items.
- **Monitoring and Reporting:** Azure Backup integrates with Log Analytics and provides the ability to see reports via Workbooks as well.
- **Snapshot management:** Azure Backup takes snapshots for some Azure native workloads (VMs and Azure Files), manages these snapshots and allows fast restores from them. This option drastically reduces the time to recover your data to the original storage.

## Vault considerations

Azure Backup uses vaults (Recovery Services and Backup vaults) to orchestrate, manage backups, and store backed-up data. Effective vault design helps organizations establish a structure to organize and manage the backup assets in Azure to support your business priorities. Consider the following guidelines when creating a vault.

## Single or multiple vaults

To use a single vault or multiple vaults to organize and manage your backup, see the following guidelines:

- **Protect resources across multiple regions globally:** If your organization has global operations across North America, Europe, and Asia, and your resources are deployed in East-US, UK West, and East Asia. One of the requirements of Azure Backup is that the vaults are required to be present in the same region as the resource to be backed-up. Therefore, you should create three separate vaults for each region to protect your resources.
- **Protect resources across various Business Units and Departments:** Consider that your business operations are divided into three separate Business Units (BU), and each business unit has its own set of departments (five departments - Finance, Sales, HR, R & D, and Marketing). Your business needs may require each department to manage and access their own backups and restores; also, enable them to track their individual usage and cost expense. For such scenarios, we recommend you to create one vault for each department in a BU. This way, you'll have 15 Vaults across your organization.
- **Protect different workloads:** If you plan to protect different types of workloads (such as 150 VMs, 40 SQL databases, and 70 PostgreSQL databases), then we recommend you create separate vaults for each type of workload (for this example, you need to create three vaults for each workload - VMs, SQL databases, and PostgreSQL databases). This helps you to separate access boundaries for the users by allowing you to grant access (using Azure role-based access control – Azure RBAC) to the relevant stakeholders.
- **Protect resources running in multiple environments:** If your operations require you to work on multiple environments, such as production, non-production, and developer, then we recommend you create separate vaults for each.
- **Protect large number (1000+) of Azure VMs:** Consider that you have 1500 VMs to back up. Azure Backup allows only 1000 Azure VMs to be backed-up in one vault. For this scenario, you can create two different vaults and distribute the resources as 1000 and 500 VMs to respective vaults or in any combination considering the upper limit.
- **Protect large number (2000+) of diverse workloads:** While managing your backups at scale, you'll protect the Azure VMs, along with other workloads, such as SQL and SAP HANA database running on those Azure VMs. For example, you've 1300 Azure VMs and 2500 SQL databases to protect. The vault limits allow you to back up 2000 workloads (with a restriction of 1000 VMs) in each vault. Therefore, mathematically you can back up 2000 workloads in one vault (1000 VMs + 1000

SQL databases) and rest 1800 workloads in a separate vault (300 VMs + 1500 SQL databases).

However, this type of segregation isn't recommended as you won't be able to define access boundaries and the workloads won't be isolated from each other. So, to distribute the workloads correctly, create four vaults. Two vaults to back up the VMs (1000 VMs + 300 VMs) and the other two vaults to back up the SQL databases (2000 databases + 500 databases).

- You can manage them with:
  - Backup center allows you to have a single pane to manage all Backup tasks. [Learn more here](#).
  - If you need consistent policy across vaults, then you can use Azure Policy to propagate backup policy across multiple vaults. You can write a custom [Azure Policy definition](#) that uses the '`deployifnotexists`' effect to propagate a backup policy across multiple vaults. You can also [assign](#) this Azure Policy definition to a particular scope (subscription or RG), so that it deploys a 'backup policy' resource to all Recovery Services vaults in the scope of the Azure Policy assignment. The settings of the backup policy (such as backup frequency, retention, and so on) should be specified by the user as parameters in the Azure Policy assignment.
- As your organizational footprint grows, you might want to move workloads across subscriptions for the following reasons: align by backup policy, consolidate vaults, trade-off on lower redundancy to save on cost (move from GRS to LRS). Azure Backup supports moving a Recovery Services vault across Azure subscriptions, or to another resource group within the same subscription. [Learn more here](#).

## Review default settings

Review the default settings for Storage Replication type and Security settings to meet your requirements before configuring backups in the vault.

- *Storage Replication type* by default is set to Geo-redundant (GRS). Once you configure the backup, the option to modify is disabled. Follow [these](#) steps to review and modify the settings.
  - Non-critical workloads like non-prod and dev are suitable for LRS storage replication.
  - Zone redundant storage (ZRS) is a good storage option for a high Data Durability along with Data Residency.

- Geo-Redundant Storage (GRS) is recommended for mission-critical workloads, such as the ones running in production environment, to prevent permanent data loss, and protect it in case of complete regional outage or a disaster in which the primary region isn't recoverable.
- *Soft delete* by default is Enabled on newly created vaults to protect backup data from accidental or malicious deletes. Follow [these](#) steps to review and modify the settings.
- *Cross Region Restore* allows you to restore Azure VMs in a secondary region, which is an Azure paired region. This option allows you to conduct drills to meet audit or compliance requirements, and to restore the VM or its disk if there's a disaster in the primary region. CRR is an opt-in feature for any GRS vault. [Learn more here](#).
- Before finalizing your vault design, review the [vault support matrixes](#) to understand the factors that might influence or limit your design choices.

## Backup Policy considerations

Azure Backup Policy has two components: *Schedule* (when to take backup) and *Retention* (how long to retain backup). You can define the policy based on the type of data that's being backed up, RTO/RPO requirements, operational or regulatory compliance needs and workload type (for example, VM, database, files). [Learn more](#)

Consider the following guidelines when creating Backup Policy:

### Schedule considerations

While scheduling your backup policy, consider the following points:

- For mission-critical resources, try scheduling the most frequently available automated backups per day to have a smaller RPO. [Learn more](#)  
If you need to take multiple backups per day for Azure VM via the extension, see the workarounds in the [next section](#).
- For a resource that requires the same schedule start time, frequency, and retention settings, you need to group them under a single backup policy.
- We recommend you to keep the backup scheduled start time during non-peak production application time. For example, it's better to schedule the daily automated backup during night, around 2-3 AM, rather than scheduling it in the day time when the usage of the resources high.

- To distribute the backup traffic, we recommend you back up different VMs at different times of the day. For example, to back up 500 VMs with the same retention settings, we recommend you to create 5 different policies associating them with 100 VMs each and scheduling them few hours apart.

## Retention considerations

- Short-term retention can be "daily". Retention for "Weekly", "monthly" or "yearly" backup points is referred to as Long-term retention.
- Long-term retention:
  - Planned (compliance requirements) - if you know in advance that data is required years from the current time, then use Long-term retention. Azure Backup supports back up of Long-Term Retention points in the archive tier, along with Snapshots and the Standard tier. [Learn more](#) about supported workloads for Archive tier and retention configuration.
  - Unplanned (on-demand requirement) - if you don't know in advance, then use you can use on-demand with specific custom retention settings (these custom retention settings aren't impacted by policy settings).
- On-demand backup with custom retention - if you need to take a backup not scheduled via backup policy, then you can use an on-demand backup. This can be useful for taking backups that don't fit your scheduled backup or for taking granular backup (for example, multiple IaaS VM backups per day since scheduled backup permits only one backup per day). It's important to note that the retention policy defined in scheduled policy doesn't apply to on-demand backups.

## Optimize Backup Policy

- As your business requirements change, you might need to extend or reduce retention duration. When you do so, you can expect the following:
  - If retention is extended, existing recovery points are marked and kept in accordance with the new policy.
  - If retention is reduced, recovery points are marked for pruning in the next clean-up job, and subsequently deleted.
  - The latest retention rules apply for all retention points (excluding on-demand retention points). So if the retention period is extended (for example to 100 days), then when the backup is taken, followed by retention reduction (for example from 100 days to seven days), all backup data will be retained according to the last specified retention period (that is, 7 days).

- Azure Backup provides you with the flexibility to *stop protecting and manage your backups*:
  - *Stop protection and retain backup data.* If you're retiring or decommissioning your data source (VM, application), but need to retain data for audit or compliance purposes, then you can use this option to stop all future backup jobs from protecting your data source and retain the recovery points that have been backed up. You can then restore or resume VM protection.
  - *Stop protection and delete backup data.* This option will stop all future backup jobs from protecting your VM and delete all the recovery points. You won't be able to restore the VM nor use Resume backup option.
  - If you resume protection (of a data source that has been stopped with retain data), then the retention rules will apply. Any expired recovery points will be removed (at the scheduled time).
- Before completing your policy design, it's important to be aware of the following factors that might influence your design choices.
  - A backup policy is scoped to a vault.
  - There's a limit on the number of items per policy (for example, 100 VMs). To scale, you can create duplicate policies with the same or different schedules.
  - You can't selectively delete specific recovery points.
  - You can't completely disable the scheduled backup and keep the data source in a protected state. The least frequent backup you can configure with the policy is to have one weekly scheduled backup. An alternative would be to stop protection with retain data and enable protection each time you want to take a backup, take an on-demand backup, and then turn off protection but retain the backup data. [Learn more here](#).

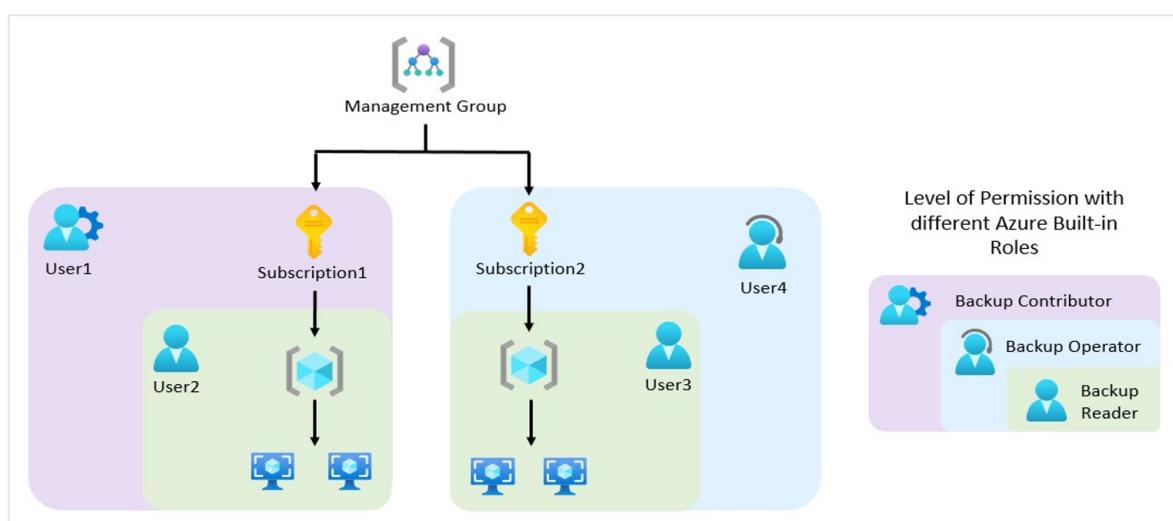
## Security considerations

To help you protect your backup data and meet the security needs of your business, Azure Backup provides confidentiality, integrity, and availability assurances against deliberate attacks and abuse of your valuable data and systems. Consider the following security guidelines for your Azure Backup solution:

## Authentication and authorization using Azure role-based access control (Azure RBAC)

- Azure role-based access control (Azure RBAC) enables fine-grained access management, segregation of duties within your team and granting only the amount of access to users necessary to perform their jobs. [Learn more here](#).
- If you've multiple workloads to back up (such as Azure VMs, SQL databases, and PostgreSQL databases) and you've multiple stakeholders to manage those backups, it is important to segregate their responsibilities so that user has access to only those resources they're responsible for. Azure role-based access control (Azure RBAC) enables granular access management, segregation of duties within your team, and granting only the types of access to users necessary to perform their jobs. [Learn more](#)
- You can also segregate the duties by providing minimum required access to perform a particular task. For example, a person responsible for monitoring the workloads shouldn't have access to modify the backup policy or delete the backup items. Azure Backup provides three built-in roles to control backup management operations: Backup contributors, operators, and readers. [Learn more here](#). For information about the minimum Azure role required for each backup operation for Azure VMs, SQL/SAP HANA databases, and Azure File Share, see [this guide](#).
- Azure role-based access control (Azure RBAC) also provides the flexibility to build [Custom Roles](#) based on your individual requirements. If you're unsure about the types of roles recommended for specific operation, you can utilize the built-in roles provided by Azure role-based access control (Azure RBAC) and get started.

The following diagram explains about how different Azure built-in roles work:



- In the above diagram, *User2* and *User3* are Backup Readers. Therefore, they have the permission to only monitor the backups and view the backup services.
- In terms of the scope of the access,

- *User2* can access only the Resources of *Subscription1*, and *User3* can access only the Resources of *Subscription2*.
- *User4* is a Backup Operator. It has the permission to enable backup, trigger on-demand backup, trigger restores, along with the capabilities of a Backup Reader. However, in this scenario, its scope is limited only to *Subscription2*.
- *User1* is a Backup Contributor. It has the permission to create vaults, create/modify/delete backup policies, and stop backups, along with the capabilities of a Backup Operator. However, in this scenario, its scope is limited only to *Subscription1*.
- Storage accounts used by Recovery Services vaults are isolated and can't be accessed by users for any malicious purposes. The access is only allowed through Azure Backup management operations, such as restore.

## Encryption of data in transit and at rest

Encryption protects your data and helps you to meet your organizational security and compliance commitments.

- Within Azure, data in transit between Azure storage and the vault is protected by HTTPS. This data remains within the Azure network.
- Backup data is automatically encrypted using Microsoft-managed keys. Alternatively, you can use your own keys, also known as [customer managed keys](#). Also, using CMK encryption for backup doesn't incur additional costs. However, the use of Azure Key Vault, where your key is stored, incur costs, which are a reasonable expense in return for the higher data security.
- Azure Backup supports backup and restore of Azure VMs that have their OS/data disks encrypted with Azure Disk Encryption (ADE). [Learn more](#)

## Protection of backup data from unintentional deletes with soft-delete

You may encounter scenarios where you've mission-critical backup data in a vault, and it gets deleted accidentally or erroneously. Also, a malicious actor may delete your production backup items. It's often costly and time-intensive to rebuild those resources and can even cause crucial data loss. Azure Backup provides safeguard against accidental and malicious deletion with the [Soft-Delete](#) feature by allowing you to recover those resources after they are deleted.

With soft-delete, if a user deletes the backup (of a VM, SQL Server database, Azure file share, SAP HANA database), the backup data is retained for 14 additional days, allowing the recovery of that backup item with no data loss. The additional 14 days retention of backup data in the soft delete state doesn't incur any cost. [Learn more](#)

## Multi-User Authorization (MUA)

**How would you protect your data if your administrator goes rogue and compromises your system?**

Any administrator that has the privileged access to your backup data has the potential to cause irreparable damage to the system. A rogue admin can delete all your business-critical data or even turn off all the security measures that may leave your system vulnerable to cyber-attacks.

Azure Backup provides you with the [Multi-User Authorization \(MUA\)](#) feature to protect you from such rogue administrator attacks. Multi-user authorization helps protect against a rogue administrator performing destructive operations (that is, disabling soft-delete), by ensuring that every privileged/destructive operation is done only after getting approval from a security administrator.

## Ransomware Protection

- Direct access to Azure Backup data to encrypt by malicious actor is ruled out, as all operations on backup data can only be performed through Recovery-Services vault or Backup Vault, which can be secured by Azure role-based access control (Azure RBAC) and MUA.
- By enabling soft-delete on backup data (which is enabled by default) will hold deleted data for 14 days (at free of cost). Disabling soft-delete can be protected using MUA.
- Use longer retention (weeks, months, years) to ensure clean backups (not encrypted by ransomware) don't expire prematurely, and there're strategies in place for early detection and mitigation of such attacks on source data.

## Monitoring and alerts of suspicious activity

You may encounter scenarios where someone tries to breach into your system and maliciously turn off the security mechanisms, such as disabling Soft Delete or attempts to perform destructive operations, such as deleting the backup resources.

Azure Backup provides security against such incidents by sending you critical alerts over your preferred notification channel (email, ITSM, Webhook, runbook, and sp pn) by creating an [Action Rule](#) on top of the alert. [Learn more](#)

## Security features to help protect hybrid backups

Azure Backup service uses the Microsoft Azure Recovery Services (MARS) agent to back up and restore files, folders, and the volume or system state from an on-premises computer to Azure. MARS now provides security features: a passphrase to encrypt before upload and decrypt after download from Azure Backup, deleted backup data is retained for an additional 14 days from the date of deletion, and critical operation (ex. changing a passphrase) can be performed only by users who have valid Azure credentials. [Learn more here](#).

## Network considerations

Azure Backup requires movement of data from your workload to the Recovery Services vault. Azure Backup provides several capabilities to protect backup data from being exposed inadvertently (such as a man-in-the-middle attack on the network). Consider the following guidelines:

### Internet connectivity

- **Azure VM backup:** All the required communication and data transfer between storage and Azure Backup service happens within the Azure network without needing to access your virtual network. So backup of Azure VMs placed inside secured networks don't require you to allow access to any IPs or FQDNs.
- **SAP HANA databases on Azure VM, SQL Server databases on Azure VM:** Requires connectivity to the Azure Backup service, Azure Storage, and Azure Active Directory. This can be achieved by using private endpoints or by allowing access to the required public IP addresses or FQDNs. Not allowing proper connectivity to the required Azure services may lead to failure in operations like database discovery, configuring backup, performing backups, and restoring data. For complete network guidance while using NSG tags, Azure firewall, and HTTP Proxy, refer to these [SQL](#) and [SAP HANA](#) articles.
- **Hybrid:** The MARS (Microsoft Azure Recovery Services) agent requires network access for all critical operations - install, configure, backup, and restore. The MARS agent can connect to the Azure Backup service over [Azure ExpressRoute](#) by using

public peering (available for old circuits) and Microsoft peering, using [private endpoints](#) or via [proxy/firewall with appropriate access controls](#).

## Private Endpoints for secure access

While protecting your critical data with Azure Backup, you wouldn't want your resources to be accessible from the public internet. Especially, if you're a bank or a financial institution, you would have stringent compliance and security requirements to protect your High Business Impact (HBI) data. Even in the healthcare industry, there are strict compliance rules.

To fulfill all these needs, use [Azure Private Endpoint](#), which is a network interface that connects you privately and securely to a service powered by Azure Private Link. We recommend you to use private endpoints for secure backup and restore without the need to add to an allowlist of any IPs/FQDNs for Azure Backup or Azure Storage from your virtual networks.

[Learn more](#) about how to create and use private endpoints for Azure Backup inside your virtual networks.

- When you enable private endpoints for the vault, they're only used for backup and restore of SQL and SAP HANA workloads in an Azure VM, MARS agent, DPM/MABS backups. You can use the vault for the backup of other workloads as well (they won't require private endpoints though). In addition to the backup of SQL and SAP HANA workloads, backup using the MARS agent and DPM/MABS Server, private endpoints are also used to perform file recovery in the case of Azure VM backup. [Learn more here](#).
- Azure Active Directory doesn't currently support private endpoints. So, IPs and FQDNs required for Azure Active Directory will need to be allowed outbound access from the secured network when performing backup of databases in Azure VMs and backup using the MARS agent. You can also use NSG tags and Azure Firewall tags for allowing access to Azure AD, as applicable. Learn more about the [prerequisites here](#).

## Governance considerations

Governance in Azure is primarily implemented with [Azure Policy](#) and [Azure Cost Management](#). [Azure Policy](#) allows you to create, assign, and manage policy definitions to enforce rules for your resources. This feature keeps those resources in compliance with your corporate standards. [Azure Cost Management](#) allows you to track cloud usage and expenditures for your Azure resources and other cloud providers. Also, the

following tools such as [Azure Price Calculator](#) and [Azure Advisor](#) play an important role in the cost management process.

## Auto-configure newly provisioned backup infrastructure with Azure Policy at Scale

- Whenever new infrastructure is provisioned and new VMs are created, as a backup admin, you need to ensure their protection. You can easily configure backups for one or two VMs. But it becomes complex when you need to configure hundreds or even thousands of VMs at scale. To simplify the process of configuring backups, Azure Backup provides you with a set of built-in Azure Policies to govern your backup estate.
- **Auto-enable backup on VMs using Policy (Central backup team model):** If your organization has a central backup team that manages backups across application teams, you can use this policy to configure backup to an existing central Recovery Services vault in the same subscription and location as that of the VMs. You can choose to include/exclude VMs that contain a certain tag from the policy scope. [Learn more](#).
- **Auto-enable backup on VMs using Policy (where backup owned by application teams):** If you organize applications in dedicated resource groups and want to have them backed-up by the same vault, use this policy to automatically manage this action. You can choose to include/exclude VMs that contain a certain tag from the policy scope. [Learn more](#).
- **Monitoring Policy:** To generate the Backup Reports for your resources, enable the diagnostic settings when you create a new vault. Often, adding a diagnostic setting manually per vault can be a cumbersome task. So, you can utilize an Azure built-in policy that configures the diagnostics settings at scale to all vaults in each subscription or resource group, with Log Analytics as the destination.
- **Audit-only Policy:** Azure Backup also provides you with an Audit-only policy that identifies the VMs with no backup configuration.

## Azure Backup cost considerations

The Azure Backup service offers the flexibility to effectively manage your costs; also, meet your BCDR (business continuity and disaster recovery) business requirement. Consider the following guidelines:

- Use the pricing calculator to evaluate and optimize cost by adjusting various levers. [Learn more](#)
- Optimize backup policy,
  - Optimize schedule and retention settings based on workload archetypes (such as mission-critical, non-critical).
  - Optimize retention settings for Instant Restore.
  - Choose the right backup type to meet requirements, while taking supported backup types (full, incremental, log, differential) by the workload in Azure Backup.
- **Reduce the backup storage cost with Selectively backup disks:** Exclude disk (preview feature) provides an efficient and cost-effective choice to selectively backup critical data. For example, you can back up only one disk when you don't want to back up all disks attached to a VM. This is also useful when you have multiple backup solutions. For example, to back up your databases or data with a workload backup solution (SQL Server database in Azure VM backup), use Azure VM level backup for selected disks.
- **Speed up your Restores and minimize RTO using the Instant Restore feature:** Azure Backup takes snapshots of Azure VMs and stores them along with the disks to boost recovery point creation and to speed up restore operations. This is called Instant Restore. This feature allows a restore operation from these snapshots by cutting down the restore times. It reduces the time needed to transform and copy data back from the vault. Therefore, it'll incur storage costs for the snapshots taken during this period. Learn more about [Azure Backup Instant Recovery capability](#).
- **Choose correct replication type:** Azure Backup vault's Storage Replication type is set to Geo-redundant (GRS), by default. This option can't be changed after you start protecting items. Geo-redundant storage (GRS) provides a higher level of data durability than Locally redundant storage (LRS), allows an opt-in to use Cross Region Restore, and costs more. Review the trade-offs between lower costs and higher data durability and choose the best option for your scenario. [Learn more](#)
- **Use Archive Tier for Long-Term Retention (LTR) and save costs:** Consider the scenario where you've older backup data that you rarely access, but is required to be stored for a long period (for example, 99 years) for compliance reasons. Storing such huge data in a Standard Tier is costly and isn't economical. To help you optimize your storage costs, Azure Backup provides you with [Archive Tier](#), which is an access tier especially designed for Long-Term Retention (LTR) of the backup data.

- If you're protecting both the workload running inside a VM and the VM itself, ensure if this dual protection is needed.

## Monitoring and Alerting considerations

As a backup user or administrator, you should be able to monitor all backup solutions and get notified on important scenarios. This section details the monitoring and notification capabilities provided by the Azure Backup service.

### Monitor

- Azure Backup provides **built-in job monitoring** for operations such as configuring backup, back up, restore, delete backup, and so on. This is scoped to the vault, and ideal for monitoring a single vault. [Learn more here](#).
- If you need to monitor operational activities at scale, then **Backup Explorer** provides an aggregated view of your entire backup estate, enabling detailed drill-down analysis and troubleshooting. It's a built-in Azure Monitor workbook that gives a single, central location to help you monitor operational activities across the entire backup estate on Azure, spanning tenants, locations, subscriptions, resource groups, and vaults. [Learn more here](#).
  - Use it to identify resources that aren't configured for backup, and ensure that you don't ever miss protecting critical data in your growing estate.
  - The dashboard provides operational activities for the last seven days (maximum). If you need to retain this data, then you can export as an Excel file and retain them.
  - If you're an Azure Lighthouse user, you can view information across multiple tenants, enabling boundary-less monitoring.
- If you need to retain and view the operational activities for long-term, then use **Reports**. A common requirement for backup admins is to obtain insights on backups based on data that spans an extended period of time. Use cases for such a solution include:
  - Allocating and forecasting of cloud storage consumed.
  - Auditing of backups and restores.
  - Identifying key trends at different levels of granularity.
- In addition,
  - You can send data (for example, jobs, policies, and so on) to the **Log Analytics** workspace. This will enable the features of Azure Monitor Logs to enable correlation of data with other monitoring data collected by Azure Monitor,

consolidate log entries from multiple Azure subscriptions and tenants into one location for analysis together, use log queries to perform complex analysis and gain deep insights on Log entries. [Learn more here](#).

- You can send data to an Azure event hub to send entries outside of Azure, for example to a third-party SIEM (Security Information and Event Management) or other log analytics solution. [Learn more here](#).
- You can send data to an Azure Storage account if you want to retain your log data longer than 90 days for audit, static analysis, or back up. If you only need to retain your events for 90 days or less, you don't need to set up archives to a storage account, since Activity Log events are kept in the Azure platform for 90 days. [Learn more](#).

## Alerts

In a scenario where your backup/restore job failed due to some unknown issue. To assign an engineer to debug it, you would want to be notified about the failure as soon as possible. There could also be a scenario where someone maliciously performs a destructive operation, such as deleting backup items or turning off soft-delete, and you would require an alert message for such incident.

You can configure such critical alerts and route them to any preferred notification channel (email, ITSM, webhook, runbook, and so on). Azure Backup integrates with multiple Azure services to meet different alerting and notification requirements:

- **Azure Monitor Logs (Log Analytics):** You can configure your [vaults to send data to a Log Analytics workspace](#), write custom queries on the workspace, and configure alerts to be generated based on the query output. You can view the query results in tables and charts; also, export them to Power BI or Grafana. (Log Analytics is also a key component of the reporting/auditing capability described in the later sections).
- **Azure Monitor Alerts:** For certain default scenarios, such as backup failure, restore failure, backup data deletion, and so on, Azure Backup sends alerts by default that are surfaced using Azure Monitor, without the need for a user to set up a Log Analytics workspace.
- **Azure Backup provides an in-built alert** notification mechanism via e-mail for failures, warnings, and critical operations. You can specify individual email addresses or distribution lists to be notified when an alert is generated. You can also choose whether to get notified for each individual alert or to group them in an hourly digest and then get notified.

- These alerts are defined by the service and provide support for limited scenarios - backup/restore failures, Stop protection with retain data/Stop protection with delete data, and so on. [Learn more here](#).
- If a destructive operation such as stop protection with delete data is performed, an alert is raised and an email is sent to subscription owners, admins, and co-admins even if notifications are **not** configured for the Recovery Services vault.
- Certain workloads can generate high frequency of failures (for example, SQL Server every 15 minutes). To prevent getting overwhelmed with alerts raised for each failure occurrence, the alerts are consolidated. [Learn more here](#).
- The in-built alerts can't be customized and are restricted to emails defined in the Azure portal.
- If you need to **create custom alerts** (for example, alerts of successful jobs) then use Log Analytics. In Azure Monitor, you can create your own alerts in a Log Analytics workspace. Hybrid workloads (DPM/MABS) can also send data to LA and use LA to provide common alerts across workloads supported by Azure Backup.
- You can also get notifications through built-in Recovery Services vault **activity logs**. However, it supports limited scenarios and isn't suitable for operations such as scheduled backup, which aligns better with resource logs than with activity logs. To learn more about these limitations and how you can use Log Analytics workspace for monitoring and alerting at scale for all your workloads that are protected by Azure Backup, refer to this [article](#).

## Automatic Retry of Failed Backup Jobs

Many of the failure errors or the outage scenarios are transient in nature, and you can remediate by setting up the right Azure role-based access control (Azure RBAC) permissions or re-trigger the backup/restore job. As the solution to such failures is simple, that you don't need to invest time waiting for an engineer to manually trigger the job or to assign the relevant permission. Therefore, the smarter way to handle this scenario is to automate the retry of the failed jobs. This will highly minimize the time taken to recover from failures. You can achieve this by retrieving relevant backup data via Azure Resource Graph (ARG) and combine it with corrective [PowerShell/CLI procedure](#).

Watch the following video to learn how to re-trigger backup for all failed jobs (across vaults, subscriptions, tenants) using ARG and PowerShell.

<https://www.youtube-nocookie.com/embed/8dioCgHNb5w>

## Route Alerts to your preferred notification channel

While transient errors can be corrected, some persistent errors might require in-depth analysis, and retriggering the jobs may not be the viable solution. You may have your own monitoring/ticketing mechanisms to ensure such failures are properly tracked and fixed. To handle such scenarios, you can choose to route the alerts to your preferred notification channel (email, ITSM, Webhook, runbook, and so on) by creating an Action Rule on the alert.

Watch the following video to learn how to leverage Azure Monitor to configure various notification mechanisms for critical alerts.

<https://www.youtube-nocookie.com/embed/oYulJKEPOYY>

## Next steps

Read the following articles as starting points for using Azure Backup:

- [Azure Backup overview](#)
- [Frequently Asked Questions](#)

# Computer forensics chain of custody in Azure

Azure Automation

Azure Disk Encryption

Azure Key Vault

Azure Storage Accounts

This article describes an infrastructure and workflow process that helps teams ensure that the digital evidence they provide in response to legal requests demonstrates a valid *chain of custody* (CoC). This discussion helps ensure a valid CoC throughout the evidence acquisition, preservation, and access processes.

## ⓘ Note

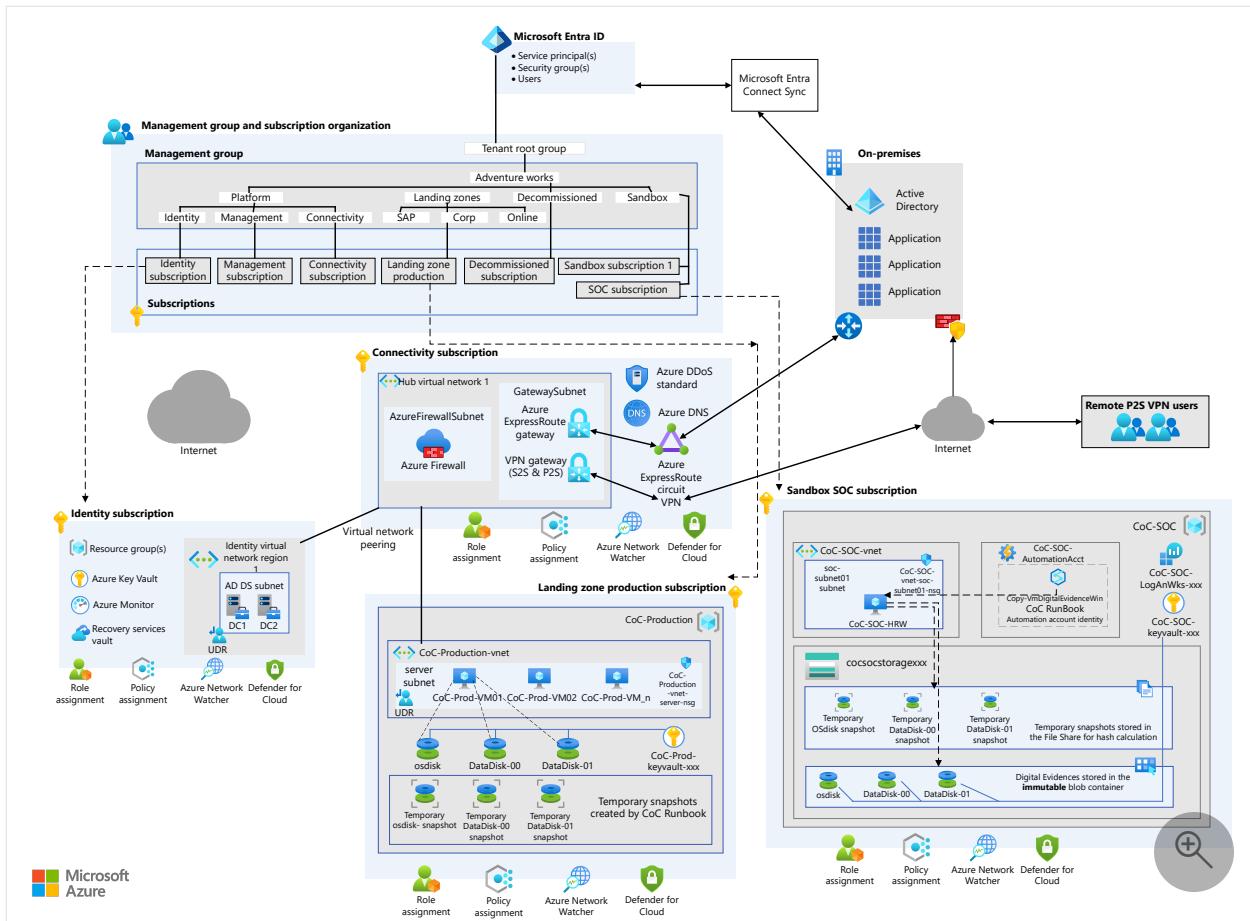
This article is based on the theoretical and practical knowledge of the authors.

Before you use it for legal purposes, you should validate its applicability with your legal department.

## Architecture

The architecture design follows the [Azure landing zone](#) principles that are described in the [Cloud Adoption Framework for Azure](#).

This scenario uses a hub-and-spoke network topology as shown in the following diagram:



Download a [Visio file](#) of this architecture.

## Workflow

In the architecture, the production virtual machines (VMs) are part of a spoke [Azure virtual network](#). Their disks are encrypted with Azure Disk Encryption. For more information, see [Overview of managed disk encryption options](#). In the production subscription, [Azure Key Vault](#) stores the VMs' BitLocker encryption keys (BEKs).

### ⓘ Note

The scenario works for production VMs with unencrypted disks.

The system and organization controls (SOC) team uses a discrete Azure [SOC](#) subscription. The team has exclusive access to that subscription, which contains the resources that must be kept protected, inviolable, and monitored. The [Azure Storage](#) account in the SOC subscription hosts copies of disk snapshots in [immutable blob storage](#), and a dedicated [key vault](#) keeps the snapshots' hash values and copies of the VMs' BEKs.

In response to a request to capture a VM's digital evidence, a SOC team member signs in to the Azure SOC subscription and uses a [hybrid runbook worker](#) VM in [Automation](#)

to implement the Copy-VmDigitalEvidence runbook. The [Automation hybrid runbook worker](#) provides control of all mechanisms involved in the capture.

The Copy-VmDigitalEvidence runbook implements these macro steps:

1. Sign in to Azure by using the [System-assigned managed identity for an Automation account](#) to access the target VM's resources and the other Azure services required by the solution.
2. Create disk snapshots for the VM's operating system (OS) and data disks.
3. Copy the snapshots to the SOC subscription's immutable blob storage, and in a temporary file share.
4. Calculate hash values of the snapshots by using the copy on the file share.
5. Copy the obtained hash values and the VM's BEK in the SOC key vault.
6. Clean up all the copies of the snapshots except the one in immutable blob storage.

#### ⓘ Note

The production VMs' encrypted disks can also use *key encryption keys* (KEKs). The Copy-VmDigitalEvidence runbook provided in the [deploy scenario](#) doesn't cover this scenario.

## Components

- [Azure Automation](#) automates frequent, time-consuming, and error-prone cloud management tasks.
- [Storage](#) is a cloud storage solution that includes object, file, disk, queue, and table storage.
- [Azure Blob Storage](#) provides optimized cloud object storage that manages massive amounts of unstructured data.
- [Azure Files](#) shares can be mounted concurrently by cloud or on-premises deployments of Windows, Linux, and macOS. Azure Files shares can also be cached on Windows Servers with Azure File Sync for fast access near where the data is being used.
- [Azure Monitor](#) supports your operations at scale by helping you maximize the performance and availability of your resources and proactively identify problems.
- [Key Vault](#) helps you safeguard cryptographic keys and other secrets used by cloud apps and services.
- [Microsoft Entra ID](#) is a cloud-based identity service that helps you control access to Azure and other cloud apps.

## Automation

The SOC team uses an [Automation](#) account to create and maintain the Copy-VmDigitalEvidence runbook. The team also uses [Automation](#) to create the hybrid runbook workers that run the runbook.

## Hybrid runbook worker

The [hybrid runbook worker](#) VM is part of the Automation account and is used exclusively by the SOC team to implement the Copy-VmDigitalEvidence runbook.

You must place the hybrid runbook worker VM in a subnet that can access the Storage account. Access to the Storage account is configured by adding the hybrid runbook worker VM subnet to the Storage account's firewall allowlist rules.

You must grant access to this VM only to the SOC team members for maintenance activities.

To isolate the virtual network that's used by the VM, don't connect that virtual network to the hub.

The hybrid runbook worker uses the [Automation system-assigned managed identity](#) to access the target VM's resources and the other Azure services required by the solution.

The minimal role-based access control (RBAC) permissions that must be assigned to system-assigned managed identity are classified in two categories:

- Access permissions to the SOC Azure architecture containing the solution core components
- Access permissions to the target architecture containing the target VM resources

Access to the SOC Azure architecture includes the following roles:

- **Storage Account Contributor** on the SOC immutable Storage account
- **Key Vault Secrets Officer** on the SOC key vault for the BEK management

Access to the target architecture includes the following roles:

- **Contributor** on the target VM's resource group, which provides snapshot rights on VM disks
- **Key Vault Secrets Officer** on the target VM's key vault used to store BEK, only if RBAC is used for the key vault
- Access policy to **Get Secret** on the target VM's key vault used to store BEK, only if you use an access policy for Key Vault

## (!) Note

To read the BEK, the target VM's key vault must be accessible from the hybrid runbook worker VM. If the key vault has the firewall enabled, ensure that the public IP address of the hybrid runbook worker VM is allowed through the firewall.

## Azure Storage account

The [Azure Storage account](#) in the SOC subscription hosts the disk snapshots in a container configured with a *legal hold* policy as Azure immutable blob storage. Immutable blob storage stores business-critical data objects in a *write once, read many* (WORM) state, which makes the data nonerasable and uneditable for a user-specified interval.

Ensure the [secure transfer](#) and [storage firewall](#) properties are both enabled. The firewall grants access only from the SOC virtual network.

The storage account also hosts an [Azure file share](#) used as a temporary repository for calculating the snapshot's hash value.

## Azure Key Vault

The SOC subscription has its own instance of [Key Vault](#), which hosts a copy of the BEK that Azure Disk Encryption uses to protect the target VM. The primary copy is kept in the key vault that is used by the target VM, so that the target VM can continue normal operation.

The SOC key vault also contains the hash values of disk snapshots calculated by the hybrid runbook worker during the capture operations.

Ensure the [firewall](#) is enabled on the key vault. It's configured to grant access only from the SOC virtual network.

## Log Analytics

A [Log Analytics workspace](#) stores activity logs used to audit all relevant events on the SOC subscription. Log Analytics is a feature of [Monitor](#).

## Scenario details

Digital forensics is a science that addresses the recovery and investigation of digital data to support criminal investigations or civil proceedings. Computer forensics is a branch of digital forensics that captures and analyzes data from computers, VMs, and digital storage media.

Companies must guarantee that the digital evidence they provide in response to legal requests demonstrates a valid CoC throughout the evidence acquisition, preservation, and access process.

## Potential use cases

- A company's Security Operation Center team can implement this technical solution to support a valid CoC for digital evidence.
- Investigators can attach disk copies that are obtained with this technique on a computer dedicated to forensic analysis. They can attach the disk copies without powering on or accessing the original source VM.

## CoC regulatory compliance

If it's necessary to submit the proposed solution to a regulatory compliance validation process, consider the topics in [considerations](#) during the CoC solution validation process.

### ⓘ Note

You should involve your legal department in the process of validation.

## Considerations

The principles that validate this solution as a Chain of Custody (CoC) are being presented in this section.

To ensure a valid CoC, digital evidence storage must demonstrate adequate access control, data protection and integrity, monitoring and alerting, and logging and auditing.

## Compliance with security standards and regulations

When you validate a CoC solution, one of the requirements to evaluate is the compliance with security standards and regulations.

All the components included in the [architecture](#) are Azure standard services built upon a foundation that supports trust, security, and [compliance](#) ↗.

Azure has a wide range of compliance certifications, including certifications specific for countries or regions, and for the key industries like healthcare, government, finance, and education.

For updated audit reports with information about standards compliance for the services that are adopted in this solution, see [Service Trust Portal](#) ↗.

Cohasset's [Azure Storage: SEC 17a-4\(f\) and CFTC 1.31\(c\)-\(d\) Compliance Assessment](#) ↗ gives details on the following requirements:

- Securities and Exchange Commission (SEC) in 17 CFR § 240.17a-4(f), which regulates exchange members, brokers, or dealers.
- Financial Industry Regulatory Authority (FINRA) Rule 4511(c), which defers to the format and media requirements of SEC Rule 17a-4(f).
- Commodity Futures Trading Commission (CFTC) in regulation 17 CFR § 1.31(c)-(d), which regulates commodity futures trading.

It's Cohasset's opinion that Storage, with the Immutable Storage for Azure blobs feature and policy lock option, retains *time-based* blobs (records) in a nonerasable and nonrewriteable format and meets relevant storage requirements of SEC Rule 17a-4(f), FINRA Rule 4511(c), and the principles-based requirements of CFTC Rule 1.31(c)-(d).

## Least privilege

When the roles of the SOC team are assigned, only two individuals within the team should have rights to modify the RBAC configuration of the subscription and its data. Grant other individuals only bare minimum access rights to data subsets they need to perform their work. Configure and enforce access through [Azure RBAC](#).

## Least access

Only the [virtual network](#) in the SOC subscription has access to the SOC Storage account and key vault that is used to archive the evidence.

Temporary access to the SOC storage is provided to investigators that require access to evidence. Authorized SOC team members can grant access.

## Evidence acquisition

Azure audit logs can show the evidence acquisition by recording the action of taking a VM disk snapshot, with elements like who has taken the snapshots and when.

## Evidence integrity

The use of Automation to move evidence to its final archive destination, without human intervention, guarantees that evidence artifacts haven't been altered.

When you apply a legal hold policy to the destination storage, the evidence is frozen in time as soon as it's written. A legal hold shows that the CoC has been maintained entirely in Azure. A legal hold also shows that there wasn't an opportunity to tamper between the time the disk images existed on a live VM and when they were added as evidence in the storage account.

Lastly, you can use the provided solution, as an integrity mechanism, to calculate the hash values of the disk images. The supported hash algorithms are: MD5, SHA256, SKEIN, KECCAK (or SHA3).

## Evidence production

Investigators need access to evidence to perform analyses, and this access must be tracked and explicitly authorized.

Provide investigators with a storage [shared access signatures \(SAS\) URI](#) key for accessing evidence. You can use an SAS URI to produce relevant log information when the SAS is generated. You can also get a copy of the evidence every time the SAS is used.

You must explicitly place the IP addresses of investigators requiring access on an allowlist in the Storage firewall.

For example, if a legal team needs to transfer a preserved virtual hard drive (VHD), one of the two SOC team custodians generates a read-only SAS URI key that expires after eight hours. The SAS limits the access to the IP addresses of the investigators to a specific time frame.

Finally, investigators need the BEKs archived in the SOC key vault to access the encrypted disk copies. An SOC team member must extract the BEKs and provide them via secure channels to the investigators.

## Regional store

For compliance, some standards or regulations require evidence and the support infrastructure to be maintained in the same Azure region.

All the solution components, including the Storage account used to archive evidence, are hosted in the same Azure region as the systems being investigated.

## Operational excellence

Operational excellence covers the operations processes that deploy an application and keep it running in production. For more information, see [Overview of the operational excellence pillar](#).

### Monitoring and alerting

Azure provides services to all customers to monitor and alert on anomalies involving their subscriptions and resources. These services include:

- [Microsoft Sentinel](#).
- [Microsoft Defender for Cloud](#).
- [Azure Storage Advanced Threat Protection \(ATP\)](#).

 **Note**

The configuration of these services isn't described in this article.

## Deploy this scenario

Follow the [CoC LAB Deployment](#) instructions to build and deploy this scenario in a laboratory environment.

The laboratory environment represents a simplified version of the architecture that is described in the article deploying two resource groups within the same subscription. The first resource group simulates the production environment, housing digital evidence, while the second resource group holds the SOC environment.

Use the following button to deploy only the SOC resource group in a production environment.



[Deploy to Azure](#)

### ⓘ Note

If you deploy the solution in a production environment, make sure that the system-assigned managed identity of the automation account has following permissions:

- Contributor: in the production resource group of the VM to be processed (needed to create the snapshots)
- Key Vault Secrets User: in the production key vault holding the BEK keys (needed to read the BEK keys)

Additionally, if the key vault has the firewall enabled, ensure that the public IP address of the hybrid runbook worker VM is allowed through the firewall.

## Extended configuration

You can deploy a hybrid runbook worker on-premises or in different cloud environments.

In this scenario, you can customize the `Copy-VmDigitalEvidence` runbook to enable the capture of evidence in different target environments and archive them in storage.

### ⓘ Note

The `Copy-VmDigitalEvidence` runbook provided in the [Deploy this scenario](#) section has been developed and tested only in Azure. To extend the solution to other platforms, you must customize the runbook to work with those platforms.

## Contributors

*This article is maintained by Microsoft. It was originally written by the following contributors.*

Principal author:

- [Fabio Masciotra](#) | Principal Consultant
- [Simone Savi](#) | Senior Consultant

*To see non-public LinkedIn profiles, sign in to LinkedIn.*

## Next steps

For more information about Azure data-protection features, see:

- [Azure Storage encryption for data at rest](#)
- [Overview of managed disk encryption options](#)
- [Store business-critical blob data with immutable storage](#)

For more information about Azure logging and auditing features, see:

- [Azure security logging and auditing](#)
- [Azure Storage analytics logging](#)
- [Azure resource logs](#)

For more information about Microsoft Azure Compliance, see:

- [Azure compliance ↗](#)
- [Microsoft Azure Compliance Offerings ↗](#)

## Related resources

- [Security architecture design](#)
- [Microsoft Entra IDaaS in security operations](#)
- [Security considerations for highly sensitive IaaS apps in Azure](#)

# End-to-end governance in Azure when using CI/CD

Microsoft Entra ID

Azure DevOps

Azure Pipelines

Azure Resource Manager

When developing a governance model for your organization, it's important to remember that [Azure Resource Manager](#) is only one way to manage resources. Azure DevOps and continuous integration and continuous delivery (CI/CD) automation can be an unintentional security back-door if not properly secured. These resources should be protected by mirroring the role-based access control (RBAC) model used for Resource Manager.

The concept of end-to-end governance is vendor agnostic. The implementation described here uses [Azure DevOps](#), but alternatives are also briefly mentioned.

## Potential use cases

This reference implementation and demo is open source and intended to be used as a teaching tool for organizations who are new to DevOps and need to create a governance model for deploying to Azure. Please read this scenario carefully to understand the decisions behind the model used in this sample repository.

Any governance model must be tied to the organization's business rules, which are reflected in any technical implementation of access controls. This example model uses a fictitious company with the following common scenario (with business requirements):

- **Microsoft Entra groups that align with business domains and permissions models**

The organization has many vertical business domains, such as "fruits" and "vegetables," which operate largely independently. In each business domain, there are two levels of privileges, which are mapped to distinct `*-admins` or `*-devs` Microsoft Entra groups. This allows developers to be targeted when configuring permissions in the cloud.

- **Deployment environments**

Every team has two environments:

- Production. Only admins have elevated privileges.
- Non-production. All developers have elevated privileges (to encourage experimentation and innovation).

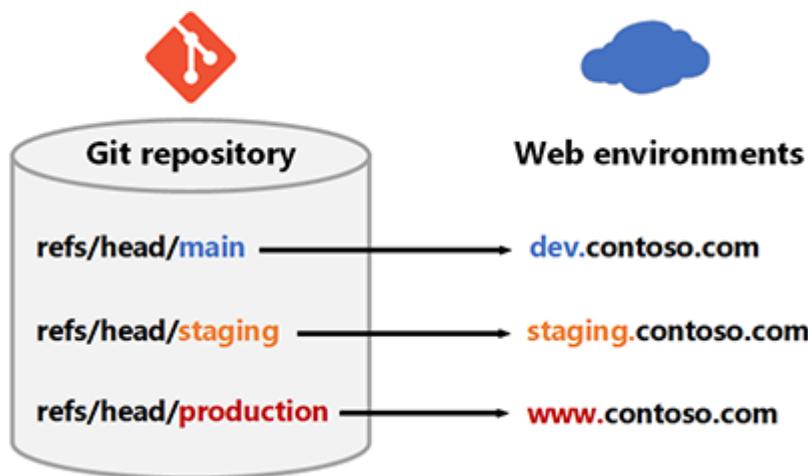
- **Automation goals**

Every application should implement Azure DevOps not just for continuous integration (CI), but also for continuous deployment (CD). For example, deployments can be automatically triggered by changes to the Git repository.

- **Cloud journey so far**

The organization started with an isolated project model to accelerate the journey to the cloud. But now they are exploring options to break silos and encourage collaboration by creating the "collaboration" and "supermarket" projects.

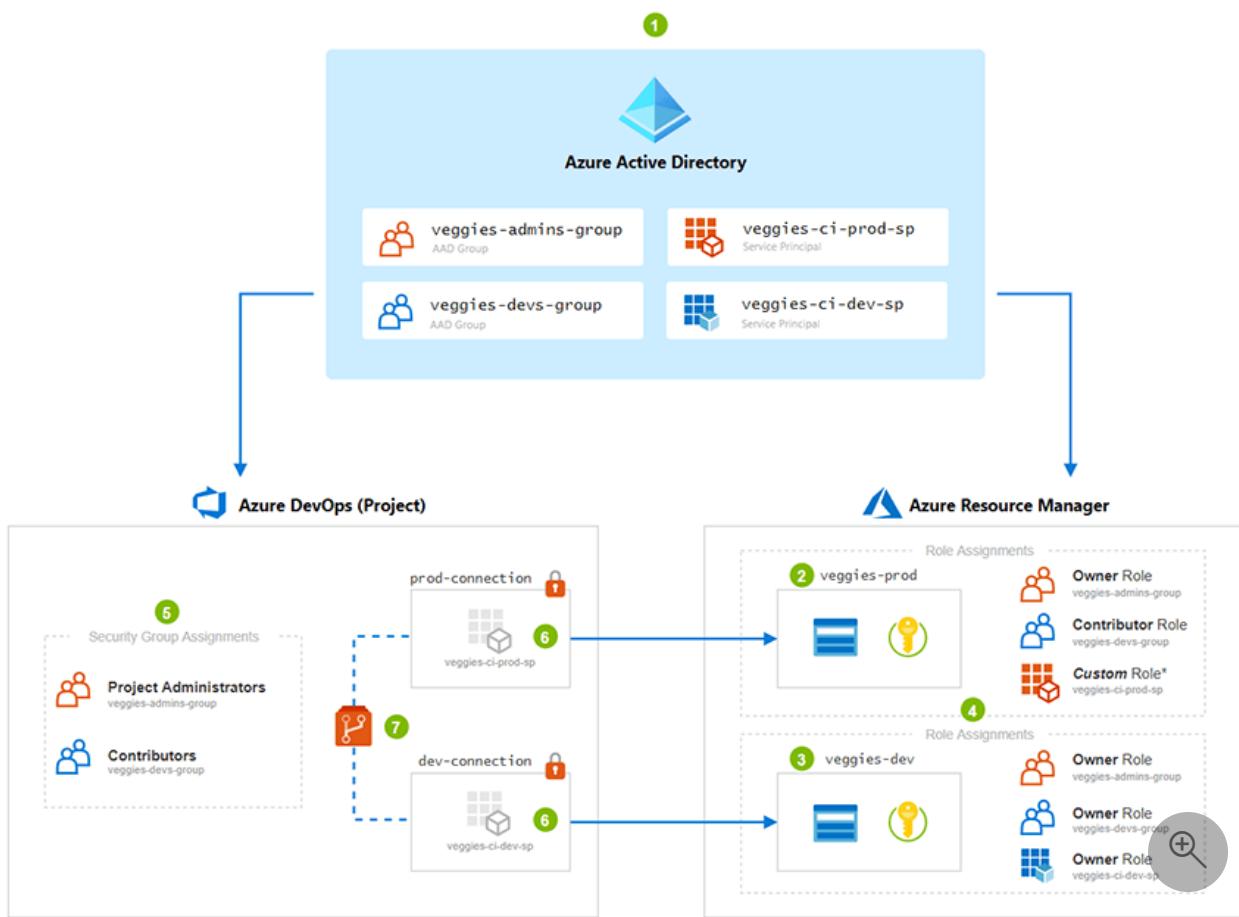
This simplified diagram shows how branches in a Git repository map to development, staging, and production environments:



*Download an [SVG of this diagram](#).*

## Architecture

This diagram shows how linking from Resource Manager and CI/CD to Microsoft Entra ID is the key to having an end-to-end governance model.



[Download an \*SVG\* of this architecture.](#)

### ⓘ Note

To make the concept easier to understand, the diagram only illustrates the "veggies" domain. The "fruits" domain would look similar and use the same naming conventions.

## Workflow

The numbering reflects the order in which IT administrators and enterprise architects think about and configure their cloud resources.

### 1. Microsoft Entra ID

We integrate Azure DevOps with [Microsoft Entra ID](#) in order to have a single plane for identity. This means a developer uses the same Microsoft Entra account for both Azure DevOps and Resource Manager. Users are not added individually. Instead, membership is assigned by Microsoft Entra groups so that we can remove a developer's access to resources in a single step—by removing their Microsoft Entra group memberships. For **each domain**, we create:

- Microsoft Entra groups. Two groups per domain (described further in step 4 and 5 later in this article).
- Service principals. One explicit service principal **per environment**.

## 2. Production environment

To simplify deployment, this reference implementation uses a resource group to represent the production environment. In practice, you should use a [different subscription](#).

Privileged access to this environment is limited to administrators only.

## 3. Development environment

To simplify deployment, this reference implementation uses a resource group to represent the development environment. In practice, you should use a [different subscription](#).

## 4. Role assignments in Resource Manager

Although our Microsoft Entra group names imply a role, access controls are not applied until a [role assignment](#) is configured. This assigns a role to a Microsoft Entra principal for a specific scope. For example, developers have the Contributor role on the production environment.

[ ] [Expand table](#)

Microsoft Entra principal	Dev environment (Resource Manager)	Production environment (Resource Manager)
veggies-devs-group	Owner	Reader
veggies-admins-group	Owner	Owner
veggies-ci-dev-sp	Custom Role *	—
veggies-ci-prod-sp	—	Custom Role *

\* To simplify deployment, this reference implementation assigns the `Owner` role to the service principals. However, in production you should create a *custom role* that prevents a service principal from removing any [management locks](#) that you've placed on your resources. This helps protect resources from accidental damage, such as database deletion.

To understand the reasoning behind the individual role assignments, see the [considerations section](#) later in this article.

## 5. Security group assignments in Azure DevOps

Security groups function like roles in Resource Manager. Take advantage of built-in roles and default to [Contributor](#) for developers. Admins get assigned to the [Project Administrator](#) security group for elevated permissions, allowing them to configure security permissions.

Note that Azure DevOps and Resource Manager have **different** permissions models:

- Azure Resource Manager uses an [additive permissions](#) model.
- Azure DevOps uses a [least permissions](#) model.

For this reason, membership to the `-admins` and `-devs` groups must be mutually exclusive. Otherwise, the affected persons would have less access than expected in Azure DevOps.

Expand table

Group name	Resource Manager role	Azure DevOps role
<code>fruits-all</code>	–	–
<code>fruits-devs</code>	Contributor	Contributor
<code>fruits-admins</code>	Owner	Project Administrators
<code>veggies-all</code>	–	–
<code>veggies-devs</code>	Contributor	Contributor
<code>veggies-admins</code>	Owner	Project Administrators

Group name	Resource Manager role	Azure DevOps role
infra-all	–	–
infra-devs	Contributor	Contributor
infra-admins	Owner	Project Administrators

In a scenario of limited collaboration, such as the fruits team inviting the veggies team to collaborate on a **single** repository, they would use the `veggies-all` group.

To understand the reasoning behind the individual role assignments, refer to the [considerations section](#) later in this article.

## 6. Service connections

In Azure DevOps, a [Service Connection](#) is a generic wrapper around a credential. We create a service connection that holds the service principal client ID and client secret. Project Administrators can configure access to this [protected resource](#) when needed, such as when requiring human approval before deploying. This reference architecture has two minimum protections on the service connection:

- Admins must configure [pipeline permissions](#) to control which pipelines can access the credentials.
- Admins must also configure a [branch control check](#) so that only pipelines running in the context of the `production` branch might use the `prod-connection`.

## 7. Git repositories

Because our service connections are tied to branches via [branch controls](#), it's critical to configure permissions to the Git repositories and apply [branch policies](#). In addition to requiring CI builds to pass, we also require pull requests to have at least two approvers.

# Components

- [Azure DevOps](#) ↗
- [Microsoft Entra ID](#) ↗
- [Azure Resource Manager](#) ↗
- [Azure Repos](#) ↗
- [Azure Pipelines](#) ↗

# Alternatives

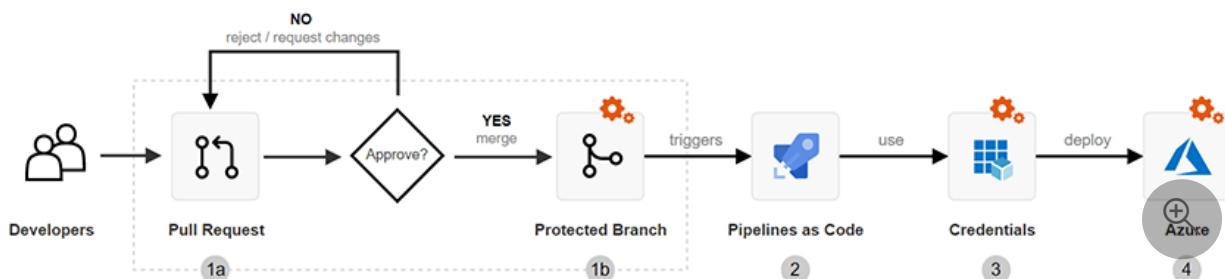
The concept of end-to-end governance is vendor agnostic. While this article focuses on Azure DevOps, [Azure DevOps Server](#) could be used as an on-premises substitute. Alternatively, you could also use a set of technologies for an open-source CI/CD development pipeline using options like [Jenkins](#) and [GitLab](#).

Both Azure Repos and GitHub are platforms that are built to use the open-source Git version control system. While their feature sets are somewhat different, both can be integrated into global governance models for CI/CD. GitLab is another Git-based platform that provides robust CI/CD capabilities.

This scenario uses Terraform as its infrastructure as code (IaC) tool. Alternatives include Jenkins, [Ansible](#), and [Chef](#).

# Considerations

To achieve end-to-end governance in Azure, it's important to understand the security and permissions profile of the path from developer's computer to production. The following diagram illustrates a baseline CI/CD workflow with Azure DevOps. The red lock icon  indicates security permissions that must be configured by the user. Not configuring or misconfiguring permissions will leave your workloads vulnerable.



[Download an SVG of this workflow.](#)

To successfully secure your workloads, you must use a combination of security permission configurations and human checks in your workflow. It's important that any RBAC model must also extend to both pipelines and code. These often run with privileged identities and will destroy your workloads if instructed to do so. To prevent this from happening, you should configure [branch policies](#) on your repository to require human approval before accepting changes that trigger automation pipelines.

[\[ \] Expand table](#)

Deployment stages	Responsibility	Description
Pull requests	User	Engineers should peer review their work, including the Pipeline code itself.
Branch protection	Shared	Configure <a href="#">Azure DevOps</a> to reject changes that do not meet certain standards, such as CI checks and peer reviews (via pull requests).
Pipeline as code	User	A build server will delete your entire production environment if the pipeline code instructs it to do so. Help prevent this by using a combination of pull requests and branch protection rules, such as human approval.
Service connections	Shared	Configure Azure DevOps to restrict access to these credentials.
Azure Resources	Shared	Configure RBAC in Resource Manager.

The following concepts and questions are important to consider when designing a governance model. Bear in mind the [potential use cases](#) of this example organization.

## 1. Safeguard your environments with branch policies

Because your source code defines and triggers deployments, your first line of defense is to secure your source code management (SCM) repository. In practice, this is achieved by using the [Pull Request workflow](#) in combination with [branch policies](#), which define checks and requirements before code can be accepted.

When planning your end-to-end governance model, privileged users (`veggies-admins`) will be responsible for configuring branch protection. Common branch protection checks used to secure your deployments include:

- **Require CI build to pass.** Useful for establishing baseline code quality, such as code linting, unit tests, and even security checks like virus and credential scans.
- **Require peer review** Have another human double check that code works as intended. Be extra careful when changes are made to pipeline code. Combine with CI builds to make peer reviews less tedious.

## What happens if a developer tries to push directly to production?

Remember that Git is a distributed SCM system. A developer can commit directly to their local `production` branch. But when Git is properly configured, such a push will be automatically rejected by the Git server. For example:

PowerShell

```
remote: Resolving deltas: 100% (3/3), completed with 3 local objects.
remote: error: GH006: Protected branch update failed for refs/heads/main.
remote: error: Required status check "continuous-integration" is expected.
To https://github.com/Azure/devops-governance
 ! [remote rejected] main -> main (protected branch hook declined)
error: failed to push some refs to 'https://github.com/Azure/devops-
governance'
```

Note that the workflow in the example is vendor agnostic. The pull request and branch protection features are available from multiple SCM providers, including [Azure Repos](#), [GitHub](#), and [GitLab](#).

Once the code has been accepted into a protected branch, the next layer of access controls will be applied by the build server (such as [Azure Pipelines](#)).

## 2. What access do security principals need?

In Azure, a [security principal](#) can be either a *user principal* or a *headless principal*, such as a service principal or managed identity. In all environments, security principals should follow the [principle of least privilege](#). While security principals might have expanded access in pre-production environments, production Azure environments should minimize standing permissions, favoring just-in-time (JIT) access and Microsoft Entra Conditional Access. Craft your Azure RBAC role assignments for user principals to align with these least privilege principals.

It's also important to model Azure RBAC distinctly from Azure DevOps RBAC. The purpose of the pipeline is to minimize direct access to Azure. Except for special cases like innovation, learning, and issue resolution, most interactions with Azure should be conducted through purpose-built and gated pipelines.

For Azure Pipeline service principals, consider using a [custom role](#) that prevents it from removing resource locks and performing other destructive actions out of scope for its purpose.

### 3. Create a custom role for the service principal used to access production

It's a common mistake to give CI/CD build agents Owner roles and permissions. Contributor permissions are not enough if your pipeline also needs to perform identity role assignments or other privileged operations like Key Vault policy management.

But a CI/CD Build Agent will delete your entire production environment if told to do so. To avoid **irreversible destructive changes**, we create a custom role that:

- Removes Key Vault access policies
- Removes **management locks** that by design should prevent resources from being deleted (a common requirement in regulated industries)

To do this, we create a custom role and remove the `Microsoft.Authorization/*/Delete` actions.

```
JSON

{
  "Name": "Headless Owner",
  "Description": "Can manage infrastructure.",
  "actions": [
    "*"
  ],
  "notActions": [
    "Microsoft.Authorization/*/Delete"
  ],
  "AssignableScopes": [
    "/subscriptions/{subscriptionId1}",
    "/subscriptions/{subscriptionId2}",
    "/providers/Microsoft.Management/managementGroups/{groupId1}"
  ]
}
```

If that removes too many permissions for your purposes, refer to the full list in the [official documentation for Azure resource provider operations](#) and adjust your role definition as needed.

## Deploy this scenario

This scenario extends beyond Resource Manager. This is why we use [Terraform](#), which allows us to also create principals in Microsoft Entra ID and bootstrap Azure DevOps using a single infrastructure as code tool.

For source code and detailed instructions, visit the GitHub repository [Governance on Azure Demo - from DevOps to ARM](#).

## Pricing

Azure DevOps costs depend on the number of users in your organization that require access, along with other factors like the number of concurrent build/releases required and number of users. Azure Repos and Azure Pipelines are features of the Azure DevOps service. For more information, see [Azure DevOps pricing](#).

In Microsoft Entra ID, the type of group access management needed for this scenario is provided in the Premium P1 and Premium P2 editions. Pricing for these tiers is calculated on a per-user basis. For more information, see [Microsoft Entra pricing](#).

## Contributors

*This article is maintained by Microsoft. It was originally written by the following contributors.*

Principal author:

- [Julie Ng](#) | Senior Service Engineer

## Next steps

- Visit the code repository for this scenario at [Governance on Azure Demo - from DevOps to ARM](#).
- Review the Cloud Adoption Framework's [Cloud governance guides](#).
- [What is Azure role-based access control \(Azure RBAC\)?](#)
- [Cloud Adoption Framework: Resource access management in Azure](#)
- [Azure Resource Manager roles](#)
  - [Owner \(built-in\)](#)
  - [Contributor \(built-in\)](#)
  - [Reader \(built-in\)](#)
  - [Custom Role](#)
- [Azure DevOps security groups](#)
  - [Project Administrators](#)
  - [Contributor](#)
  - [Reader](#)

# Related resources

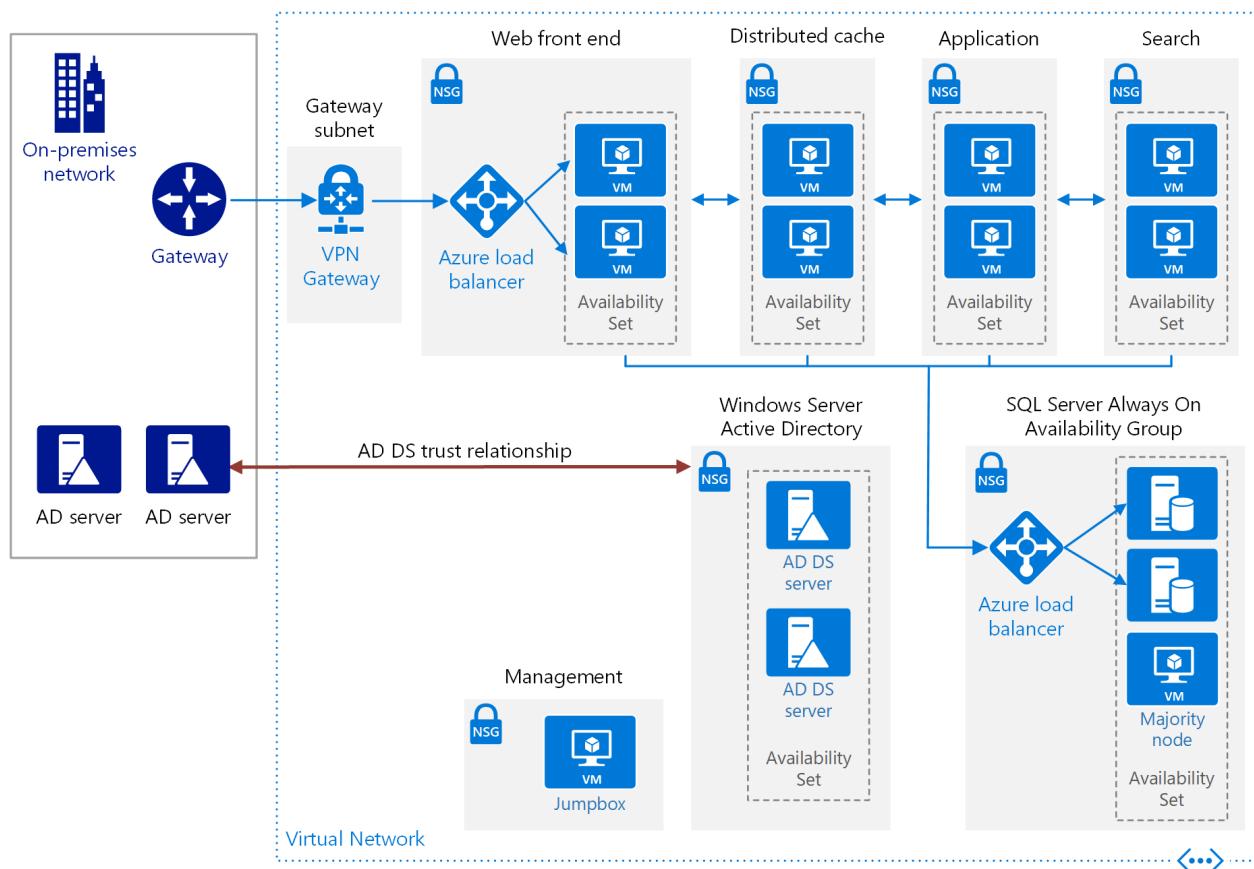
- [Design a CI/CD pipeline using Azure DevOps](#)
- [Computer forensics Chain of Custody in Azure](#)
- [Azure Arc hybrid management and deployment for Kubernetes clusters](#)
- [Azure Automation in a hybrid environment](#)
- [Azure Automation Update Management](#)
- [Browse Azure Architectures - CI/CD](#)

# Run a highly available SharePoint Server 2016 farm in Azure

Azure ExpressRoute   Azure Managed Disks   Azure Virtual Machines   Azure Virtual Network  
Azure VPN Gateway

This reference architecture shows proven practices for deploying a highly available SharePoint Server 2016 farm on Azure, using MinRole topology and SQL Server Always On availability groups. The SharePoint farm is deployed in a secured virtual network with no internet-facing endpoint or presence.

## Architecture



Download a [Visio file](#) of this architecture.

This architecture builds on the one shown in [Run Windows VMs for an N-tier application][windows-n-tier]. It deploys a SharePoint Server 2016 farm with high availability inside an Azure virtual network. This architecture is suitable for a test or production environment, a SharePoint hybrid infrastructure with Microsoft 365, or as the basis for a disaster recovery scenario.

# Components

- [Resource groups](#) are containers that hold related Azure resources. One resource group is used for the SharePoint servers, and another resource group is used for infrastructure components that are independent of virtual machines (VMs), such as the virtual network and load balancers.
- [Azure Virtual Network](#) is the fundamental building block for private networks in Azure. The VMs are deployed in a virtual network with a unique intranet address space. The virtual network is further subdivided into subnets.
- [VMs](#) are on-demand, scalable computing resources that are available with Azure. The VMs are deployed into the virtual network, and private static IP addresses are assigned to all the VMs. Static IP addresses are recommended for the VMs running SQL Server and SharePoint Server 2016, to avoid issues with IP address caching and changes of addresses after a restart.
- [Availability sets](#) are logical groupings of VMs. Place the VMs for each SharePoint role into separate availability sets, and provision at least two VMs for each role. This configuration makes the VMs eligible for a higher service level agreement (SLA).
- [Azure Load Balancer](#) distributes SharePoint request traffic from the on-premises network to the front-end web servers of the SharePoint farm. This solution uses an internal load balancer. Alternative for HTTP traffic, [Azure Application Gateway could be used](#).
- [Network security groups](#) filter traffic in an Azure virtual network. For each subnet that contains VMs, a network security group is created. Use network security groups to restrict network traffic within a virtual network, in order to isolate subnets.
- [Azure ExpressRoute](#) or a site-to-site VPN such as [Azure VPN Gateway](#) provides a connection between your on-premises network and the Azure virtual network. For more information, see [Connect an on-premises network to Azure](#).
- [Windows Server Active Directory \(AD\) domain controllers](#) authenticate users within a domain. The domain controllers in this reference architecture run in the Azure virtual network and have a trust relationship with the on-premises Windows Server AD forest. Client web requests for SharePoint farm resources are authenticated in the virtual network rather than sending that authentication traffic across the gateway connection to the on-premises network. In DNS, intranet A or CNAME

records are created so that intranet users can resolve the name of the SharePoint farm to the private IP address of the internal load balancer.

SharePoint Server 2016 also supports using [Microsoft Entra Domain Services](#).

Microsoft Entra Domain Services provides managed domain services so that you don't need to deploy and manage domain controllers in Azure.

- [SQL Server Always On availability groups](#) provide a high-availability and disaster-recovery solution. We recommend them for high availability of the SQL Server database. Two VMs are used for SQL Server. One contains the primary database replica, and the other contains the secondary replica.
- A majority node VM allows the failover cluster to establish a quorum. For more information, see [Understanding Quorum Configurations in a Failover Cluster](#).
- [SharePoint Server](#) provides a platform for shared access. The SharePoint servers perform the web front-end, caching, application, and search roles.
- A jump box, which is also called a [bastion host](#), is a secure VM on the network that administrators use to connect to the other VMs. The jump box has a network security group that allows remote traffic only from public IP addresses on a safe list. The network security group should permit remote desktop (RDP) traffic.

## Recommendations

Your requirements might differ from the architecture described here. Use these recommendations as a starting point.

### Resource group recommendations

We recommend separating resource groups according to the server role, and having a separate resource group for infrastructure components that are global resources. In this architecture, the SharePoint resources form one group, while the SQL Server and other utility assets form another.

### Virtual network and subnet recommendations

Use one subnet for each SharePoint role, plus a subnet for the gateway and one for the jump box.

The gateway subnet must be named *GatewaySubnet*. Assign the gateway subnet address space from the last part of the virtual network address space. For more

information, see [Connect an on-premises network to Azure using a VPN gateway](#).

## VM recommendations

This architecture requires a minimum of 44 cores:

- Eight SharePoint servers on Standard\_DS3\_v2 (4 cores each) = 32 cores
- Two Active Directory domain controllers on Standard\_DS1\_v2 (1 core each) = 2 cores
- Two SQL Server VMs on Standard\_DS3\_v2 = 8 cores
- One majority node on Standard\_DS1\_v2 = 1 core
- One management server on Standard\_DS1\_v2 = 1 core

For all SharePoint roles except the Search Indexer, we recommend using the [Standard\\_DS3\\_v2](#) VM size. The Search Indexer should be at least the [Standard\\_DS13\\_v2](#) size. For more information, see [Hardware and software requirements for SharePoint Server 2016](#). For the SQL Server VMs, we recommend a minimum of 4 cores and 8 GB of RAM. For more information, see [Storage and SQL Server capacity planning and configuration \(SharePoint Server\)](#).

## Network security group recommendations

We recommend having one network security group for each subnet that contains VMs, to enable subnet isolation. If you want to configure subnet isolation, add network security group rules that define the allowed or denied inbound or outbound traffic for each subnet. For more information, see [Filter network traffic with network security groups](#).

Don't assign a network security group to the gateway subnet, or the gateway will stop functioning.

## Storage recommendations

The storage configuration of the VMs in the farm should match the appropriate best practices used for on-premises deployments. SharePoint servers should have a separate disk for logs. SharePoint servers hosting search index roles require additional disk space for the search index to be stored. For SQL Server, the standard practice is to separate data and logs. Add more disks for database backup storage, and use a separate disk for [tempdb](#).

For best reliability, we recommend using [Azure Managed Disks](#). Managed disks ensure that the disks for VMs within an availability set are isolated to avoid single points of

failure.

Use Premium managed disks for all SharePoint and SQL Server VMs. You can use Standard managed disks for the majority node server, the domain controllers, and the management server.

## SharePoint Server recommendations

Before configuring the SharePoint farm, make sure you have one Windows Server Active Directory service account per service. For this architecture, you need, at a minimum, the following domain-level accounts to isolate privilege per role:

- SQL Server Service account
- Setup User account
- Server Farm account
- Search Service account
- Content Access account
- Web App Pool accounts
- Service App Pool accounts
- Cache Super User account
- Cache Super Reader account

To meet the support requirement for disk throughput of 200 MB per second minimum, make sure to plan the Search architecture. See [Plan enterprise search architecture in SharePoint Server 2013](#). Also, follow the guidelines in [Best practices for crawling in SharePoint Server 2016](#).

In addition, store the search component data on a separate storage volume or partition with high performance. To reduce load and improve throughput, configure the object cache user accounts, which are required in this architecture. Split the Windows Server operating system files, the SharePoint Server 2016 program files, and diagnostics logs across three separate storage volumes or partitions with normal performance.

For more information about these recommendations, see [Initial deployment administrative and service accounts in SharePoint Server 2016](#).

## Hybrid workloads

This reference architecture deploys a SharePoint Server 2016 farm that can be used as a [SharePoint hybrid environment](#)—that is, extending SharePoint Server 2016 to SharePoint Online. If you have Office Online Server, see [Office Web Apps and Office Online Server supportability in Azure](#).

The default service applications in this solution are designed to support hybrid workloads. All SharePoint Server 2016 and Microsoft 365 hybrid workloads can be deployed to this farm without changes to the SharePoint infrastructure, with one exception: The Cloud Hybrid Search Service Application must not be deployed onto servers hosting an existing search topology. Therefore, one or more search-role-based VMs must be added to the farm to support this hybrid scenario.

## SQL Server Always On availability groups

This architecture uses SQL Server VMs because SharePoint Server 2016 can't use Azure SQL Database. To support high availability in SQL Server, we recommend using Always On availability groups, which specify a set of databases that fail over together, making them highly available and recoverable. For more information, see [Create the availability group and add the SharePoint databases](#).

We also recommend adding a listener IP address to the cluster, which is the private IP address of the internal load balancer for the SQL Server VMs.

For recommended VM sizes and other performance recommendations for SQL Server running in Azure, see [Performance best practices for SQL Server in Azure Virtual Machines](#). Also follow the recommendations in [Best practices for SQL Server in a SharePoint Server 2016 farm](#).

We recommend that the majority node server resides on a separate computer from the replication partners. The server enables the secondary replication partner server in a high-safety mode session to recognize whether to initiate an automatic failover. Unlike the two partners, the majority node server doesn't serve the database but rather supports automatic failover.

## Considerations

These considerations implement the pillars of the Azure Well-Architected Framework, which is a set of guiding tenets that can be used to improve the quality of a workload. For more information, see [Microsoft Azure Well-Architected Framework](#).

## Scalability

To scale up the existing servers, change the VM size.

With the [MinRoles](#) capability in SharePoint Server 2016, you can scale out servers based on the server's role and also remove servers from a role. When you add servers to a role,

you can specify any of the single roles or one of the combined roles. If you add servers to the Search role, however, you must also reconfigure the search topology using PowerShell. You can also convert roles using MinRoles. For more information, see [Managing a MinRole Server Farm in SharePoint Server 2016](#).

SharePoint Server 2016 doesn't support using Virtual Machine Scale Sets for autoscaling.

## Availability

This reference architecture supports high availability within an Azure region because each role has at least two VMs deployed in an availability set.

To protect against a regional failure, create a separate disaster recovery farm in a different Azure region. Your recovery time objectives (RTOs) and recovery point objectives (RPOs) will determine the setup requirements. For details, see [Choose a disaster recovery strategy for SharePoint 2016](#). The secondary region should be a *paired region* with the primary region. In the event of a broad outage, recovery of one region is prioritized out of every pair. For more information, see [Business continuity and disaster recovery \(BCDR\): Azure Paired Regions](#).

## Manageability

To operate and maintain servers, server farms, and sites, follow the recommended practices for SharePoint operations. For more information, see [Operations for SharePoint Server 2016](#).

The tasks to consider when managing SQL Server in a SharePoint environment may differ from the ones typically considered for a database application. A best practice is to fully back up all SQL databases weekly with incremental nightly backups. Back up transaction logs every 15 minutes. Another practice is to implement SQL Server maintenance tasks on the databases while disabling the built-in SharePoint ones. For more information, see [Storage and SQL Server capacity planning and configuration](#).

## Security

Security provides assurances against deliberate attacks and the abuse of your valuable data and systems. For more information, see [Overview of the security pillar](#).

The domain-level service accounts used to run SharePoint Server 2016 require Windows Server AD domain controllers or Microsoft Entra Domain Services for domain-join and authentication processes. However, to extend the Windows Server AD identity infrastructure already in place in the intranet, this particular architecture uses two VMs

as Windows Server AD replica domain controllers of an existing on-premises Windows Server AD forest.

In addition, it's always wise to plan for security hardening. Other recommendations include:

- Add rules to network security groups to isolate subnets and roles.
- Don't assign public IP addresses to VMs.
- For intrusion detection and analysis of payloads, consider using a network virtual appliance in front of the front-end web servers instead of an internal Azure load balancer.
- As an option, use IPsec policies for encryption of cleartext traffic between servers. If you're also doing subnet isolation, update your network security group rules to allow IPsec traffic.
- Install anti-malware agents for the VMs.

## Cost optimization

Cost optimization is about looking at ways to reduce unnecessary expenses and improve operational efficiencies. For more information, see [Overview of the cost optimization pillar](#).

Use the [Azure pricing calculator](#) to estimate costs. Here are some factors for optimizing the cost for this architecture.

## Active Directory Domain Services

Consider having Active Directory Domain Services as a shared service that is consumed by multiple workloads to lower costs. For more information, see [Active Directory Domain Services pricing](#) for more information.

## VPN Gateway

The billing model is based on the amount of time the gateway is provisioned and available. See [VPN Gateway pricing](#).

All inbound traffic is free. All outbound traffic is billed. Internet bandwidth costs are applied to VPN outbound traffic.

## Virtual Network

Virtual Network is free. Every subscription is allowed to create up to 50 virtual networks across all regions. All traffic that originates within the boundaries of a virtual network is free. So, communication between two VMs in the same virtual network is free.

This architecture builds on the architecture deployed in [Run Windows VMs for an N-tier application][windows-n-tier].

For more information, see the cost section in [Microsoft Azure Well-Architected Framework](#).

## DevOps

Consider using separate resource groups for production, development, and test environments. Separate resource groups make it easier to manage deployments, delete test deployments, and assign access rights. In general, put resources that have the same lifecycle in the same resource group. Use the Developer tier for development and test environments. To minimize costs during preproduction, deploy a replica of your production environment, run your tests, and then shut down.

Use Azure [Azure Resource Manager templates](#) or Azure Bicep templates for defining the infrastructure. In both cases, you follow the infrastructure as code (IaC) practice for deploying the resources. To automate infrastructure deployment, you can use Azure DevOps Services or other CI/CD solutions. The deployment process is also idempotent - that is, repeatable to produce the same results. Azure [Pipelines](#) is part of [Azure DevOps Services](#) and runs automated builds, tests, and deployments.

Structure your deployment templates following the workload criteria, identify single units of works, and include them in their own template. In this scenario, at least seven workloads are identified and isolated in their own templates: the Azure virtual network and the VPN gateway, the management jump box, AD domain controllers, and SQL Server VMs, the Failover cluster and the availability group, and the Remaining VMs, Sharepoint primary node, Sharepoint cache, and network security group rules. Workload isolation makes it easier to associate the workload's specific resources to a team, so that the team can independently manage all aspects of those resources. This isolation enables DevOps to perform continuous integration and continuous delivery (CI/CD). This configuration also allows the staging of your workloads, which means deploying to various stages and running validations at each stage before moving on to the next one. This way, you can push updates to your production environments in a highly controlled way and minimize unanticipated deployment issues.

Consider using the [Azure Monitor](#) to Analyze and optimize the performance of your infrastructure, Monitor and diagnose networking issues without logging into your VMs.

For more information, see the DevOps section in [Azure Well-Architected Framework](#).

## Next steps

For more information about the individual pieces of the solution architecture, see the following topics:

- [Availability Sets](#)
- [Azure VPN Gateway](#)
- [Azure Load Balancer](#)
- [Hardware and software requirements for SharePoint Server 2016](#)
- [Overview of MinRole Server Roles in SharePoint Servers 2016 and 2019](#)
- [Always On availability groups](#)
- [Best practices for SQL Server in a SharePoint Server farm](#)
- [SharePoint Server 2016 in Microsoft Azure](#)
- [Azure Resource Manager resource group](#)
- [Microsoft Entra Domain Services](#)

## Related resources

- [Highly available SharePoint farm - Azure Solution Ideas](#)
- [Hybrid SharePoint farm with Microsoft 365](#)

# Azure Arc hybrid management and deployment for Kubernetes clusters

Azure Arc

Azure Kubernetes Service (AKS)

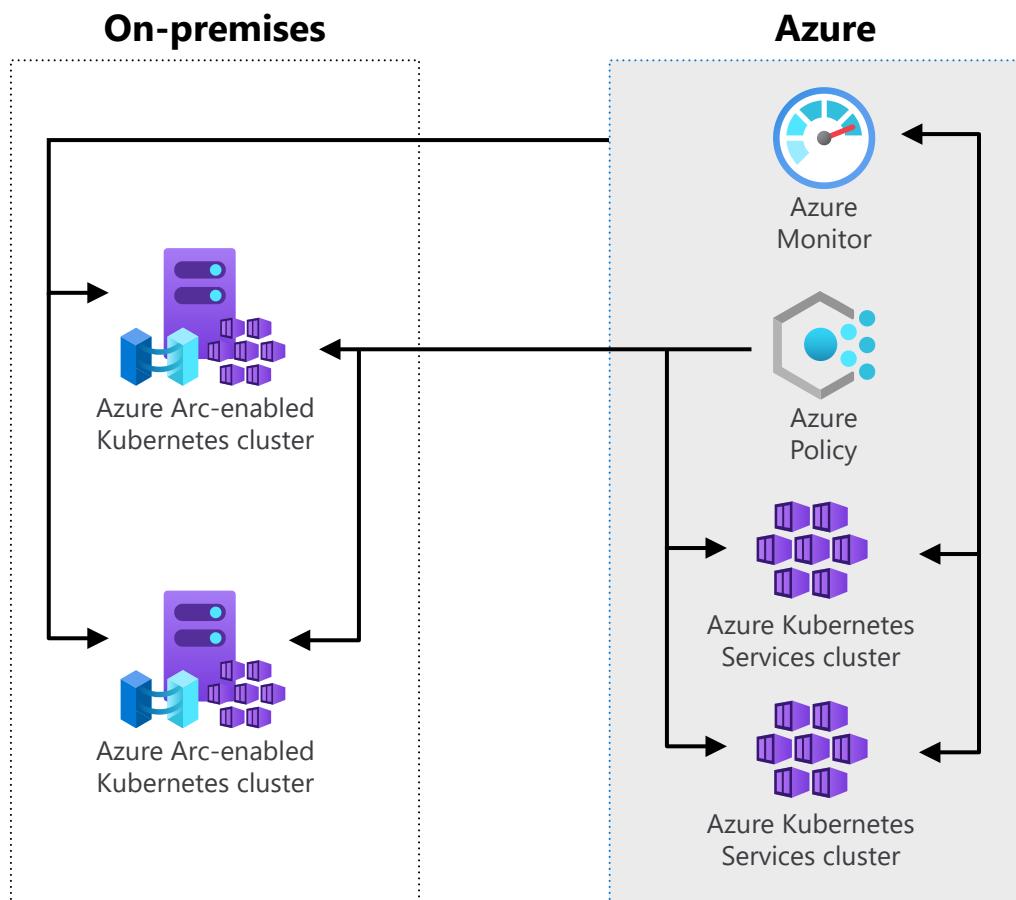
Azure Monitor

Azure Policy

Azure Role-based access control

This reference architecture demonstrates how Azure Arc extends Kubernetes cluster management and configuration across customer data centers, edge locations, and multiple cloud environments.

## Architecture



Download a [Visio file](#) of this architecture.

# Workflow

The architecture consists of the following aspects:

- **Azure Arc-enabled Kubernetes**. Attach and configure Kubernetes clusters inside or outside of Azure by using Azure Arc-enabled Kubernetes. When a Kubernetes cluster is attached to Azure Arc, it's assigned an Azure Resource Manager ID and a managed identity.
- **Azure Kubernetes Service**. Host Kubernetes clusters in Azure to reduce the complexity and operational overhead of Kubernetes cluster management.
- **On-premises Kubernetes cluster**. Attach Cloud Native Computing Foundation (CNCF)-certified Kubernetes clusters that are hosted in on-premises or third-party cloud environments.
- **Azure Policy**. Deploy and manage policies for Arc-enabled Kubernetes clusters.
- **Azure Monitor**. Observe and monitor Arc-enabled Kubernetes clusters.

## Components

- **Azure Arc** is a bridge that extends the Azure platform, making it possible to build applications and services that can run across datacenters, at the edge, and in multicloud environments.
- **Azure Kubernetes Service (AKS)** is a managed service for deploying and scaling Kubernetes clusters.
- **Azure Policy** makes it possible to achieve real-time cloud compliance at scale with consistent resource governance.
- **Azure Monitor** provides end-to-end observability for your applications, infrastructure, and network.

## Scenario details

You can use Azure Arc to register Kubernetes clusters that are hosted outside of Microsoft Azure. You can then use Azure tools to manage these clusters along with clusters that are hosted in Azure Kubernetes Service (AKS).

## Potential use cases

Typical uses for this architecture include:

- Managing on-premises Kubernetes clusters and clusters hosted in AKS for inventory, grouping, and tagging.
- Using Azure Monitor to monitor Kubernetes clusters across hybrid environments.

- Using Azure Policy to deploy and enforce policies for Kubernetes clusters across hybrid environments.
- Using Azure Policy to deploy and enforce GitOps.

## Recommendations

The following sections offer recommendations that apply to most scenarios. Microsoft recommends that you follow them unless you have a requirement that overrides them.

### Cluster registration

You can register any active CNCF Kubernetes cluster. You need a **kubeconfig** file to access the cluster and a cluster-admin role on the cluster to deploy Arc-enabled Kubernetes agents. You use the Azure Command-Line Interface (Azure CLI) to perform cluster registration tasks. The user or service principal that you use for the **az login** and **az connectedk8s connect** commands requires Read and Write permissions on the Microsoft.Kubernetes/connectedClusters resource type. The Kubernetes Cluster - Azure Arc Onboarding role has these permissions and can be used for role assignments on either the user principal or the service principal. Helm 3 is required for onboarding the cluster that uses the connectedk8s extension. Azure CLI version 2.3 or later is required to install the Azure Arc-enabled Kubernetes command-line interface extensions.

### Azure Arc agents for Kubernetes

Azure Arc-enabled Kubernetes consists of a few agents (also referred to as *operators*) that run in the cluster that's deployed to the **azure-arc** namespace:

- **deployment.apps/config-agent**. Watches the connected cluster for source control configuration resources that are applied on the cluster, and updates the compliance state.
- **deployment.apps/controller-manager**. An operator of operators that orchestrates interactions between Azure Arc components.
- **deployment.apps/metrics-agent**. Collects metrics from other Arc agents to ensure that these agents are exhibiting optimal performance.
- **deployment.apps/cluster-metadata-operator**. Gathers cluster metadata, cluster version, node count, and Azure Arc agent version.
- **deployment.apps/resource-sync-agent**. Syncs the previously mentioned cluster metadata to Azure.
- **deployment.apps/clusteridentityoperator**. Maintains the Managed Service Identity (MSI) certificate that's used by other agents to communicate with Azure.

- **deployment.apps/flux-logs-agent.** Collects logs from the flux operators that are deployed as a part of source control configuration.
- **deployment.apps/extension-manager.** Installs and manages the lifecycle of extension Helm charts.
- **deployment.apps/kube-azure-ad-proxy.** Used for the authentication of the requests that are sent to the cluster by using Cluster Connect.
- **deployment.apps/clusterconnect-agent.** A reverse proxy agent that enables the cluster connect feature to provide access to the apiserver of the cluster. It's an optional component that's deployed only if the cluster connect feature is enabled on the cluster.
- **deployment.apps/guard.** An authentication and authorization webhook server that's used for Microsoft Entra role-based access control (RBAC). It's an optional component that's deployed only if the azure-rbac feature is enabled on the cluster.

For more information, see [Connect an Azure Arc-enabled Kubernetes cluster](#).

## Monitor clusters by using Azure Monitor Container insights

Monitoring your containers is critical. Azure Monitor Container insights provides a rich monitoring experience for the AKS and AKS engine clusters. You can also configure Azure Monitor Container insights to monitor Azure Arc-enabled Kubernetes clusters that are hosted outside of Azure. Doing this provides comprehensive monitoring of your Kubernetes clusters across Azure, on-premises, and third-party cloud environments.

Azure Monitor Container insights can provide you with performance visibility by collecting memory and processor metrics from controllers, nodes, and containers, metrics that are available in Kubernetes through the Metrics application programming interface (API). Container logs are also collected. After you enable monitoring from Kubernetes clusters, metrics and logs are automatically collected for you by a containerized version of the Log Analytics agent. Metrics are written to the metrics store, and log data is written to the logs store that's associated with your Log Analytics workspace. For more information about Azure Monitor Container insights, see [Azure Monitor Container insights overview](#).

You can enable Azure Monitor Container insights for one or more deployments of Kubernetes by using either a PowerShell script or a Bash script.

To enable monitoring for Arc-enabled Kubernetes clusters, see [Enable monitoring of Azure Arc-enabled Kubernetes cluster](#)

# Use Azure Policy to enable GitOps-based application deployment

Use Azure Policy to enforce that each GitOps-enabled **Microsoft.Kubernetes/connectedclusters** resource or **Microsoft.ContainerService/managedClusters** resource has specific **Microsoft.KubernetesConfiguration/fluxConfigurations** applied on it. For example, you can apply a baseline configuration to one or more clusters, or deploy specific applications to multiple clusters. To use Azure Policy, select a definition from the [Azure Policy built-in definitions for Azure Arc-enabled Kubernetes](#), and then create a policy assignment.

When you create the policy assignment, set the scope to an Azure resource group or subscription. Also set the parameters for the **fluxConfiguration** that's created. When the assignment is created, the Policy engine will identify all **connectedCluster** or **managedCluster** resources that are located within the scope, and then apply the **fluxConfiguration** to each.

If you're using multiple source repositories for each cluster (for example, one repo for the central IT/cluster operator and other repos for application teams), activate this by using multiple policy assignments and configure each policy assignment to use a different source repo.

For more information, see [Deploy applications consistently at scale using Flux v2 configurations and Azure Policy](#).

## Deploy applications using GitOps

GitOps is the practice of declaring the desired state of Kubernetes configuration (deployments, namespaces, and so on) in a source repository, such as a Git or Helm repository, Buckets, or Azure Blob Storage. This is followed by a polling and pull-based deployment of these configurations to the cluster by using an operator.

The connection between your cluster and one or more source repositories is enabled by deploying the **microsoft.flux** extension to your cluster. The **fluxConfiguration** resource properties represent where and how Kubernetes resources should flow from the source repository to your cluster. The **fluxConfiguration** data is stored encrypted at rest in an Azure Cosmos DB database to ensure data confidentiality.

The **flux-config** agent that runs in your cluster is responsible for watching for new or updated **fluxConfiguration** extension resources on the Azure Arc-enabled Kubernetes resource, for deploying applications from the source repository, and for propagating

any updates that are made to the `fluxConfiguration`. You can even create multiple `fluxConfiguration` resources by using the `namespace` scope on the same Azure Arc-enabled Kubernetes cluster to achieve multi-tenancy.

The source repository can contain any valid Kubernetes resources, including Namespaces, ConfigMaps, Deployments, and DaemonSets. It can also contain Helm charts for deploying applications. Common source repository scenarios include defining a baseline configuration for your organization that can include common role-base access control (RBAC) roles and bindings, monitoring agents, logging agents, and cluster-wide services.

You can also manage a larger collection of clusters that are deployed across heterogeneous environments. For example, you can have one repository that defines the baseline configuration for your organization, and then apply that to multiple Kubernetes clusters simultaneously. You can also deploy applications to a cluster from multiple source repositories.

For more information, see [Deploy applications using GitOps with Flux v2](#).

## Topology, network, and routing

Azure Arc agents require the following protocols/ports/outbound URLs in order to function:

[+] [Expand table](#)

Endpoint (DNS)	Description
<code>https://management.azure.com:443</code>	Required for the agent to connect to Azure and register the cluster.
<code>https://[region].dp.kubernetesconfiguration.azure.com:443</code>	Data plane endpoint for the agent to push status and fetch configuration information, where [region] represents the Azure region that hosts the AKS instance.
<code>https://docker.io:443</code>	Required to pull container images.

Endpoint (DNS)	Description
<code>https://github.com:443, git://github.com:9418</code>	Example GitOps repos are hosted on GitHub. The configuration agent requires connectivity to whichever git endpoint that you specify.
<code>https://login.microsoftonline.com:443</code>	Required to fetch and update Azure Resource Manager tokens.
<code>https://azurearcfork8s.azurecr.io:443</code>	Required to pull container images for Azure Arc agents.

For a complete list of URLs across Azure Arc services, see [Azure Arc network requirements](#).

## Considerations

These considerations implement the pillars of the Azure Well-Architected Framework, which is a set of guiding tenets that can be used to improve the quality of a workload.

For more information, see [Microsoft Azure Well-Architected Framework](#).

## Reliability

Reliability ensures your application can meet the commitments you make to your customers. For more information, see [Overview of the reliability pillar](#).

- In most cases, the location that you select when you create the installation script should be the Azure region that's geographically closest to your on-premises resources. The rest of the data is stored within the Azure geography that contains the region that you specify, a fact that might affect your choice of region if you have data residency requirements. If an outage affects the Azure region that your machine is connected to, the outage doesn't affect the connected machine, but management operations that use Azure might not complete. For resilience when there's a regional outage, it's best, if you have multiple locations that provide a geographically redundant service, to connect the machines in each location to a

different Azure region. For available regions, consult [Supported regions for Azure Arc-enabled Kubernetes](#).

- You should ensure that the services that are referenced in the **Architecture** section are supported in the region where Azure Arc is deployed.

## Security

Security provides assurances against deliberate attacks and the abuse of your valuable data and systems. For more information, see [Overview of the security pillar](#).

- You can use Azure RBAC to manage access to Azure Arc-enabled Kubernetes across Azure and on-premises environments that use Microsoft Entra identities. For more information, see [Use Azure RBAC for Kubernetes Authorization](#).
- Microsoft recommends that you use a service principal that has limited privileges to onboard Kubernetes clusters to Azure Arc. This practice is useful in CI/CD pipelines such as Azure Pipelines and GitHub Actions. For more information, see [Create an Azure Arc-enabled onboarding Service Principal](#).
- To simplify service principal management, you can use managed identities in AKS. However, clusters must be created by using the managed identity, and the existing clusters (including Azure and on-premises clusters) can't be migrated to managed identities. For more information, see [Use managed identities in Azure Kubernetes Service](#).

## Cost optimization

Cost optimization is about looking at ways to reduce unnecessary expenses and improve operational efficiencies. For more information, see [Overview of the cost optimization pillar](#).

General cost considerations are described in the [Principles of cost optimization](#) section in the Microsoft Azure Well-Architected Framework.

## Operational excellence

Operational excellence covers the operations processes that deploy an application and keep it running in production. For more information, see [Overview of the operational excellence pillar](#).

- Before you configure your Azure Arc-enabled Kubernetes clusters, review the Azure Resource Manager [Subscription limits](#) and [Resource group limits](#) to plan for the number of clusters.

- Use Helm, the open-source packaging tool, to install and manage the Kubernetes application lifecycles. Similar to Linux package managers such as APT and Yum, you use Helm to manage Kubernetes *charts*, which are packages of preconfigured Kubernetes resources.

## Contributors

*This article is maintained by Microsoft. It was originally written by the following contributors.*

Principal author:

- [Pieter de Bruin](#) ↗ | Senior Program Manager

*To see non-public LinkedIn profiles, sign in to LinkedIn.*

## Next steps

- [Azure Arc documentation](#)
- [Azure Arc-enabled Kubernetes documentation](#)
- [Azure Kubernetes Service documentation](#)
- [Azure Policy documentation](#)
- [Azure Monitor documentation](#)
- [Connect an Azure Arc-enabled Kubernetes cluster](#)

## Related resources

Related hybrid guidance:

- [Hybrid architecture design](#)
- [Azure hybrid options](#)
- [Hybrid app design considerations](#)
- [Deploy a hybrid app with on-premises data that scales cross-cloud](#)

Related architectures:

- [Run containers in a hybrid environment](#)
- [Baseline architecture for AKS on Azure Stack HCI](#)
- [Network architecture for AKS on Azure Stack HCI](#)
- [Optimize administration of SQL Server instances in on-premises and multicloud environments by using Azure Arc](#)
- [Enterprise monitoring with Azure Monitor](#)

# Azure Automation in a hybrid environment

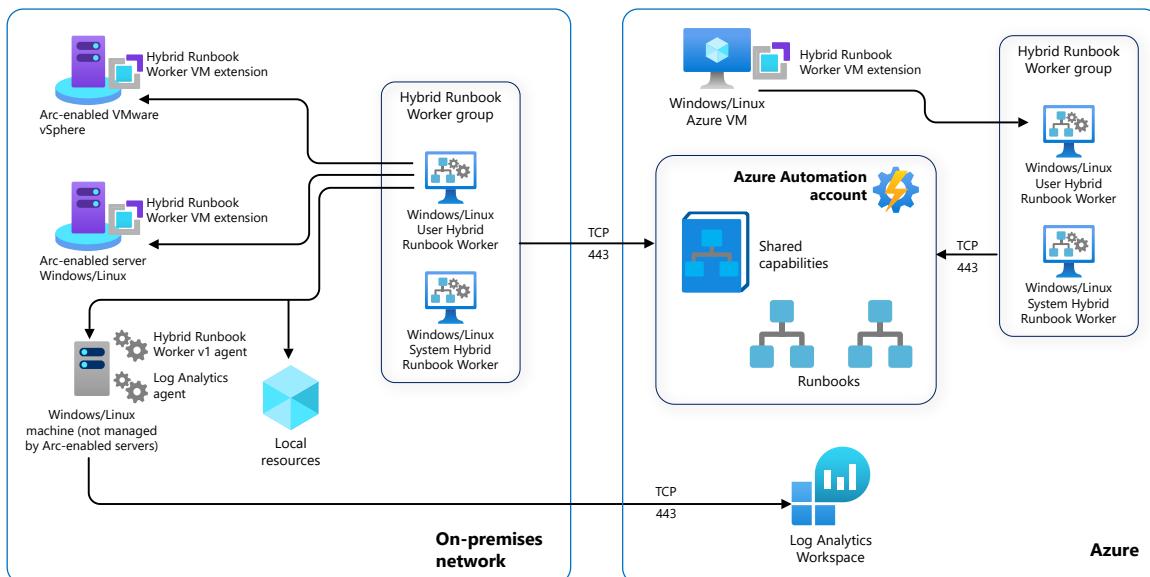
Azure Automation   Azure Portal   Azure Monitor   Azure Virtual Machines   Azure Arc

## ⓘ Important

Azure Automation Agent-based User Hybrid Runbook Worker (Windows and Linux) will retire on **31 August 2024** and wouldn't be supported after that date. You must complete migrating existing Agent-based User Hybrid Runbook Workers to Extension-based Workers before 31 August 2024. Moreover, starting **1 October 2023**, creating new Agent-based Hybrid Workers wouldn't be possible. [Learn more](#)

Runbooks in Azure Automation run on the Azure cloud platform and might not have access to resources that are in other clouds or in your on-premises environment. You can use the Hybrid Runbook Worker feature of Azure Automation to run runbooks directly on the machine that hosts the role and against resources in the environment to manage those local resources. Runbooks are stored and managed in Azure Automation and then delivered to one or more assigned machines.

## Architecture



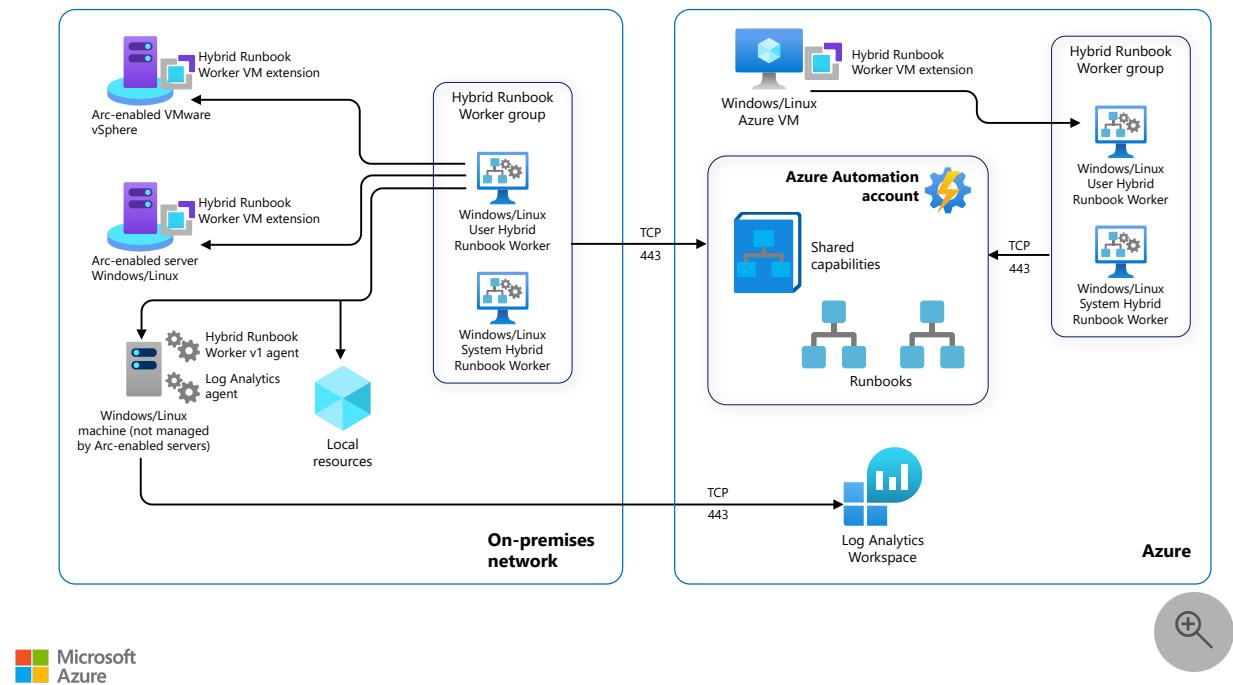
Download a [Visio file](#) of this architecture.

## Workflow

The Hybrid Runbook Worker architecture consists of the following:

- **Automation account:** A cloud service that automates configuration and management across your Azure and non-Azure environments.
- **Hybrid Runbook Worker:** A computer that's configured with the Hybrid Runbook Worker feature and can execute runbooks directly on the computer and against the resources in the local environment.
- **Hybrid Runbook Worker group:** Group with multiple Hybrid runbook workers for higher availability and scale to run a set of runbooks.
- **Runbook:** A collection of one or more linked activities that together automate a process or operation. [Learn more](#).
- **On-premises machines and VMs:** Windows or Linux on-premises computers and VMs that are hosted in a private local-area network.
- **Components that are applicable to the extension-based approach (V2):**
  - **Hybrid Runbook Worker VM extension:** A small application that's installed on a computer. The application configures the computer as a Hybrid Runbook Worker.
  - **Arc-enabled server:** Azure Arc-enabled servers make it possible for you to manage Windows and Linux computers and virtual machines that are hosted outside of Azure, whether on your corporate network or on another cloud provider. This management experience is designed to be consistent with how you manage native Azure virtual machines. [Learn more](#).
- **Components that are applicable to the agent-based approach (V1):**
  - **Log Analytics workspace:** A Log Analytics workspace is a data repository for log data that's collected from resources that run in Azure, on-premises, or in another cloud provider.
  - **Automation Hybrid Worker solution:** With this solution, you can create Hybrid Runbook Workers to run Azure Automation runbooks on your Azure and non-Azure computers.

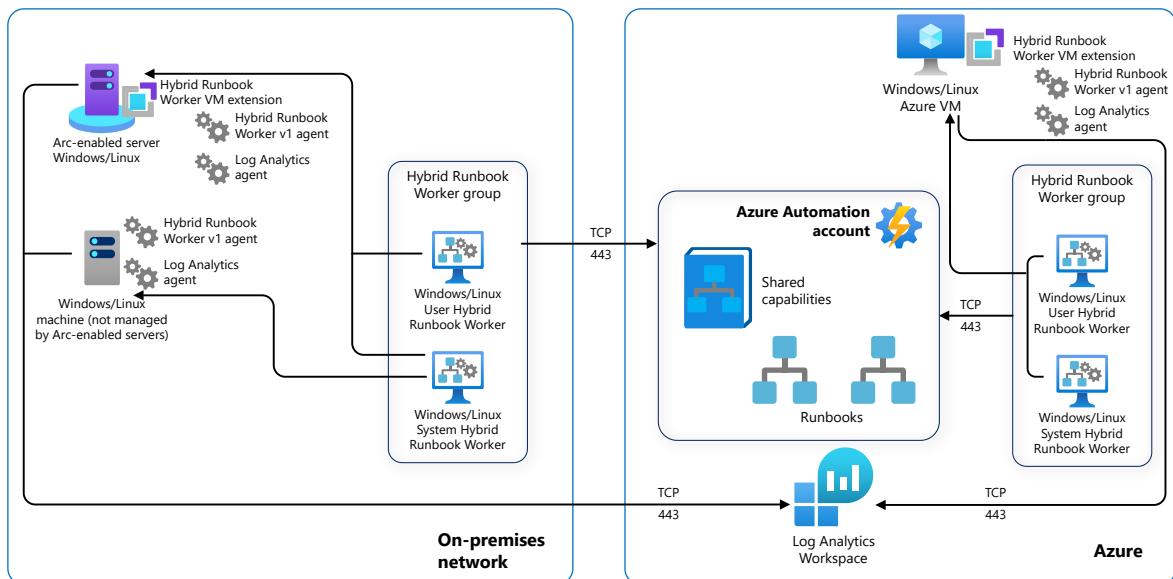
## User Hybrid Runbook Worker



Download a [Visio file](#) of this architecture.

Each user Hybrid Runbook Worker is a member of a Hybrid Runbook Worker group that you specify when you install the worker. A group can include a single worker, but you can include multiple workers in a group for high availability. Each machine can host one Hybrid Runbook Worker reporting to one Automation account; you can't register the hybrid worker across multiple Automation accounts. A hybrid worker can only listen for jobs from a single Automation account.

## System Hybrid Runbook Worker



Download a [Visio file](#) of this architecture.

Machines that host the system Hybrid Runbook Worker that's managed by Update Management can be added to a Hybrid Runbook Worker group. But you must use the same Automation account for both Update Management and the Hybrid Runbook Worker group membership.

## Job execution on Hybrid Runbook Worker

When you start a runbook on a user Hybrid Runbook Worker, you specify the group that it runs on. Each worker in the group polls Azure Automation to see if any jobs are available. If a job is available, the first worker to get the job takes it. The processing time of the jobs queue depends on the hybrid worker hardware profile and load. You can't specify a particular worker. Hybrid worker works on a polling mechanism (every 30 secs) and follows an order of first-come, first-served.

## Components

- [Azure Automation](#) is an Azure service for automating cloud management tasks. The **Hybrid Runbook Worker** feature makes it possible to run runbooks on machines that are located in your datacenter in order to manage local resources.
- [Azure Monitor](#) gives you full observability into applications, infrastructure, and network. **Azure Monitor Logs** is a feature of Azure Monitor that collects and organizes log and performance data from monitored resources. **Log Analytics** is a tool in the Azure portal for querying logs and for analyzing the results

## Scenario details

### Hybrid Runbook Worker installation approach

Azure Automation provides native integration of the Hybrid Runbook Worker role through the Azure virtual machine extension framework. The Azure VM agent is responsible for managing the extension on Azure VMs, both Windows and Linux, and on non-Azure machines through the Arc-enabled servers connected machine agent. There are two Hybrid Runbook Workers installation platforms that are supported by Azure Automation.

[\[ \]](#) Expand table

Platform	Description
Extension-based(V2)	Installed by using the Hybrid Runbook Worker VM extension, without any dependency on the reporting activity of the Log Analytics agent that reports to an Azure Monitor Log Analytics workspace. <b>This is the recommended approach</b> , as it offers seamless onboarding and is easy to manage.
Agent-based(V1)	Installed after the Log Analytics agent finishes reporting to an Azure Monitor Log Analytics workspace.

A hybrid worker can co-exist with both platforms: Agent based (V1) and Extension based (V2). If you install Extension based (V2) on a hybrid worker that's already running Agent based (V1), you see two entries of the Hybrid Runbook Worker in the group. One with Platform Extension based (V2) and the other Agent based (V1). [Learn more](#)

## Runbook worker types

There are two types of Runbook workers, System and User.

**System** supports a set of hidden runbooks that are used by the Update Management feature. The runbooks are designed to install user-specified updates on Windows and Linux machines. This type of Hybrid Runbook Worker isn't a member of a Hybrid Runbook Worker group, and therefore doesn't run runbooks that target a Runbook Worker group.

**User** supports user-defined runbooks that are intended to run directly on the Windows and Linux machines that are members of one or more Runbook Worker groups.

The extension-based Hybrid Runbook Worker only supports the user Hybrid Runbook Worker type and doesn't include the system Hybrid Runbook Worker required for the Update Management feature.

Agent-based (V1) Hybrid Runbook Workers rely on the [Log Analytics agent](#) reporting to an Azure Monitor [Log Analytics workspace](#). The workspace isn't only to collect monitoring data from the machine, but also to download the components that are required to install the agent-based Hybrid Runbook Worker. When Azure Automation [Update Management](#) is enabled, any machine that's connected to your Log Analytics workspace is automatically configured as a system Hybrid Runbook Worker.

## Potential use cases

- To execute Azure Automation runbooks directly on an existing Azure virtual machine (VM) or on-premises Arc-enabled server.
- To overcome the Azure Automation sandbox limitation. The common scenarios include executing long-running operations beyond the three-hour limit for cloud jobs, performing resource-intensive automation operations, interacting with local services that run on-premises or in hybrid environments, running scripts that require elevated permissions, and so on.
- To overcome organizational restrictions on keeping data in Azure for governance and security reasons. Even though you can't execute Automation jobs in the cloud, you can run them on an on-premises machine that's onboarded as a Hybrid Runbook Worker.
- To automate operations on multiple non-Azure resources that run in on-premises, hybrid, or multicloud environments. You can onboard one of those machines as a Hybrid Runbook Worker and target automation on the remaining on-premises machines.
- To access other services privately from the Azure Virtual Network (VNet) without having to open an outbound connection to the internet, you can execute runbooks on a Hybrid Worker connected to the Azure virtual network.

## Considerations

These considerations implement the pillars of the Azure Well-Architected Framework, which is a set of guiding tenets that can be used to improve the quality of a workload. For more information, see [Microsoft Azure Well-Architected Framework](#).

## Reliability

Reliability ensures that your application can meet the commitments that you make to your customers. For more information, see [Overview of the reliability pillar](#).

- A Hybrid Runbook Worker Group with more than one machine configured with the Hybrid Worker Role provides high availability because runbooks will start only on servers that are running and healthy.
- The extension-based (V1) Hybrid Runbook Worker only supports the User Hybrid Runbook Worker type and doesn't include the System Hybrid Runbook Worker that's required for the Update Management feature.
- The following applies only to the agent-based approach (V1). Currently, mappings between a Log Analytics Workspace and an Automation account are supported in several regions. For more information, see [Supported regions for linked Log Analytics workspace](#).

# Security

Security provides assurances against deliberate attacks and the abuse of your valuable data and systems. For more information, see [Overview of the security pillar](#).

- **Encryption of sensitive assets in Automation:** An Azure Automation Account can contain sensitive assets such as credentials, certificate, connection, and encrypted variables that might be used by the runbooks. Each secure asset is encrypted by default by using a Data Encryption key that's generated for each Automation Account. These keys are encrypted and stored in Azure Automation with an Account Encryption Key (AEK) that can be stored in the Key vault for customers who want to manage encryption with their own keys. By default, AEK is encrypted by using Microsoft-managed keys. Use the following guidelines to [apply encryption of secure assets in Azure Automation](#).
- **Runbook permission:** By default, runbook permissions for a Hybrid Runbook Worker run in a system context on the machine where they're deployed. A runbook provides its own authentication to local resources. Authentication can be configured by using managed identities for Azure resources or by specifying a Run As account to provide a user context for all runbooks.
- **Network planning:**
  - If you use a proxy server for communication between Azure Automation and the machines that run the Hybrid Runbook Worker, ensure that the appropriate resources are accessible. The timeout for requests from the Hybrid Runbook Worker and Automation services is 30 seconds. After three attempts, the request fails.
  - Hybrid Runbook Worker requires outbound internet access over TCP port 443 to communicate with Automation. If you use a firewall to restrict access to the Internet, you must configure the firewall to permit access. For agent-based (V1) computers with restricted internet access, use Log Analytics gateway to configure communication with Azure Automation and Azure Log Analytics Workspace.
  - There is a CPU quota limit of 5% while configuring extension-based Linux Hybrid Runbook worker. There is no such limit for Windows extension-based Hybrid Runbook Worker.
- **Azure security baseline for Automation:** [The Azure security baseline for Automation](#) contains recommendations on how to improve your security configuration to protect your assets by following the best-practice guidance.

# Cost optimization

Cost optimization is about looking at ways to reduce unnecessary expenses and to improve operational efficiencies. For more information, see [Overview of the cost optimization pillar](#).

- Azure Automation costs are priced for job execution per minute. Every month, the first 500 minutes of process automation are free. Use the [Azure pricing calculator](#) to estimate costs. For more information about the Azure Automation pricing models, see [Automation pricing](#).
- For the agent-based approach (V1), Azure Log Analytics Workspace might generate additional costs that are related to the amount of log data that's stored in Azure Log Analytics. The pricing model is based on consumption. The costs are associated for data ingestion and data retention. For ingesting data into Azure Log Analytics, use Capacity Reservation or the Pay-As-You-Go model that includes 5 gigabytes (GB) free per billing account per month. Data retention for the first 31 days is free of charge. For the pricing models for Log Analytics, see [Azure Monitor pricing](#).

## Operational excellence

Operational excellence covers the operations processes that deploy an application and keep it running in production. For more information, see [Overview of the operational excellence pillar](#).

## Manageability

- The extension-based approach (V2) offers ease of manageability as compared to agent-based approach (V1) through:
  - Native integration with ARM identity for Hybrid Runbook Worker and provides the flexibility for governance at scale through policies and templates.
  - Centralized control and management of identities and resource credentials, since it uses VM system-assigned identities that are provided by Microsoft Entra ID.
  - Unified experience for both Azure and non-Azure machines while onboarding and deboarding Hybrid Runbook Workers.
- Applicable only for agent-based approach (V1):
  - To accelerate deployment of the Log Analytics Agent with Hybrid Worker Role running on Windows machine, use the PowerShell script [New-OnPremiseHybridWorker.ps1](#)
  - Deployment of many agents in on-premises infrastructure can be orchestrated by using command line scripts and deployed by using Group Policy or System Center Configuration Manager.

## DevOps

- Azure Automation allows integration with popular source control systems, Azure DevOps and GitHub. With source control, you can integrate the existing development environment that contains your scripts and custom code that have been previously tested in an isolated environment.
- For information on how to integrate Azure Automation with your source control environment, see [Use source control integration](#).

## Contributors

*This article is maintained by Microsoft. It was originally written by the following contributors.*

Principal author:

- [Mike Martin](#) | Senior Cloud Solution Architect

*To see non-public LinkedIn profiles, sign in to LinkedIn.*

## Next steps

More about Azure Automation:

- [Hybrid Runbook Worker overview](#)
- [Deploy extension-based Windows or Linux User Hybrid Runbook Worker](#)
- [Deploy an agent-based Windows Hybrid Runbook Worker in Automation](#)
- [Deploy an agent-based Linux Hybrid Runbook Worker in Automation](#)
- [Create an Azure Automation account](#)
- [Create a runbook in Azure Automation using Managed Identities](#)
- [Run Automation runbooks on a Hybrid Runbook Worker](#)
- [Pre-requisites: Azure Automation network configuration details](#)
- [Azure Arc Overview](#)
- [What is Azure Arc enabled servers?](#)

More about Azure Monitor and Monitor Logs:

- [Azure Monitor Logs overview](#)
- [Overview of Log Analytics in Azure Monitor](#)

## Related resources

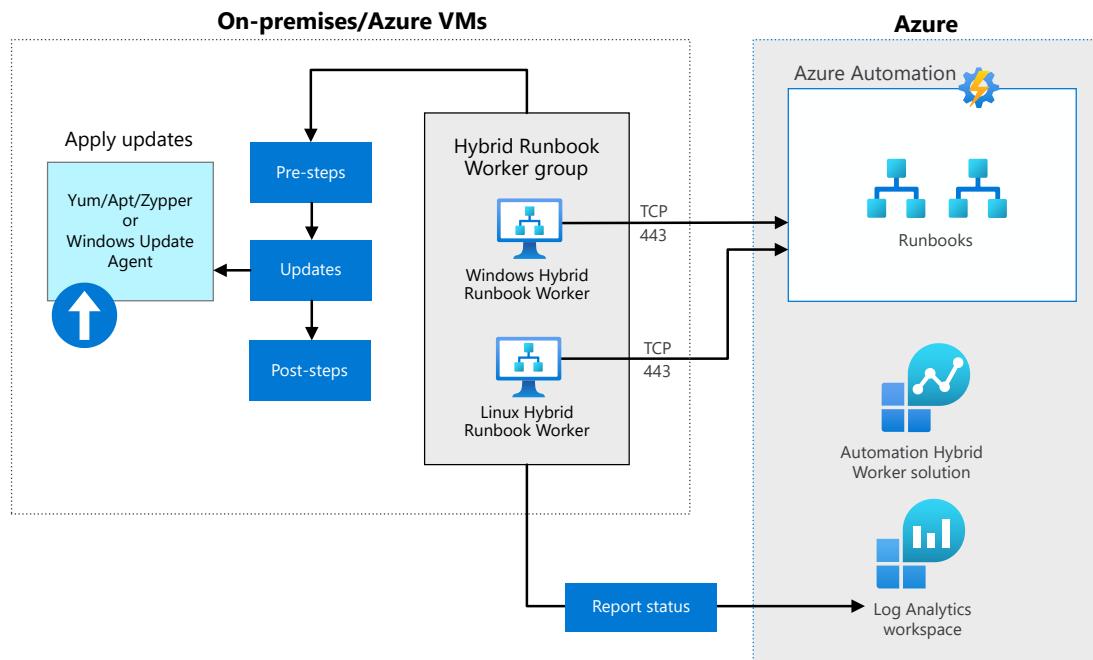
- Hybrid architecture design
- Connect an on-premises network to Azure
- Enterprise monitoring with Azure Monitor
- Computer forensics chain of custody in Azure
- Disaster Recovery for Azure Stack Hub virtual machines

# Azure Automation update management

Azure Automation   Azure Log Analytics   Azure Monitor   Azure Resource Manager   Azure Virtual Machines

This reference architecture illustrates how to design a hybrid update management solution to manage updates on both Microsoft Azure and on-premises Windows and Linux computers.

## Architecture



Download a [Visio file](#) of this architecture.

## Workflow

The architecture consists of the following services:

- **Log Analytics workspace:** A [Log Analytics workspace](#) is a data repository for log data that's collected from resources that run in Azure, on-premises, or in another cloud provider.

- **Automation Hybrid Worker solution:** Create [Hybrid Runbook Workers](#) to run [Azure Automation](#) runbooks on your Azure and non-Azure computers.
- **Automation account:** This is a cloud service that automates configuration and management across your Azure and non-Azure environments.
- **Hybrid Runbook Worker:** This is a computer that's configured with the Hybrid Runbook Worker feature and can run runbooks directly on the computer and against the resources in the local environment.
- **Hybrid Runbook Worker group:** It's a group of Hybrid Runbook Workers used for high availability.
- **Runbook:** This is a collection of one or more linked activities that together automate a process or operation.
- **On-premises computers and VMs:** These are on-premises computers and VMs with Windows or Linux operating systems that reside on-premises.
- **Azure VMs:** Azure VMs include Windows or Linux VMs that are hosted in Azure.

## Components

- [Azure Automation](#) ↗
- [Azure Virtual Machines](#) ↗
- [Azure Monitor](#) ↗
- [Azure Arc](#) ↗

## Scenario details

Typical uses for this architecture include:

- Managing updates across on-premises and in Azure using the Update Management component of Automation Account.
- Using scheduled deployments to orchestrate the installation of updates within a defined maintenance window.

## Recommendations

The following recommendations apply for most scenarios. Follow them unless you have a specific requirement that overrides them.

## Update Management

[Update Management](#) is a configuration component of Automation. Windows and Linux computers, both in Azure and on-premises, send assessment information about missing

updates to the Log Analytics workspace. Azure Automation then uses that information to create a schedule for automatic deployment of the missing updates.

The following steps highlight the actual implementation:

1. Create a Log Analytics workspace.
2. Create an Automation account.
3. Link the Automation account with the Log Analytics workspace.
4. Enable Update Management for Azure VMs.
5. Enable Update Management for non-Azure VMs.

## Create a Log Analytics workspace

Before you create a Log Analytics workspace, ensure that you have at least Log Analytics Contributor role permissions.

You can have more than one Log Analytics workspace for data isolation or for geographic location of data storage, but the Log Analytics agent can be configured to report to one Log Analytics workspace. For more information, review the [Designing your Azure Monitor Logs deployment](#) before you create the workspace.

Use the following procedure to create a Log Analytics workspace:

1. Sign in to the Azure portal at <https://portal.azure.com>.
2. In the Azure portal, select **Create a resource**.
3. In the **Search the Marketplace** box, enter **Log Analytics**. As you begin entering this text, the list filters based on your input. Select **Log Analytics workspaces**.
4. Select **Create**, and then configure the following items:
  - a. Select a different **Subscription** in the drop-down list if the default selection isn't appropriate.
  - b. For the **Resource Group**, choose to use an existing resource group that's already set up or create a new one.
  - c. Provide a unique name for the new **Log Analytics workspace**, such as *HybridWorkspace-yourname*
  - d. Select the **Location** for your deployment.
  - e. Select **pricing tier** to proceed to further customizations.
  - f. If you're creating a workspace in a subscription that was created after April 2, 2018, it'll automatically use the **Per GB** pricing plan, and the option to select a pricing tier won't be available. If you're creating a workspace for an existing subscription that was created before that date or for a subscription that was tied to an existing Enterprise Agreement enrollment, select your preferred

pricing tier. For more information about the particular tiers, refer to [Log Analytics Pricing details](#).

- g. Select **Tags** and optionally provide a name and value for categorization of the resources.
  - h. Select **Review + Create**.
5. After providing the required information in the **Log Analytics workspace** pane, select **Create**.

## Create an Automation account

After the Automation Hybrid Worker solution has been added to the Log Analytics workspace, proceed with creation of the Automation account. Refer to [Supported regions for linked Log Analytics workspace](#) to select the regions for Automation account and Log Analytics workspace. It's important that you create the Automation account based on the region mapping document and preferably in the same resource group as the Log Analytics workspace.

Use the following procedure to create an Automation account:

1. In the Azure portal, select **Create a resource**.
2. In the **Search the Marketplace** box, enter **Automation**. As you begin entering this text, the list filters based on your input. Select **Automation**, and then select **Create**.
3. Select **Create**, and then configure the following items:
  - a. Provide the **Name** for the Automation account, such as *hybrid-auto*.
  - b. Select a different **Subscription** in the drop-down list if the default selection isn't appropriate.
  - c. For the **Resource Group**, choose the same resource group in which you want to create the automation account.
  - d. Select the **Location** based on the region mapping document.
  - e. **Create Azure Run As account** is optional because this only provides authentication with Azure to manage Azure resources from Automation runbooks.
4. After providing the required information in the **Add Automation Account** pane, select **Create**.

## Link the Automation account with the Log Analytics workspace

Automation accounts use the Hybrid Runbook Worker components that deploy in the Log Analytics workspace. You must integrate those services before you deploy a Log Analytics agent on an on-premises computer. Currently, mappings between Log

Analytics workspaces and Automation accounts are supported in several regions. For further information, refer to [Supported regions for linked Log Analytics workspace](#).

Use the following procedure to link an Automation account with a Log Analytics workspace:

1. In the Azure portal, select **All services**, and then enter **automation**. As you begin entering this text, the list filters based on your input. Select **Automation Account**, and then select the Automation account that you created earlier.
2. In the **Automation Account** pane, select **Update Management** in the **Update Management** section.
3. In the **Update Management** pane, configure the following items:
  - a. Select a different **Subscription** in the drop-down list if the default selection isn't appropriate.
  - b. For **Log Analytics workspace**, select your existing Log Analytics workspace; for example, *HybridWorkspace-yourname*.
4. After providing the required information in the **Update Management** pane, select **Enable**.

## Enable Update Management for Azure VMs

Enable Update Management for Azure VMs by using the following tools:

- [Azure Resource Manager template](#). Microsoft provides a sample template that can automate creation of an Azure Log Analytics workspace, creation of an Automation account, linking the Automation account to the Log Analytics workspace, and enabling Update Management.
- [Update Management from the Azure portal](#). Use this method when you want to update multiple VMs that reside in different regions.
- [Update Management from an Azure VM](#). This will configure updates for a selected VM.
- [Update Management from an Automation account](#). Use this method when you want to update both Azure and non-Azure computers and VMs at the same time.
- [Update Management from a runbook](#). Use this method to enable Update Management as an automated procedure combined with other automation activities.

Use the following procedure to enable Update Management for Azure VMs:

1. In the Azure portal, select **All services**, and then enter **automation**. As you begin entering this text, the list filters based on your input. Select **Automation Account**, and then select the Automation account that you created earlier.

2. In the **Automation Account** pane, select **Update Management** in the **Update Management** section.
3. In the **Update Management** pane, select **Add Azure VMs**, select one or more VMs that are ready for Update Management, and then select **Enable**.

## Deploy the Log Analytics agent and connect to a Log Analytics workspace

Deploying a Hybrid Runbook Worker component is part of the deployment of a Log Analytics agent.

If you test the solution by using an Azure VM, you can install the Log Analytics agent and enroll the VM in an existing Log Analytics workspace by using a VM extension for both [Linux](#) and [Windows](#). You can also deploy the agent by using Azure Automation Desired State Configuration, a Windows PowerShell script, or by using a Resource Manager template for VMs. For more information, refer to [Connect Windows computers to Azure Monitor](#).

For non-Azure VMs, deploy the agent by using a manual or automated process both on physical Windows and Linux computers or VMs that are in your environment.

For Windows computers, configure the agent to communicate with the Log Analytics service by using the Transport Layer Security (TLS) 1.2 protocol. Refer to [Connect Windows computers to Azure Monitor](#) for a detailed explanation of the deployment procedure.

The Log Analytics agent for Linux can be deployed:

- [Manually](#) by using a shell script bundle that contains Debian and Red Hat Package Manager (RPM) packages for each of the agent components. This is recommended when a Linux computer doesn't have internet connectivity and will communicate with the Log Analytics service through the [Log Analytics gateway](#).
- By using a [wrapper-script](#) that's hosted on GitHub when the computer has connectivity to the internet.

The Log Analytics agent must be configured to communicate with a Log Analytics workspace by using the [workspace ID and key](#) of the Log Analytics workspace.

Use the following procedure to deploy a Log Analytics agent and connect to a Log Analytics workspace:

1. In the Azure portal, search for and select **Log Analytics workspaces**.

2. In your list of Log Analytics workspaces, select the workspace that the agent uses for reporting.
3. Select **Agents management**.
4. Copy and paste the **Workspace ID** and **Primary Key** into your favorite editor.
5. In your Log Analytics workspace, from the **Windows Servers** page that you browsed to earlier, select the appropriate **Download Windows Agent** version to download based on the processor architecture of the Windows operating system.
6. Run **Setup** to install the agent on your computer.
7. On the **Welcome** page, select **Next**.
8. On the **License Terms** page, read the license, and then select **I Agree**.
9. On the **Destination Folder** page, change or keep the default installation folder, and then select **Next**.
10. On the **Agent Setup Options** page, choose to connect the agent to Azure Log Analytics, and then select **Next**.
11. On the **Azure Log Analytics** page, perform the following steps:
  - a. Paste the **Workspace ID** and **Workspace Key (Primary Key)** that you copied earlier. If the computer reports to a Log Analytics workspace in a Microsoft Azure Government cloud, select **Azure US Government** in the **Azure Cloud** drop-down list.
  - b. If the computer needs to communicate through a proxy server to the Log Analytics service, select **Advanced**, and then provide the URL and port number of the proxy server. If your proxy server requires authentication, enter the username and password to authenticate with the proxy server, and then select **Next**.
12. Select **Next** after you finish providing the necessary configuration settings.

## Enable Update Management for non-Azure computers

Enabling Update Management on non-Azure computers has the following prerequisites:

- Deploy the Log Analytics agent and connect to a Log Analytics workspace.

Previous procedures explain how to configure those prerequisites.

After installing the Log Analytics agent on an on-premises computer, enable Update Management in the Azure portal by using the following procedure:

1. In the Azure portal, select **All services**, and then enter **automation**. As you begin entering this text, the list filters based on your input. Select **Automation Account**, and then select the Automation account that you created earlier.
2. In the **Automation Account** pane, select **Update Management** in the **Update Management** section.

3. In the **Update Management** pane, select **Manage machines**, and then select computers that are listed and have been configured to send log data to the Log Analytics workspace.
4. Select **Enable** to finish the configuration of Update Management on non-Azure machines.

Each Windows computer managed by Update Management is listed in the **Hybrid Worker groups** pane as a System Hybrid Worker group for the Automation account. Use these groups only for deploying updates, not for targeting the groups with runbooks for automated tasks.

## Considerations

These considerations implement the pillars of the Azure Well-Architected Framework, which is a set of guiding tenets that can be used to improve the quality of a workload. For more information, see [Microsoft Azure Well-Architected Framework](#).

## Manageability

### Manage updates for Azure VMs and non-Azure machines

Update assessment for all missing updates that both Azure VMs and non-Azure computers require is visible in the **Update Management** section of your Automation account.

Schedule an update deployment by using the Azure portal or by using PowerShell, which creates schedule assets that are linked to the **Patch-MicrosoftOMSComputers** runbook.

Use the following procedure to schedule a new update deployment:

1. In your Automation account, go to **Update management** under **Update management**, and then select **Schedule update deployment**.
2. Under **New update deployment**, use the **Name** box to enter a unique name for your deployment.
3. Select the operating system to target for the update deployment.
4. In the **Groups to update** pane, define a query that combines subscription, resource groups, locations, and tags to build a dynamic group of Azure VMs to include in

your deployment. To learn more, refer to [Use dynamic groups with Update Management](#).

5. In the **Machines to update** pane, select a saved search, an imported group, or pick **Machines** from the drop-down menu, and then select individual machines.
6. Use the **Update classifications** drop-down menu to specify [update classifications](#) for products.
7. Use the **Include/exclude updates** pane to select specific updates for deployment.
8. Select **Schedule settings** to define a time when the update deployment will run on computers.
9. Use the **Recurrence** box to specify if the deployment occurs once or uses a recurring schedule, and then select **OK**.
10. In the **Pre-scripts + Post-scripts (Preview)** region, select the scripts to run before and after your deployment. To learn more, refer to [Manage pre-scripts and post-scripts](#).
11. Use the **Maintenance window (minutes)** box to specify the amount of time that's allowed for updates to install.
12. Use the **Reboot options** box to specify the way to handle reboots during deployment.
13. When you finish configuring the deployment schedule, select **Create**.

Results of a completed update deployment are visible in the **Update Management** pane on the **History** tab.

## Configure Windows Update settings

Azure Update Management depends on Windows Update Client to download and install updates either from Windows Update (default setting) or from Windows Server Update Server. Configure Windows Update Client settings to connect to Windows Server Update Services (WSUS) by using:

- Local Group Policy Editor
- Group Policy
- PowerShell
- Directly editing the registry

For more information, refer to how to [configure Windows Update settings](#).

## Integrate Update Management with Microsoft Endpoint Configuration Manager

The Software Update Management cycle can integrate with Microsoft Endpoint Configuration Manager for customers that are already using this product to manage PCs, servers, and mobile devices.

To integrate Software Update Management with Endpoint Configuration Manager, first integrate Endpoint Configuration Manager with Azure Monitor logs and import the collections in the Log Analytics workspace.

For details, refer to [Connect Configuration Manager to Azure Monitor](#).

To manage updates on local computers, configure them with:

- The Endpoint Configuration Manager client.
- The Log Analytics agent, which is configured to report to a Log Analytics workspace that's enabled for Update Management.
- Windows agents that are configured to communicate with WSUS or have access to Microsoft Update.

To manage updates on computers with Endpoint Configuration Manager, deploy the following roles on the Endpoint Configuration Manager computer:

- Management point. This site system role manages clients with a policy that contains configuration settings and service location information.
- Distribution point. This contains source files for clients.
- Software update point. This is a role on the server that's hosting WSUS.

Manage software updates by using:

- Endpoint Configuration Manager
- Azure Automation

Partner updates on Windows machines can be deployed from a custom repository that [System Center Updates Publisher](#) (SCUP) provides. SCUP can import custom updates either in standalone WSUS or integrated with Endpoint Configuration Manager.

For more information, refer to [Integrate Update Management with Windows Endpoint Configuration Manager](#).

## Deploy the Log Analytics agent by using a PowerShell script

To accelerate deployment of the Log Analytics agent with the Hybrid Worker role running on a Windows computer, use the **New-OnPremiseHybridWorker.ps1** PowerShell script. The script:

- Installs the necessary modules.
- Signs in with your Azure account.
- Verifies the existence of a specified resource group and Automation account.
- Creates references to Automation account attributes.
- Creates an Azure Monitor Log Analytics workspace if not specified.
- Enables the Automation solution in the workspace.
- Downloads and installs the Log Analytics agent for the Windows operating system.
- Registers the machine as a Hybrid Runbook Worker.

Deploying many agents in an on-premises infrastructure can be orchestrated by using command-line scripts and by using Group Policy or Endpoint Configuration Manager.

## Use dynamic groups for Azure and non-Azure machines

Dynamic groups for Azure VMs filter VMs based on a combination of:

- Subscriptions
- Resource groups
- Locations
- Tags

Dynamic groups for non-Azure computers use saved searches to filter the computers for deployment of the update. Saved searches, also known as *computer groups*, can be created by using:

- A log query. Use Azure Data Explorer to define a logical expression to filter the computers.
- Active Directory Domain Services. A group is created in Log Analytics workspace for any members of an Active Directory domain.
- Endpoint Configuration Manager. Import computer collections from Endpoint Configuration Manager into a Log Analytics workspace.
- WSUS. Groups that are created in WSUS servers can be imported into a Log Analytics workspace.

For more information on how to create computer groups for filtering machines for update deployment, refer to [Computer groups in Azure Monitor log queries](#).

## Scalability

Azure Automation can process up to 1,000 computers per update deployment. If you expect to update more than 1,000 computers, you can split up the updates among multiple update schedules. Refer to [Azure subscription and service limits, quotas, and constraints](#).

## Availability

- Currently, mappings between Log Analytics Workspace and Automation Account are supported in several regions. For further information, refer to [Supported regions for linked Log Analytics workspace](#).
- Supported client types: Update assessment and patching is supported on Windows and Linux computers that run in Azure or in your on-premises environment. Currently, the Windows client isn't officially supported. For a list of the supported clients, refer to [Supported client types](#).

## Security

Security provides assurances against deliberate attacks and the abuse of your valuable data and systems. For more information, see [Overview of the security pillar](#).

- Update Management permissions: The Update Management component of Automation and the Log Analytics workspace component of Monitor can use Azure role-based access control (Azure RBAC) with built-in roles from Azure Resource Manager. For segregation of the duties, these roles can be assigned to different users, groups, and security principals. For a list of the roles in Automation accounts, refer to [Manage role permissions and security](#).
- Encryption of sensitive assets in Automation: An Automation account can contain sensitive assets such as credentials, certificates, and encrypted variables that runbooks might use. Each secure asset is encrypted by default using a data encryption key that's generated for each Automation account. These keys are encrypted and stored in Automation with an account encryption key that can be stored in the Azure Key Vault for customers who want to manage encryption with their own keys. By default, an account encryption key is encrypted by using Microsoft-managed keys. Use the following guidelines to [apply encryption of secure assets in Azure Automation](#).
- Runbook permissions for a Hybrid Runbook Worker: By default, runbook permissions for a Hybrid Runbook Worker run in a system context on the machine where they're deployed. A runbook provides its own authentication to local resources. Authentication can be configured using managed identities for Azure resources or by specifying a Run As account to provide a user context for all runbooks.

- Network planning: Hybrid Runbook Worker requires outbound internet access over TCP port 443 to communicate with Automation. For computers with restricted internet access, you can use the [Log Analytics gateway](#) to configure communication with Automation and an Azure Log Analytics workspace.
- Azure Security baseline for Automation: [Azure security baseline for Automation](#) contains recommendations about how to increase overall security to protect your assets following best practice guidance.

## DevOps

- You can schedule update deployment programmatically through the REST API. For more information, refer to [Software Update Configurations - Create](#).
- Azure Automation allows integration with popular source control systems like Azure DevOps and GitHub. With source control, you can integrate an existing development environment that contains your scripts and custom code that has been previously tested in an isolated environment.
- For more information about how to integrate Automation with your source control environment, refer to [Use source control integration](#).

## Cost optimization

Cost optimization is about looking at ways to reduce unnecessary expenses and improve operational efficiencies. For more information, see [Overview of the cost optimization pillar](#).

- Use the [Azure pricing calculator](#) to estimate costs. For more information about Automation pricing models, refer to [Automation pricing](#).
- Azure Automation costs are priced for job execution per minute or for configuration management per node. Every month, the first 500 minutes of process automation and configuration management on five nodes are free.
- An Azure Log Analytics workspace might generate more costs related to the amount of log data that's stored in Azure Log Analytics. The pricing is based on consumption, and the costs are associated with data ingestion and data retention. For ingesting data into Azure Log Analytics, use the capacity reservation or pay-as-you-go model that includes 5 gigabytes (GB) free a month for each billing account. Data retention for the first 31 days is free of charge.
- Use the Azure pricing calculator to estimate costs. For more information about Log Analytics pricing models, refer to [Azure Monitor pricing](#).

## Contributors

*This article is maintained by Microsoft. It was originally written by the following contributors.*

Principal author:

- [Mike Martin](#) | Senior Cloud Solution Architect

*To see non-public LinkedIn profiles, sign in to LinkedIn.*

## Next steps

More about Azure Automation:

- [Hybrid Runbook Worker overview](#)
- [Create an Azure Automation account](#)
- [Pre-requisites: Azure Automation network configuration details](#)
- [Azure Automation Update Management](#)
- [Overview of Log Analytics in Azure Monitor](#)
- [Overview of VM insights](#)
- [Azure Arc Overview](#)
- [What is Azure Arc enabled servers?](#)

## Related resources

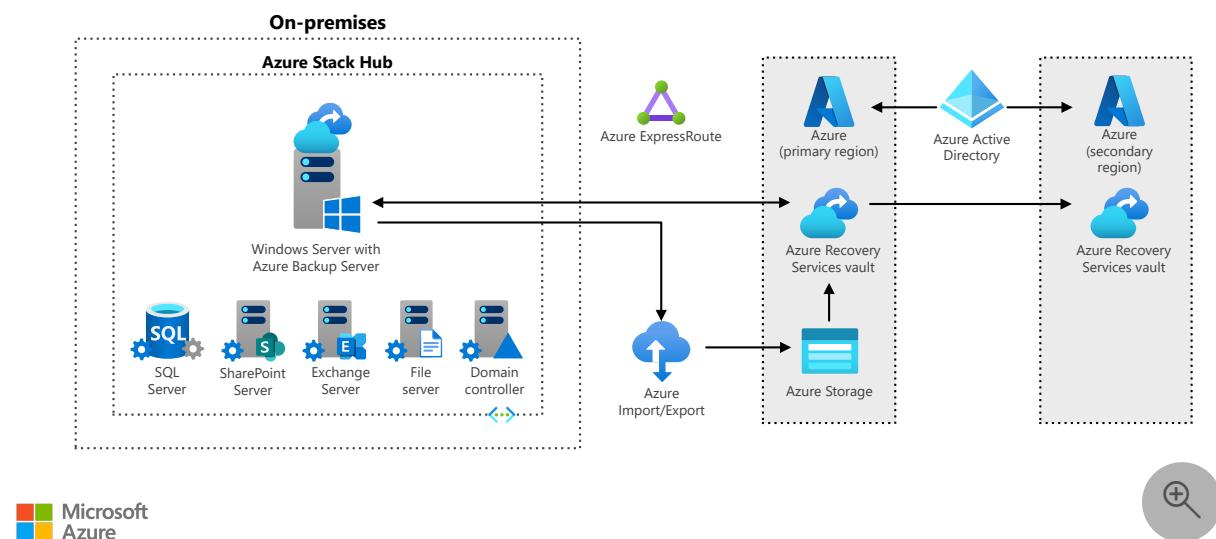
- [Azure Automation in a hybrid environment](#)
- [Event-based cloud automation](#)
- [Azure Automation State Configuration](#)

# Back up files and applications on Azure Stack Hub

Microsoft Entra ID   Azure Backup   Azure ExpressRoute   Azure Stack Hub   Azure Storage

Consider a situation in which Azure Stack Hub hosts virtual machines (VMs) that run user workloads. There's a need to back up and restore the files and applications of the workloads. This reference architecture article describes an approach that delivers an optimized solution for backup and restore activities.

## Architecture



 Microsoft Azure



[Download a Visio file](#) of this architecture.

## Workflow

The cloud components include the following services:

- An Azure subscription that hosts all cloud resources that are included in this solution.
- A Microsoft Entra tenant that's associated with the Azure subscription. It provides authentication of Microsoft Entra security principals to authorize access to Azure resources.
- An Azure Recovery Services vault in the Azure region that's closest to the on-premises datacenter that hosts the Azure Stack Hub deployment.

Depending on the criteria presented in this article, cloud components can also include the following services:

- An Azure ExpressRoute circuit that connects the on-premises datacenter and the Azure region that hosts the Azure Recovery Services vault. The circuit is configured to have Microsoft peering in order to accommodate larger backup sizes.
- The Azure Import/Export service, for enabling MABS offline backups to Azure.

 **Note**

As of 08/20, the MABS offline backup to Azure that uses Azure Data Box is in preview.

Depending on the use of the Azure Import/Export service for MABS offline backup to Azure, the solution might also have an Azure Storage account in the same Azure region as the Recovery Services vault.

The on-premises components include the following services:

- An Azure Stack Hub–integrated system in the connected deployment model that runs the current update (2002 as of August 2020), and located within the customer's on-premises datacenter.
- A Windows Server 2016 or Windows Server 2019 VM that's hosted by the Azure Stack Hub–integrated system and that runs MABS v3 Update Release (UR) 1.
- Azure Stack Hub VMs with the MABS protection agent, which manages backups to and restores from the MABS Azure Stack Hub VM. The MABS protection agent tracks changes to protected workloads, and transfers the changes to the MABS data store. The protection agent also identifies data on its local computer that can be protected and plays a role in the recovery process.
- A Microsoft Azure Recovery Services (MARS) agent that's installed on the server that runs MABS. The agent integrates MABS and Azure Recovery Services vault.

 **Note**

A MARS agent is also referred to as an *Azure Backup agent*.

## Components

- [Azure Stack Hub](#)
- [Microsoft Entra ID](#)
- [Azure storage account](#)
- [Azure ExpressRoute](#)

## Alternatives

The recommended solution that's described in this article isn't the only way to provide backup and restore functionality to user workloads that run on Azure Stack Hub. Customers have other options, including:

- Local backup and restore by using the **Windows Server Backup** feature that's included in the Windows Server operating system. Users can then copy local backups to longer-term storage. This approach supports application-consistent backups by relying on Windows VSS providers, but increases local disk space usage and backup maintenance overhead.
- Backup and restore by using Azure Backup with the locally installed MARS agent. This approach minimizes the use of local disk space and automates the process of uploading backups to cloud-based storage. However, it doesn't support application-consistent backups.
- Backup and restore by using a backup solution that's installed in the same datacenter but outside of Azure Stack Hub. This approach facilitates scenarios that involve an Azure Stack Hub disconnected deployment model.
- Azure Stack Hub-level backup and restore by using disk snapshots. This approach requires that the VM that's being backed up is stopped, which is typically not a viable option for business-critical workloads, but can be acceptable in some scenarios.

## Scenario details

Backup and restore are essential components of any comprehensive business continuity and disaster recovery strategy. Designing and implementing a consistent and reliable backup approach in a hybrid environment is challenging, but can be considerably simplified by integrating with Microsoft Azure services. This applies not only to the workloads that run on traditional on-premises infrastructure, but also to those hosted by third-party public and private cloud providers. However, the benefits of integration with Azure services are evident when the hybrid environments incorporate Azure Stack portfolio offerings, including the Azure Stack Hub.

While one of the primary strengths of Azure Stack Hub is that it supports the platform-as-a-service (PaaS) model, it also helps customers to modernize their existing

infrastructure-as-a-service (IaaS) workloads. Such workloads can include file shares, Microsoft SQL Server databases, Microsoft SharePoint farms, and Microsoft Exchange Server clusters. Migrating them to VMs that run on hyperconverged, highly resilient clusters that have administrative and programming models that are consistent with Microsoft Azure results in minimized management and maintenance overhead.

For implementing backup of files and applications that run on Azure Stack Hub VMs, Microsoft recommends a hybrid approach that relies on a combination of cloud and on-premises components to deliver a scalable, performant, resilient, secure, straightforward to manage, and cost-efficient backup solution. The central component of this solution is Microsoft Azure Backup Server (MABS) v3, which is part of the Azure Backup offering. MABS relies on Azure Stack Hub infrastructure for compute, network, and short-term storage resources, and it uses Azure-based storage to serve as the long-term backup store. This approach minimizes or eliminates the need to maintain physical backup media such as tapes.

 **Note**

MABS is based on Microsoft System Center Data Protection Manager (DPM) and provides similar functionality with just a few differences. However, DPM is not supported for use with Azure Stack Hub.

## Core functionality

The proposed solution supports the following functionality on Azure Stack Hub VMs that run Windows Server 2019, 2016, 2012 R2, 2012, 2008 R2 SP1 (64-bit with Windows Management Framework 4.0), 2008 SP2 (64-bit with Windows Management Framework 4.0), and Windows 10 (64-bit):

- Backup and restore of New Technology File System (NTFS) and Resilient File System (ReFS) volumes, shares, folders, files, and system state.
- Backup and restore of SQL Server 2019, 2017, 2016 (with required service packs (SPs)), and 2014 (with required SPs) instances and their databases.
- Backup and restore of Exchange Server 2019 and Exchange Server 2016 servers and databases, including standalone servers and databases in a database availability group (DAG).
- Restore of individual mailboxes and mailbox databases in a DAG.

- Backup and restore of SharePoint 2019 and SharePoint 2016 (with the latest SPs) farms and front-end web server content.
- Restore of SharePoint 2019 and SharePoint 2016 databases, web applications, files, list items, and search components.

 **Note**

To deploy Windows 10 client operating systems on Azure Stack Hub, you must have Windows per-user licensing or have purchased it through a Qualified Multitenant Hoster (QMTH).

MABS implements the disk-to-disk-to-cloud (D2D2C) backup scheme, with the primary backup stored locally on the server that hosts the MABS installation. Local backups are then copied to an Azure Site Recovery vault. The local disk functions as short-term storage, while the vault provides long-term storage.

 **Note**

Unlike DPM, MABS doesn't support tape backups.

The backup process consists of the following four stages:

1. You install the MABS protection agent on a computer that you want to protect, and add it to a protection group.
2. You set up protection for the computer or its app, including backup to MABS local disks for short-term storage and to Azure for long-term storage. As part of the setup, you specify the backup schedule for both types of backups.
3. The protected workload backs up to the local MABS disks according to the schedule that you specified.
4. The local backup that's stored on MABS disks is backed up to the Azure Recovery Services vault by the MARS agent that runs on the MABS server.

## Prerequisites

Implementing the recommended solution depends on meeting the following prerequisites:

- Access to an Azure subscription, with permissions that are sufficient to provision an Azure Recovery Services vault in the Azure region that's closest to an on-premises datacenter that hosts the Azure Stack Hub deployment.

- An Active Directory Domain Services (AD DS) domain that's accessible from an Azure Stack Hub VM that will host a MABS instance.
- An Azure Stack Hub-hosted VM that will run a MABS instance, satisfying the prerequisites that are listed in [Install Azure Backup Server on Azure Stack](#) and with outbound connectivity to URLs that are listed in [DPM/MABS networking support](#).

① **Note**

Additional disk space and performance considerations for MABS are described in more detail later in this article.

① **Note**

To validate whether the VM that hosts MABS has connectivity to the Azure Backup service, you can use the **Get-DPMCloudConnection** cmdlet (included in the Azure Backup Server PowerShell module).

① **Note**

MABS also requires a local instance of SQL Server. For details regarding SQL Server requirements, see [Install and upgrade Azure Backup Server](#).

## Data types

From the perspective of MABS, there are two data types to consider:

- *File data* is data that typically resides on file servers (such as Microsoft Office files, text files, or media files), and that needs to be protected as flat files.
- *Application data* is data that exists on application servers (such as Exchange storage groups, SQL Server databases, or SharePoint farms) and that requires MABS to be aware of the corresponding application requirements.

① **Note**

As an alternative to file data backup with MABS, it's possible to install the MABS agent directly on an Azure Stack Hub VM and back up its local file system directly to an Azure Recovery Services vault. However, unlike MABS, this approach does not

provide centralized management and always relies on cloud-based storage for backups and restores.

## Backup types

Whether you're protecting file data or application data, protection begins with creating a replica of the data source in the local storage of a MABS instance. The replica is synchronized or updated at regular intervals according to the settings that you configure. The method that MABS uses to synchronize the replica depends on the type of data that's being protected. If a replica is identified as being inconsistent, MABS performs a consistency check, which is a block-by-block verification of the replica against the data source.

For a file volume or share on a server, after the initial full backup, the MABS protection agent uses a volume filter and change journal to determine which files have changed. It then performs a checksum procedure for those files to synchronize only the changed blocks. During synchronization, these changes are transferred to MABS and then applied to the replica, thereby synchronizing the replica with the data source.

If a replica becomes inconsistent with its data source, MABS generates an alert that specifies which computer and which data sources are affected. To resolve the issue, you can repair the replica by initiating a synchronization with consistency check on the replica. During a consistency check, MABS performs a block-by-block verification and repairs the replica to return it to consistency with the data source. You can schedule a daily consistency check for protection groups or initiate a consistency check manually.

At regular intervals that you can configure, MABS creates a recovery point for the protected data source. A *recovery point* is a version of the data that can be recovered.

For application data, after the replica is created by MABS, changes to volume blocks that belong to application files are tracked by the volume filter. How changes are transferred to the MABS server depends on the application and the type of synchronization. The operation that's labeled *synchronization* in MABS Administrator Console is analogous to an incremental backup, and it creates a transactionally consistent and accurate reflection of the application data when combined with the replica.

During the type of synchronization that's labeled *express full backup* in MABS Administrator Console, a full Volume Shadow Copy Service (VSS) snapshot is created, but only the changed blocks are transferred to the MABS server.

Each express full backup creates a recovery point for application data. If the application supports incremental backups, each synchronization also creates a recovery point. The

synchronization process is application-dependent:

- For Exchange data, synchronization transfers an incremental VSS snapshot by using the Exchange VSS writer. Recovery points are created for each synchronization and for each express full backup.
- SQL Server databases that are log-shipped, that are in read-only mode, or that use the simple recovery model, don't support incremental backup. Recovery points are created for each express full backup only. For all other SQL Server databases, synchronization transfers a transaction log backup, with recovery points that are created for each incremental synchronization and express full backup. The transaction log is a serial record of all transactions that have been performed against the database since the transaction log was last backed up.
- SharePoint servers don't support incremental backup. Recovery points are created for each express full backup only.

Incremental synchronizations require less time than it takes to do an express full backup. However, the time that's required to recover data increases as the number of synchronizations increases. This is because MABS must restore the last full backup and then restore and apply all the incremental synchronizations up to the point in time that's specified for recovery.

To enable faster recovery time, MABS regularly performs an express full backup, which updates the replica to include the changed blocks. During the express full backup, MABS takes a snapshot of the replica before it updates the replica by using the changed blocks. To enable more frequent RPOs and to reduce the data loss window, MABS also performs incremental synchronizations in the time between two express full backups.

As with file data protection, if a replica becomes inconsistent with its data source, MABS generates an alert that specifies which server and which data sources are affected. To resolve the inconsistency, you can repair the replica by initiating a synchronization with consistency check on the replica. During a consistency check, MABS performs a block-by-block verification of the replica and repairs it to bring it back into consistency with the data sources. You can schedule a daily consistency check for protection groups or initiate a consistency check manually.

## Protection policies

A computer or its workload becomes protected when you install the MABS protection agent software on the computer and add the data of the computer or its workload to a protection group. Protection groups are used to configure and manage the protection of data sources on computers. A *protection group* is a collection of data sources that share the same protection configuration. The *protection configuration* is the collection of

settings that are common to a protection group, such as the protection group name, protection policy, storage allocations, and replica creation method.

MABS stores a separate replica of each protection group member in the storage pool. A protection group member can contain such data sources as:

- A volume, share, or folder on a file server or server cluster.
- A storage group of an Exchange server or server cluster.
- A database of an instance of SQL Server or server cluster.

For each protection group, you configure a protection policy that's based on your recovery goals for that protection group. *Recovery goals* represent data protection requirements that correspond to the RTOs and RPOs of your organization. Within MABS, they're defined based on a combination of the following parameters:

- Short-term retention range. This determines how long you want to retain backed up data on the local MABS storage.
- Synchronization and recovery point frequencies. This corresponds directly to data loss tolerance, which in turn reflects the RPOs of your organization. It also determines how often MABS should synchronize its local replicas with protected data sources by collecting their data changes. You can set the synchronization frequency to any interval between 15 minutes and 24 hours. You can also select to synchronize just before a recovery point is created rather than on a specified time schedule.
- Short-term recovery point schedule. This establishes how many recovery points should be created in the local storage for the protection group. For file protection, you select the days and times for which you want recovery points created. For data protection of applications that support incremental backups, the synchronization frequency determines the recovery point schedule.
- Express full backup schedule. This is the recovery point schedule for data protection of applications that don't support incremental backups and do support express full backups.
- Online backup schedule. This determines the frequency of creating a copy of local backups in the Azure Recovery Services vault that's associated with the local MABS instance. You can schedule on a daily, weekly, monthly, or yearly basis, with maximum allowed frequency of two backups per day. MABS automatically creates a recovery point for online backups by using the most recent local replica, without transferring new data from the protected data source.

(!) Note

A Recovery Services vault supports as many as 9,999 recovery points.

- Online retention policy. This specifies the time period during which daily, weekly, monthly, and yearly backups are retained in the Azure Site Recovery vault that's associated with the local MABS instance.

(!) Note

To protect the latest content of the data source online, create a new recovery point on the local disk before creating an online recovery point.

(!) Note

By default, Azure Recovery Services vault is *geo-redundant*, meaning that any backup that's copied to its storage is automatically replicated to an Azure region that's part of a pre-defined region pair. You can change the replication settings to locally redundant if that's sufficient for your resiliency needs and if you need to minimize storage costs. However, you should consider keeping the default setting. This option can't be changed if the vault contains any protected items.

(!) Note

For the listing of Azure region pairs, see [Business continuity and disaster recovery \(BCDR\): Azure Paired Regions](#).

## Testing restores

In addition to an optimally designed and implemented backup strategy, it's equally important to define, document, and test the restore process for each type of protected workload. While MABS provides built-in consistency checks that automatically verify the integrity of data backups, the testing of restores should be part of routine operating procedures. The testing validates a restore by examining the state of restored workloads. The testing results should be available to workload owners.

In general, the testing of restores tends to be challenging because it requires an environment that closely resembles the one that hosts the protected workloads. Azure Stack Hub, with its built-in DevOps and infrastructure as code capabilities, greatly simplifies addressing this challenge.

## Roles and responsibilities

Planning for and implementing backup and restore of Azure Stack Hub-based workloads typically involves interaction among many stakeholders:

- Azure Stack Hub operators. Azure Stack Hub operators manage Azure Stack Hub infrastructure, ensuring that there are sufficient compute, storage, and network resources for implementing a comprehensive backup and restore solution, and making these resources available to tenants. They also collaborate with application and data owners to help determine the optimal approach to deploying their workloads to Azure Stack Hub.
- Azure administrators. Azure administrators manage the Azure resources that are needed to implement hybrid backup solutions.
- Microsoft Entra administrators. Microsoft Entra administrators manage Microsoft Entra resources, including user and group objects that are used to provision, configure, and manage Azure resources.
- Azure Stack Hub tenant IT staff. These stakeholders design, implement, and manage MABS, including the MABS backups and restores.
- Azure Stack Hub users. These users provide RPO and RTO requirements and submit requests to back up and restore data and applications.

## Considerations

These considerations implement the pillars of the Azure Well-Architected Framework, which is a set of guiding tenets that can be used to improve the quality of a workload. For more information, see [Microsoft Azure Well-Architected Framework](#).

## Reliability

Reliability ensures your application can meet the commitments you make to your customers. For more information, see [Overview of the reliability pillar](#).

Azure Stack Hub helps increase workload availability because of the resiliency that's inherent to its infrastructure. You can further increase availability by designing and implementing solutions that extend the scope of workload protection. This is the added

value that MABS provides. In the context of MABS that runs on Azure Stack Hub, there are two aspects of availability that need to be explored in more detail:

- The availability of MABS and its data stores
- The availability of the point-in-time restore capability of MABS-protected workloads

You need to consider both of these when you develop a backup strategy that's driven by recovery point objectives (RPOs) and recovery time objectives (RTOs). RTO and RPO represent continuity requirements that are stipulated by business functions within an organization. RPO designates a time period that represents the maximum acceptable data loss due to an incident that makes the data unavailable for a time. RTO designates the maximum acceptable duration of time it can take to reinstate access to business functions after an incident that makes the functions unavailable.

 **Note**

To address the RTO requirements for Azure Stack Hub workloads, you should account for recovery of the Azure Stack infrastructure, user VMs, applications, and user data. In this article we consider only the last two of these, applications and user data, although we also present considerations regarding availability of the Modern Backup Storage (MBS) functionality.

The availability of MABS and its data stores is contingent on the availability of the VM that hosts the MABS installation and its local and cloud-based storage. Azure Stack Hub VMs are highly available by design. If there's a MABS failure, you have the ability to restore Azure Backup-protected items from any other Azure Stack Hub VM that hosts MABS. Note, however, that for a server that hosts MABS to recover backups that were done by using MABS that runs on another server, both servers must be registered with the same Azure Site Recovery vault.

 **Note**

In general, you can deploy another instance of MABS and configure it to back up the primary MABS deployment. This is similar to the primary-to-secondary protection, chaining, and cyclic protection configurations that are available when you use DPM. However, this approach is not supported for MABS and it doesn't yield meaningful availability advantages in the scenario that's described in this article.

The point-in-time restore capability of MABS-protected workloads is dependent to a large extent on the type of data, its backups, and its protection policies. To understand these dependencies, it's necessary to explore these concepts in more detail.

## Security

Security provides assurances against deliberate attacks and the abuse of your valuable data and systems. For more information, see [Overview of the security pillar](#).

Managing user data and applications in hybrid scenarios warrants additional security considerations. These considerations can be grouped into the following categories:

- Backup encryption
- Azure Recovery Services vault protection

MABS and Azure Backup enforce the encryption of backups at rest and in transit:

- Encryption at rest. During the installation of MABS, the user provides a passphrase. This passphrase is then used to encrypt all backups before they're uploaded to an Azure Recovery Services vault. Decryption takes place only after backups are downloaded from that vault. The passphrase is available only to the user who created it and to the locally installed MARS agent. It's critical to ensure that the passphrase is stored in a secure location, because it serves as the authorization mechanism when performing cloud-based restores on a MABS server other than the one where the backups took place.
- Encryption in transit. MABS v3 relies on Transport Layer Security (TLS) protocol version 1.2 to protect its connections to Azure.

Azure Recovery Services vault offers mechanisms that further protect online backups, including:

- Azure role-based access control (Azure RBAC). Azure RBAC allows for delegating and segregating responsibilities according to the principle of least privilege. There are three Azure Backup-related built-in roles that restrict access to backup management operations:
  - Backup Contributor. Provides access to create and manage backups, except for deleting Recovery Services vault and delegating access to others.
  - Backup Operator. Provides access that's equivalent to that of the Backup Contributor, except for removing backups and managing backup policies.
  - Backup Reader. Provides access to monitor backup management operations.
- Azure Resource Locks. You can create and assign read-only and delete locks to an Azure Site Recovery vault to mitigate the risk of the vault being accidentally or maliciously changed or deleted.

- Soft delete. Soft delete helps protect vault and backup data from accidental or malicious deletions. With soft delete, if a user deletes a backup item, the corresponding data is retained for 14 days, allowing for its recovery with no data loss during that period. The 14-day retention of backup data in the soft delete state doesn't incur any cost. Soft delete is enabled by default.
- Protection of security-sensitive operations. The Azure Recovery Services vault automatically implements another layer of authentication whenever a security-sensitive operation, such as changing a passphrase, is attempted. This extra validation helps ensure that only authorized users do these operations.
- Suspicious activity monitoring and alerts. Azure Backup provides built-in monitoring and alerting of security-sensitive events that are related to Azure Backup operations. Backup reports facilitate usage tracking, auditing of backups and restores, and identifying key backup trends.

## Cost optimization

Cost optimization is about looking at ways to reduce unnecessary expenses and improve operational efficiencies. For more information, see [Overview of the cost optimization pillar](#).

When considering the cost of the backup solution that's described in this article, remember to account for both on-premises and cloud-based components. The pricing of on-premises components is determined by the Azure Stack Hub pricing model. As with Azure, Azure Stack Hub offers a pay-as-you-use arrangement that's available through enterprise agreements and the Cloud Solution Provider program. This arrangement includes a monthly price per Windows Server VM. If you can use existing Windows Server licenses, you can significantly reduce that cost down to the base VM pricing. MABS relies on SQL Server as its data store, but there's no licensing cost associated with running that SQL Server instance if it's only used for MABS.

There are Azure-related charges for use of the following resources:

- Azure Backup. Pricing for Azure Backup is largely determined by the number of protected workloads and the size of data backups (before compression and encryption) for each. The cost is also affected by the choice between locally redundant storage (LRS) and geo-redundant storage (GRS) for the replication of the Azure Recovery Services vault content. For details, see [Azure Backup pricing](#).
- Azure ExpressRoute. Azure ExpressRoute pricing is based on one of two models:
  - Unlimited data. This is a monthly fee with all inbound and outbound data transfers included.
  - Metered data. This is a monthly fee with all inbound data transfers free of charge and outbound data transfers charged per gigabyte.

- Azure Import/Export. The cost for Azure Import/Export includes a flat, per-device fee for device handling.
- Azure Storage. When you use Azure Import/Export, standard Azure Storage rates and transaction fees apply.

Without ExpressRoute, you might have to account for increased bandwidth usage of your internet connections for backups and restores. The cost will vary depending on many factors, including geographical area, current bandwidth usage, and the internet service provider.

## Operational excellence

Operational excellence covers the operations processes that deploy an application and keep it running in production. For more information, see [Overview of the operational excellence pillar](#).

## Manageability

One of the primary factors that affects that affects your backup and restore strategy is the configuration of protection groups and the criteria you use to decide which protected workloads should belong to the same protected groups. As described earlier in this article, a protection group is a collection of data sources such as volumes, shares, or application data stores that have common backup and restore settings. When defining a protection group, you need to specify:

- Data sources, such as servers and workloads that you want to protect.
- Back-up storage, including short-term and long-term protection settings.
- Recovery points, which are points in time from which backed up data can be recovered.
- Allocated disk space, which is the amount of disk space from the storage pool that's allocated for backups.
- Initial replication, which is the method used for the initial backup of data sources. The method can be an online transfer (over a network) or an offline transfer (such as via the Azure Import/Export service).
- The consistency checking method, which is the method of verifying the integrity of data backups.

The following methods are often used to decide which protected workloads should belong to the same protected groups:

- By computer. This method combines all data sources for a computer into the same protection group.

- By workload. This method separates files and each application data type into different protection groups. However, recovering a server hosting multiple workloads might require multiple restores from different protection groups.
- By RPO and RTO. This method groups data sources with similar RPOs. You control the RPO by setting the synchronization frequency for the protection group, which determines the amount of potential data loss (measured in time) during unexpected outages. In the scenario that's described in this article, you control the RTO by setting the retention period within the short-term storage. This determines the period during which backups can be restored from the local short-term storage, rather than from the cloud-based long-term storage. Backing up from the local short-term storage results in a faster restore.
- By data characteristics. This method accounts for the frequency of data changes, data growth rate, or its storage requirements as the criteria for groupings.

When naming protection groups, use unique, meaningful names. A name can be any combination of alphanumeric characters and spaces up to 64 characters in length.

When you create a protection group, you choose a method for creating the initial replica. The initial replication copies all the data that's selected for protection to the server that hosts MABS, and then copies it to the Azure Site Recovery vault. Both copies are checked for consistency. MABS can create replicas automatically over the network, but you can create replicas manually by backing up, transferring, and restoring data offline.

For information about choosing the replica creation method, see [Initial replication over the network](#). The article has a table that provides estimates of how long MABS takes to create a replica automatically over the network for various protected data sizes and network speeds.

The offline-seeding process supports use of the Azure Import/Export service, which can transfer data to an Azure Storage account by using SATA disks. This capability can be used when online backup is too slow because of the amount of backup data or the speed of the network connection to Azure.

The offline-seeding workflow has the following steps:

1. You copy the initial backup data to one or more SATA disks by using the `AzureOfflineBackupDiskPrep` tool.
2. The tool automatically generates an Azure Import job and a Microsoft Entra app in the subscription that hosts the target Azure Storage account and Azure Recovery Services vault. The app provides Azure Backup with secure and scoped access to the Azure Import Service, as required by the offline seeding process.

3. You ship the disks to the Azure datacenter that hosts the target Azure Storage account.
4. The Azure datacenter staff copies data from the disks to the Azure Storage account.
5. The workflow triggers a copy from the Azure Storage account to the Azure Recovery Services vault.

## DevOps

Although backup and restore are considered to be part of IT operations, there are some DevOps-specific considerations that are worth incorporating into a comprehensive backup strategy. Azure Stack Hub facilitates automated deployment of various workloads, including VM-based applications and services. You can use this capability to streamline MABS deployment to Azure Stack Hub VMs, which simplifies the initial setup in multitenant scenarios. By combining Azure Resource Manager templates, VM extensions, and the DPM PowerShell module, it's possible to automate the configuration of MABS, including the setup of its protection groups, retention settings, and backup schedules. In the spirit of the best practices of DevOps, you should store templates and scripts in a source control facility, and configure their deployment by using pipelines. These practices help to minimize the recovery time in cases where the infrastructure that's needed to restore file and application data has to be recreated.

## Performance efficiency

Performance efficiency is the ability of your workload to scale to meet the demands placed on it by users in an efficient manner. For more information, see [Performance efficiency pillar overview](#).

When you plan to deploy MABS on Azure Stack Hub, you need to consider the amount of processing, storage, and network resources that are allocated to the VMs that host the deployment. Microsoft recommends allocating a 2.33 gigahertz (GHz) quad-core CPU to satisfy MABS processing needs, and about 10 GB of disk space to accommodate the installation binaries. Other storage requirements can be categorized as follows:

- Disk space for backups. The general recommendation for backup disk space is to allocate a storage pool of disk space that's equivalent to about 1.5 times the size of all data that's to be backed up. After the disks are attached to the VM, MABS manages volume and disk space management. The number of disks that you can attach to a VM depends on its size.

### Note

You should not store backups locally for more than five days. Backups older than five days should be offloaded to the Azure Site Recovery vault.

- Disk space for MARS agent cache location. Consider using drive **C** on the VM that hosts the MABS installation.
- Disk space for local staging area during restores. Consider using the temporary drive **D** on the VM that hosts the MABS installation.

To provision storage for the VM that hosts the MABS installation, use managed disks in the Premium performance tier. The expected performance characteristics are 2,300 I/O operations per second (IOPS) and 145 MB/s per disk. Unlike Azure, there are no performance guarantees for Azure Stack Hub.

To obtain a more accurate estimate of the storage that's required to accommodate Azure Stack Hub-based workload backups, consider using the Azure Stack VM Size Calculator for MABS, which is available from [Microsoft Downloads](#). The calculator is implemented as a Microsoft Excel workbook that has macros that derive optimum Azure Stack Hub sizing information that's based on a number of parameters that you provide. These parameters include:

- Source details that include a list of the VMs to be protected, including for each:
  - The size of the protected data
  - The workload type (Windows Server, SharePoint, or SQL Server)
- Data retention range, in days

Each workload type is, by default, associated with a predefined daily rate of changes (or *churn*). You can adjust these values if they don't reflect the usage patterns in your environment.

The Azure Stack VM Size Calculator for MABS uses the information that you specify to provide:

- An estimated size of the Azure Stack Hub VM that hosts the installation of MABS.
- An estimated amount of MABS disk space that's needed to host backed up data.
- A total number of disks at 1 terabyte (TB) each.
- The IOPS rate that's available for MABS usage.
- An estimated time to complete the initial backup. The estimate is based on the total size of the protected data and on the IOPS available for MABS usage.
- An estimated time to complete daily backups. The estimate is based on the total size of daily churn and on the IOPS available for MABS usage.

 **Note**

Azure Stack VM Size Calculator for MABS was released in April 2018, which means that it doesn't take into account optimizations incorporated into MABS v3 (including those included in UR1). However, it does include enhancements that are specific to MBS, which was introduced in MABS v2 released in June 2017.

If you create a protection group by using the MABS graphical interface, whenever you add a data source to a protection group, MABS calculates the local disk space allocation that's based on the short-term recovery goals that you specify. You can then decide how much space to allocate in the storage pool for replicas and recovery points for each data source in the group. You need to ensure that there's sufficient space on the local disks of the protected servers for the change journal. MABS provides default space allocations for the members of the protection group. For details regarding the default space allocations for different MABS components, see [Deploy protection groups documentation](#).

Consider using the default space allocations unless you know that they don't meet your needs. Overriding the default allocations can result in allocation of too little or too much space. Allocation of too little space for the recovery points can prevent MABS from storing enough recovery points to meet your retention range objectives. Allocation of too much space wastes disk capacity. After creating a protection group, if you allocated too little space for a data source, you can increase the allocations for the replica and recovery point volumes for each data source. If you allocated too much space for the protection group, you can remove the data source from the protection group and delete the replica. Then add the data source to the protection group with smaller allocations.

After deployment, if you need to adjust the estimated sizing of the Azure Stack Hub VMs that host MABS in order to accommodate changes in processing or storage requirements, you have three options:

- Implement vertical scaling. This requires that you modify the amount and type of processor, memory, and disk resources of the Azure Stack Hub VMs that host MABS.
- Implement horizontal scaling. This requires provisioning or deprovisioning the Azure Stack Hub VMs that have MABS installed to match the processing demands of the protected workloads.
- Modify protection policies. This requires changing the parameters of the protection policies, including retention range, recovery point schedule, and express full backup schedule.

 **Note**

MABS is subject to limits in regard to the number of recovery points, express full backups, and incremental backups. For details regarding these limits, see [Recovery process](#).

If you opt to automatically grow volumes, then MABS accounts for the increased backup volume as the production data grows. Otherwise, MABS limits the backup storage to the size of data sources in the protection group.

There are two main options to adjust available bandwidth:

- Increase the VM size. For Azure Stack Hub VMs, the size determines maximum network bandwidth. However, there are no bandwidth guarantees. Instead, VMs can use the amount of available bandwidth up to the limit determined by their size.
- Increase the throughput of uplink switches. Azure Stack Hub systems support a range of hardware switches that offer several choices of uplink speeds. Each Azure Stack Hub cluster node has two uplinks to the top-of-rack switches for fault tolerance. The system allocates half of the uplink capacity for critical infrastructure, and the remainder is shared capacity for Azure Stack services and all user traffic. Systems that are deployed with faster speeds have more bandwidth available for backup traffic.

Although it's possible to segregate network traffic by attaching a second network adapter to a server, all Azure Stack Hub VM traffic to the internet shares the same uplink. A second virtual network adapter doesn't segregate traffic at the physical transport level.

To accommodate larger backup sizes, you can consider using Azure ExpressRoute with Microsoft peering to connect between Azure Stack Hub virtual networks and Azure Recovery Services vault. Azure ExpressRoute extends on-premises networks into the Microsoft cloud over a private connection that's supplied by a connectivity provider. You can purchase ExpressRoute circuits for a wide range of bandwidths, from 50 Mbps to 10 Gbps.

#### Note

For details about implementing Azure ExpressRoute in Azure Stack Hub scenarios, see [Connect Azure Stack Hub to Azure using Azure ExpressRoute](#).

#### Note

MABS v3 uses enhancements that are built into MBS, and optimizes network and storage usage by transferring only changed data during consistency checks.

## Summary

Azure Stack Hub is a unique offering that differs in many aspects from other virtualization platforms. As such, it warrants special consideration in regard to business continuity strategies for workloads that run on its VMs. Using Azure services simplifies the design and implementation strategy. In this article, we explored using MABS for backing up file and application data on Azure Stack Hub VMs in the connected deployment model. This approach allows customers to benefit from the resiliency and manageability of Azure Stack Hub, and from the hyperscale and global presence of the Azure cloud.

The backup solution that's described here focuses exclusively on file and application data on Azure Stack Hub VMs. This is just a part of an overall business continuity strategy that should account for various other scenarios that affect workload availability. Some examples are: localized hardware and software failures, system outages, catastrophic events, and large-scale disasters.

## Next steps

- [How-to guides - Backup Storage Accounts on Azure Stack](#)
- [How-to guides - Backup of VMs on Azure Stack Hub using Commvault](#)
- [Backup Cloud and On-Premises workloads to Cloud](#)
- [Install Azure Backup Server](#)
- [Backup files and applications on Azure Stack](#)
- [Backup a SharePoint farm on Azure Stack](#)
- [Backup a SQL Server in Azure Stack](#)

## Related resources

Related hybrid guidance:

- [Hybrid architecture design](#)
- [Azure hybrid options](#)
- [Hybrid app design considerations](#)
- [Deploy a hybrid app with on-premises data that scales cross-cloud](#)

Related architectures:

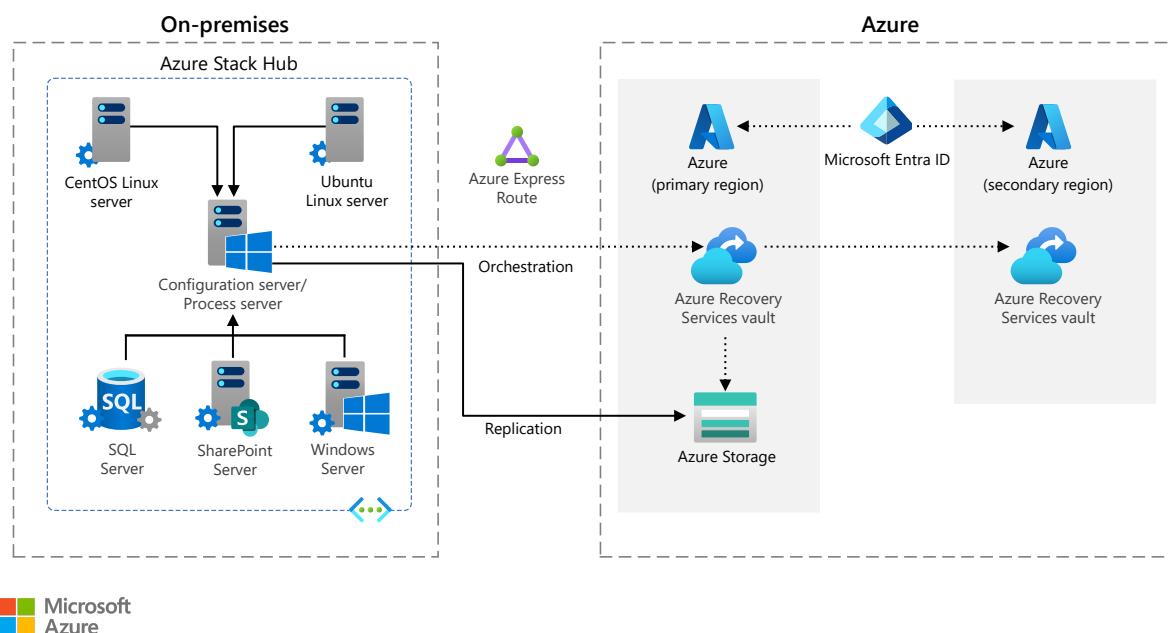
- Disaster Recovery for Azure Stack Hub VMs
- Backup on premises applications and data to the cloud

# Disaster recovery for Azure Stack Hub virtual machines

Microsoft Entra ID   Azure Site Recovery   Azure Stack   Azure Stack Hub   Azure Virtual Network

This article describes the architecture and design considerations of a solution that delivers an optimized approach to the disaster recovery of workloads that run on virtual machines (VMs) that are hosted on Azure Stack Hub.

## Architecture



Download a [Visio file](#) of this architecture.

## Workflow

The cloud components of the proposed solution include the following services:

- An Azure subscription that hosts all cloud resources that are part of this solution.
- An [Microsoft Entra ID](#) tenant associated with the Azure subscription that provides authentication of Microsoft Entra security principals to authorize access to Azure resources.
- An [Azure Recovery Services](#) vault in the Azure region that's closest to an on-premises datacenter that hosts the Azure Stack Hub deployment.

(!) **Note**

The choice of the Azure region that's closest to the on-premises datacenter is specific to the sample scenario that's included in this article. From a disaster recovery standpoint, it's better to select an Azure region that's further away from the location that hosts the production environment. The decision, however, can depend on other factors, such as the need to minimize the latency of regional data feeds or to satisfy data residency requirements.

- An [Azure ExpressRoute](#) circuit that connects the on-premises datacenters to the Azure region that hosts the Azure Recovery Services vault, configured with private peering and Microsoft peering. The former ensures that latency requirements are met after a failover. The purpose of the latter is to minimize the amount of time that it takes to replicate changes between the on-premises workloads and the failover site in Azure.
- An [Azure Storage](#) account that holds blobs that contain the VHD files that are created by replication of the operating system and data volumes of protected [Azure Stack Hub VMs](#). These VHD files serve as the source for the managed disks of Azure VMs that are automatically provisioned after a failover.
- An [Azure virtual network](#) that hosts the disaster recovery environment. It's configured in a manner that mirrors the virtual network environment in the Azure Stack Hub that hosts the production workloads, including components such as load balancers and network security groups. This virtual network is typically connected to the Azure Stack Hub virtual network via an ExpressRoute connection to facilitate workload-level recovery.

(!) **Note**

Sometimes a site-to-site VPN connection suffices in scenarios where Recovery Point Objective (RPO) requirements are less stringent.

- An isolated Azure virtual network intended for test failovers, configured in a manner that mirrors the virtual network environment in Azure Stack Hub hosting the production workloads, including components such as load balancers and network security groups.

The on-premises components of the proposed solution include the following services:

- An Azure Stack Hub integrated system in the connected deployment model. The system runs the current update (2002 as of 9/20) and is in the customer's on-premises datacenter.

- An Azure Stack Hub subscription and a virtual network, or multiple peered virtual networks, that hosts all the on-premises VMs for this solution.
- Azure Site Recovery configuration and process servers that run on Windows Server 2016 or 2012 R2 Azure Hub Stack VMs. The servers manage communications with the Azure Recovery Services vault, and the routing, optimization, and encryption of replication traffic.

 **Note**

By default, a configuration server hosts a single process server. You can deploy dedicated process servers to accommodate a larger volume of replication traffic.

- The Azure Stack Hub VMs that are to be protected. They run supported versions of the Windows Server, CentOS, and Ubuntu operating systems.
- The Site Recovery Mobility service (also referred to as *mobility agent*) that runs on protected VMs. It tracks changes to local disks, records the changes in replication logs, and replicates the logs to the process server. The process server routes them to the target Azure storage account. The logs are used to create recovery points for managed disks that are implemented by using blobs that are stored in the Azure storage account that you designate.

## Components

- [Microsoft Entra ID](#)
- [Azure Virtual Network](#)
- [Azure Recovery Services](#)
- [Azure ExpressRoute](#)
- [Azure Blob Storage](#)
- [Azure Stack Hub](#)

## Alternatives

The recommended solution that's described in this article isn't the only way to provide disaster recovery functionality for Azure Stack Hub VM-based workloads. Customers have other options, including:

- A failover to another Azure Stack Hub stamp. Users that need to protect against a datacenter or site outage might be able to use another Azure Stack Hub deployment to implement disaster recovery provisions. With primary and secondary locations, users can deploy applications in an active/passive configuration across two environments. For less critical workloads, it might be acceptable to use unused capacity in the

secondary location to perform on-demand restoration of applications from backup. You can also implement a recovery site in another datacenter, which, in turn, uses Site Recovery to provision a replica of the recovery site in Azure. Several factors determine whether the use of Site Recovery with Azure serving as the failover site is a viable solution. These factors include government regulations, corporate policies, and latency requirements.

 **Note**

As of July 2020, Site Recovery doesn't support this scenario, which means that the implementation has to use a partner or in-house solution.

- **Back up and restore.** Backing up your applications and datasets makes it possible for you to recover quickly from downtime that results from data corruption, accidental deletions, or localized outages. For Azure Stack Hub VM-based applications, you can use an in-guest agent to protect application data, operating system configurations, and data that's stored on volumes. Backing up a VM by using a guest OS agent typically includes capturing operating system configurations, files, folders, volumes, application binaries, and application data. Recovering an application from an agent requires recreation of the VM, followed by installation of the operating system and the guest agent. At that point, you can restore data into the guest OS.
- **Backup of disk snapshots.** It's possible to use snapshots to capture an Azure Stack Hub VM configuration and the disks that are attached to a stopped VM. This process requires backup products that integrate with Azure Stack Hub APIs to capture VM configuration and create disk snapshots.

 **Note**

As of July 2020, using disk snapshots for a running VM isn't supported. Creating a snapshot of a disk that's attached to a running VM might degrade the performance or affect the availability of the operating system or of the application in the VM.

- **Back up and restore VMs** by using an external backup solution in the same datacenter, and then replicate the backups to another location. In this way, you can restore Azure Stack Hub VMs to the same Azure Stack Hub instance, or to a different one, or to Azure.

## Scenario details

Azure Stack Hub includes self-healing functionality, providing auto-remediation in a range of scenarios that involve localized failures of its components. However, large-scale failures, including outages that affect server racks or site-level disasters, require additional considerations. These considerations should be part of the business continuity and disaster recovery strategy for VM-based user workloads. This strategy must also account for recovery of the Azure Stack infrastructure, which is separate from workload recovery.

Traditional on-premises workload recovery solutions are complex to configure, expensive and labor-intensive to maintain, and difficult to automate, especially when you use another on-premises location as the failover site. Microsoft recommends an alternative solution that relies on a combination of cloud and on-premises components to deliver resilient, performance-based, highly automated, and straightforward ways to implement, secure, and manage a cost-efficient disaster recovery strategy. The core element of this solution is Site Recovery, with the failover site residing in Azure.

## Potential use cases

Site Recovery with Azure as the failover site eliminates all of the aforementioned drawbacks. You can use its capabilities to protect both physical and virtual servers, including those running on either Microsoft Hyper-V or VMware ESXi virtualization platforms. You can also use the same capabilities to facilitate the recovery of workloads that run on Azure Stack Hub VMs.

## Core functionality

Site Recovery is a disaster recovery solution that facilitates protection of physical and virtual computers by providing two sets of features:

- Replication of changes to computer disks that are between the production and disaster recovery locations
- Orchestration of failover and failback between these two locations

Site Recovery offers three types of failovers:

- Test failover. This failover gives you the opportunity to validate your Site Recovery configuration in an isolated environment, without any data loss or impact to the production environment.
- Planned failover. This failover gives you the option to initiate disaster recovery without data loss, typically as part of planned downtime.
- Unplanned failover. This failover serves as the last resort in case of an unplanned outage affecting availability of the primary site and potentially resulting in data loss.

Site Recovery supports several scenarios, such as failover and failback between two on-premises sites, failover and failback between two Azure regions, and migration from third

party provider clouds. However, in the context of this article, the focus is on replication of local disks of Azure Stack Hub VMs to Azure Storage, and on VM failover and failback between an Azure Stack Hub stack and an Azure region.

 **Note**

The Site Recovery scenario which involves replicating between on-premises VMware-based or physical datacenters reaches its end of service on December 31, 2020.

 **Note**

There's no support for Site Recovery between two deployments of Azure Stack Hub.

Details of Site Recovery architecture and its components depend on a number of criteria, including:

- The types of computers to be protected (physical versus virtual).
- For virtualized environments, the type of hypervisor hosting the virtual machines to be protected (Hyper-V versus VMware ESXi).
- For Hyper-V environments, the use of System Center Virtual Machine Manager (SCVMM) for management of Hyper-V hosts.

With Azure Stack Hub, the architecture matches the one applicable to physical computers. This isn't particularly surprising, because in both cases, Site Recovery can't benefit from direct access to a hypervisor. Instead, the mechanism that tracks and replicates changes to local disks is implemented within the protected operating system.

 **Note**

Incidentally, this is also the reason that you need to select **Physical machines** as the **Machine type** when configuring replication of Azure Stack Hub VMs in the Site Recovery interface within the Azure portal. Another implication is a unique approach to failback, which doesn't offer the same degree of automation as the one available in Hyper-V or ESXi-based scenarios.

To accomplish this, Site Recovery relies on the Site Recovery Mobility service (also referred to as *mobility agent*), which is automatically deployed to individual VMs as you enroll them into the scope of Site Recovery-based protection. On each protected VM, the locally installed instance of the mobility agent continuously monitors and forwards changes to the operating system and data disks to the process server. However, to optimize and manage the flow of replication traffic originating from protected VMs, Site Recovery implements an additional set of components running on a separate VM, referred to as the configuration server.

The configuration server coordinates communications with the Site Recovery vault and manages data replication. In addition, the configuration server hosts a component referred to as the process server, which acts as a gateway, receiving replication data, optimizing it through caching and compression, encrypting it, and finally forwarding it to Azure Storage. Effectively, the configuration server functions as the integration point between Site Recovery and protected VMs running on Azure Stack Hub. You implement that integration by deploying the configuration server and registering it with the Azure Recovery Services vault.

As part of Site Recovery configuration, you define the intended disaster recovery environment, including such infrastructure components as virtual networks, load balancers, or network security groups in the manner that mirrors the production environment. The configuration also includes a replication policy, which determines recovery capabilities and consists of the following parameters:

- RPO threshold. This setting represents the desired recovery point objective that you want to implement and determines the frequency in which Site Recovery generates crash-consistent recovery point snapshots. Its value doesn't affect the frequency of replication because that replication is continuous. Site Recovery will generate an alert, and optionally, an email notification, if the current effective RPO provided by Site Recovery exceeds the threshold that you specify. Site Recovery generates crash-consistent recovery point snapshots every five minutes.

 **Note**

A crash consistent snapshot captures data that was on the disk when the snapshot was taken. It doesn't include memory content. Effectively, a crash-consistent snapshot doesn't guarantee data consistency for the operating system or locally installed apps.

- Recovery point retention. This setting represents the duration (in hours) of the retention window for each recovery point snapshot. Protected VMs can be recovered to any recovery point within a retention window. Site Recovery supports up to 24 hours of retention for VMs replicated to Azure Storage accounts with the premium performance tier. There's a 72-hour retention limit when using Azure Storage accounts with the standard performance tier.
- App-consistent snapshot frequency. This setting determines the frequency (in hours) in which Site Recovery generates application-consistent snapshots. An app-consistent snapshot represents a point-in-time snapshot of applications running in a protected VM. There's a limit of 12 app-consistent snapshots. For VMs running Windows Server, Site Recovery uses Volume Shadow Copy Service (VSS). Site Recovery also supports app-consistent snapshots for Linux, but that requires implementing custom scripts. The scripts are used by the mobility agent when applying an app-consistent snapshot.

ⓘ Note

For details regarding implementing app-consistent snapshots on Azure Stack Hub VMs running Linux, refer to [General questions about Site Recovery](#).

For each disk of a protected Azure Stack Hub VM that you designate, data is replicated to a corresponding managed disk in Azure Storage. The disk stores the copy of the source disk and all the recovery point crash-consistent and app-consistent snapshots. As part of a failover, you choose a recovery point crash-consistent or app-consistent snapshot that should be used when attaching the managed disk to the Azure VM, which serves as a replica of the protected Azure Stack Hub VM.

During regular business operations, protected workloads run on Azure Stack Hub VMs, with changes to their disks being continuously replicated through interactions among the mobility agent, process server, and configuration server to the designated Azure Storage account. When you initiate a test, planned, or unplanned failover, Site Recovery automatically provisions Azure VMs using the replicas of disks of the corresponding Azure Stack Hub VMs.

ⓘ Note

The process of provisioning Azure VMs by using Site Recovery-replicated disks is referred to as *hydration*.

You have the option to orchestrate a failover by creating recovery plans that contain manual and automated steps. To implement the latter, you can use Azure Automation runbooks, which consist of custom PowerShell scripts, PowerShell workflows, or Python 2 scripts.

After the primary site becomes available again, Site Recovery supports reversing the direction of replication, allowing you to perform a fallback with minimized downtime and without data loss. However, with Azure Stack Hub, this approach isn't available. Instead, to fail back, it's necessary to download Azure VM disk files, upload them into Azure Stack Hub, and attach them to existing or new VMs.

## Considerations

These considerations implement the pillars of the Azure Well-Architected Framework, which is a set of guiding tenets that can be used to improve the quality of a workload. For more information, see [Microsoft Azure Well-Architected Framework](#).

## Reliability

Reliability ensures that your application can meet the commitments that you make to your customers. For more information, see [Overview of the reliability pillar](#).

Azure Stack Hub helps increase workload availability through resiliency inherent to its infrastructure. This resiliency provides high availability for Azure Stack Hub VMs protected by Site Recovery and to essential components of the on-premises Site Recovery infrastructure, including the configuration and process servers.

Similarly, you have the option to use resiliency of cloud-based components of Site Recovery infrastructure. By default, Azure Recovery Services is geo-redundant, which means that its configuration is automatically replicated to an Azure region that's part of a pre-defined region pair. You have the option to change the replication settings to locally redundant if that's sufficient for your resiliency needs. Note that you can't change this option if the vault contains any protected items. The same resiliency option is available for any Azure Storage accounts with the standard performance tier, although it's possible to change it at any point.

 **Note**

For the listing of Azure region pairs, refer to [Business continuity and disaster recovery \(BCDR\): Azure Paired Regions](#).

You can further enhance the degree of this resiliency by designing and implementing solutions that extend the scope of workload protection. This is the added value provided by Site Recovery. In the context of Site Recovery running on Azure Stack Hub, there are two main aspects of workload availability that need to be explored in more detail:

- Failover to Azure
- Failback to Azure Stack Hub

You need to consider both when developing a disaster recovery strategy driven by recovery point objectives (RPOs) and recovery time objectives (RTOs). RTO and RPO represent continuity requirements stipulated by individual business functions within an organization. RPO designates a time period representing maximum acceptable data loss following an incident that affected availability of that data. RTO designates the maximum acceptable duration of time it can take to reinstate business functions following an incident that affected the availability of these functions.

## **Failover to Azure**

Failover to Azure is at the core of availability considerations in the context of Site Recovery-based protection of Azure Stack Hub VMs. To maximize workload availability, the failover strategy should address both the need to minimize potential data loss (RPO) and minimize failover time (RTO).

To minimize potential data loss, you might consider:

- Maximizing throughput and minimizing latency of the replication traffic by following scalability and performance considerations. For more information, refer to the next section of this article.
- Increasing the frequency of app-consistent recovery points for database workloads (up to the maximum of one recovery point per hour). App-consistent recovery points are created from app-consistent snapshots. App-consistent snapshots capture app data on disk and in memory. While this approach minimizes potential data loss, it has one major drawback. App-consistent snapshots require the use of [Volume Shadow Copy Service](#) on Windows or custom scripts on Linux, to quiesce locally installed apps. The capture process can hurt performance, especially if resource utilization is high. We don't recommend that you use low frequency for app-consistent snapshots for non-database workloads.

The primary method of minimizing failover time involves the use of Site Recovery recovery plans. A recovery plan orchestrates a failover between the primary and secondary sites, defining the sequence in which protected servers fail over. You can customize a plan by adding manual instructions and automated tasks. Its purpose is to make the process consistent, accurate, repeatable, and automated.

When creating a recovery plan, you assign protected servers to recovery groups for the purpose of failover. Servers in each group fail over together. This helps you to divide the failover process into smaller, easier to manage units, representing sets of servers which can fail over without relying on external dependencies.

To minimize failover time, as part of creating a recovery plan, you should:

- Define groups of Azure Stack Hub VMs that should fail over together.
- Define dependencies between groups of Azure Stack Hub VMs to determine the optimal sequence of a failover.
- Automate failover tasks, if possible.
- Include custom manual actions, if required.

 **Note**

A single recovery plan can contain up to 100 protected servers.

 **Note**

In general, recovery plans can be used for both failover to and failback from Azure. This doesn't apply to Azure Stack Hub, which doesn't support Site Recovery-based failback.

You define a recovery plan and create recovery groups to capture app-specific properties. As an example, let's consider a traditional three-tier app with a SQL Server-based back end, a middleware component, and a web front end. When creating a recovery plan, you can control the startup order of servers in each tier, with the servers running SQL Server instances coming online first, followed by those in the middleware tier, and joined afterwards by servers hosting the web front end. This sequence ensures that the app is working by the time the last server starts. To implement it, you can simply create a recovery plan with three recovery groups, containing servers in the respective tiers.

In addition to controlling failover and startup order, you also have the option to add actions to a recovery plan. In general, there are two types of actions:

- Automated. This action is based on Azure Automation runbooks, which involves one of two types of tasks:
  - Provisioning and configuring Azure resources, including, for example, creating a public IP address and associating it with the network interface attached to an Azure VM.
  - Modifying the configuration of the operating system and applications running within an Azure VM that was provisioned following a failover.
- Manual. This action doesn't support automation and is included in a recovery plan primarily for documentation purposes.

 **Note**

To determine the failover time of a recovery plan, perform a test failover and then examine the details of the corresponding Site Recovery job.

 **Note**

To address the RTO requirements for Azure Stack Hub workloads, you should account for recovery of the Azure Stack infrastructure, user VMs, applications, and user data. In the context of this article, we are interested only in the last two of these components, although we also present considerations regarding the availability of the Modern Backup Storage functionality.

## Fallback to Azure Stack Hub

In Site Recovery-based scenarios, fallback, if properly implemented, doesn't involve data loss. This means that the focus of the failover strategy is to minimize failback time (RTO). However, as previously mentioned, when failing back to Azure Stack Hub, you can't rely on your recovery plans. Instead, the fallback involves the following sequence of steps:

1. Stop and deallocate Azure VMs in the disaster recovery environment.
2. Identify the URI parameter of each of the managed disks attached to the VMs you intend to download.
3. Download the virtual hard disk (VHD) files identified by the URI parameters you identified in the previous step to your on-premises environment.
4. Upload the VHD files to Azure Stack Hub.
5. Attach the uploaded VHDs to new or existing Azure Stack Hub VMs.
6. Start the Azure Stack Hub VMs.

The optimal approach to minimizing the failback time is to automate it.

 **Note**

For more information regarding automating the failback procedure described in this section, refer to [Create VM disk storage in Azure Stack Hub](#).

 **Note**

For more information regarding identifying the URI parameter of managed disks, refer to [Download a Windows VHD from Azure](#).

## Workload-specific considerations

Site Recovery integrates with Windows Server-based apps and roles, including SharePoint, Exchange, SQL Server, and Active Directory Domain Services (AD DS). This allows you to use the following capabilities to implement app-level protection and recovery:

- Integration with app-level replication technologies, such as SQL Server AlwaysOn Availability Groups, Exchange Database Availability Groups (DAGs), and AD DS replication
- App-consistent snapshots, for single or multiple tier applications
- A rich automation library that provides production-ready, application-specific scripts that can be downloaded and integrated with recovery plans

Alternatively, you have the option to use workload-specific replication mechanisms to provide site-level resiliency. This is a commonly used option when implementing disaster recovery for AD DS domain controllers, SQL Server, or Exchange, all of which natively support replication. Though this requires provisioning Azure VMs hosting these workloads in the disaster recovery environment, which increases the cost, it offers the following benefits:

- Reduces time required for failover and failback

- Simplifies workload-level failover, accommodating scenarios in which site-level failover isn't required

(!) **Note**

For more information regarding Site Recovery workload-specific considerations, refer to [About disaster recovery for on-premises apps](#).

## Security

Managing disaster recovery of user VM-based workloads in hybrid scenarios warrants additional security considerations. These considerations can be grouped into the following categories:

- Encryption in transit. This includes communication between protected Azure Stack Hub VMs, on-premises Site Recovery components, and cloud-based Site Recovery components:
  - Mobility agent installed on the protected VMs always communicates with the process server via Transport Layer Security (TLS) 1.2.
  - It's possible for communication from the configuration server to Azure and from the process server to Azure to use TLS 1.1 or 1.0. To increase the level of security for hybrid connectivity, you should consider enforcing the use of TLS 1.2.

(!) **Note**

For details regarding configuring TLS 1.2-based encryption, refer to [Transport Layer Security \(TLS\) registry settings](#) and [Update to enable TLS 1.1 and TLS 1.2 as default secure protocols in WinHTTP in Windows](#)

- Encryption at rest. This includes Azure Storage and Azure VMs in the disaster recovery site.
  - Azure Storage is encrypted at rest for all storage accounts using 256-bit Advanced Encryption Standard encryption and is Federal Information Processing Standard 140-2 compliant. Encryption is enabled automatically and can't be disabled. By default, encryption uses Microsoft-managed keys, but customers have the option to use their own keys stored in an Azure Key vault.
  - Managed disks of Azure VMs are automatically encrypted by using *server-side encryption of Azure managed disks*, which also applies to their snapshots, by relying on using platform-managed encryption keys.

In addition, you can enforce restricted access to the Azure Storage accounts hosting content of Site Recovery-replicated disks. To do this, enable the managed identity for the Recovery

Services vault and assign to that managed identity the following Azure role-based access control (Azure RBAC) roles at the Azure Storage account level:

- Resource Manager-based storage accounts (standard performance tier):
  - Contributor
  - Storage Blob Data Contributor
- Resource Manager-based storage accounts (premium performance tier):
  - Contributor
  - Storage Blob Data Owner

The Azure Recovery Services vault offers mechanisms that further protect its content, including the following protections:

- Azure RBAC. This allows for delegation and segregation of responsibilities according to the principle of least privilege. There are three Site Recovery-related built-in roles that restrict access to Site Recovery operations:
  - Site Recovery Contributor. This role has all the permissions required to manage Site Recovery operations in an Azure Recovery Services vault. A user with this role, however, can't create or delete the vault or assign access rights to other users. This role is best suited for disaster recovery administrators who can enable and manage disaster recovery for an Azure Stack Hub tenant.
  - Site Recovery Operator. This role has permissions to execute and manage failover and fallback operations. A user with this role can't enable or disable replication, create or delete vaults, register new infrastructure, or assign access rights to other users. This role is best suited for a disaster recovery operator who can fail over Azure Stack Hub VMs when instructed by application owners and IT administrators in an actual or simulated disaster scenario.
  - Site Recovery Reader. This role has permissions to track all Site Recovery management operations. This role is best suited for IT staff responsible for monitoring the status of protected Azure Stack Hub VMs and raising support tickets if required.
- Azure Resource Locks. You have the option to create and assign read-only and delete locks to a Site Recovery vault to mitigate the risk of the vault being accidentally and maliciously changed or deleted.
- Soft delete. The purpose of soft delete is to help protect the vault and its data from accidental or malicious deletions. With soft delete, any deleted content is retained for 14 additional days, allowing for its retrieval during that period. The additional 14-day retention of vault content doesn't incur any cost. Soft delete is enabled by default.
- Protection of security-sensitive operations. Azure Recovery Services vault allows you to enable an additional layer of authentication whenever a security-sensitive operation, such as disabling protection, is attempted. This extra validation helps ensure that authorized users perform such operations.

- Monitoring and alerts of suspicious activity. Azure Recovery Services provides built-in monitoring and alerting of security-sensitive events related to the vault operations.

## Cost optimization

Cost optimization is about looking at ways to reduce unnecessary expenses and improve operational efficiencies. For more information, see [Overview of the cost optimization pillar](#).

When considering the cost of the Site Recovery-based disaster recovery solution described in this article, you need to account for both on-premises and cloud-based components. The Azure Stack Hub pricing model determines the pricing of on-premises components. As with Azure, Azure Stack Hub offers a pay-as-you-use arrangement, available through enterprise agreements and the Cloud Solution Provider program. This arrangement includes a monthly price for each Windows Server VM. If you have the option to use existing Windows Server licenses, you can significantly reduce the cost to the base VM pricing. However, with Site Recovery, you will usually need only a single Azure Stack Hub VM per tenant, which is required to implement the tenant-specific configuration server.

Azure-related charges are associated with the use of the following resources:

- Azure Recovery Services. The pricing is determined by the number of protected instances. It's worth noting that every protected instance incurs no Site Recovery charges for the first 31 days.
- Azure Storage. The pricing reflects a combination of the following factors:
  - Performance tier
  - Volume of data stored
  - Volume of outbound data transfer
  - Quantity and types of operations performed (for standard performance tier only)
  - Data redundancy (for standard performance tier only)
- Azure ExpressRoute. The pricing is based on one of two models:
  - Unlimited data. This model includes a monthly fee with all inbound and outbound data transfers included.
  - Metered data. This model includes a monthly fee with all inbound data transfers free of charge and outbound data transfers charged per GB.

 **Note**

This assessment doesn't include the costs of physical connections delivered by third party connectivity providers.

- Azure VMs. The pricing of Azure VMs reflects a combination of two components:

- Compute cost. The VM size, its uptime, and the licensing model of its operating system determine the cost.
- Managed disk cost. The disk size and performance tier determine the cost.

 **Note**

It's worth noting that hydration eliminates the need to run Azure VMs during regular business operations, with workloads running on Azure Stack Hub, which considerably reduces the compute costs of Site Recovery-based implementations, especially in comparison to traditional disaster recovery solutions.

 **Note**

The prices of resources vary between Azure regions.

 **Note**

For details regarding pricing, refer to [Azure Pricing](#).

## Operational excellence

Operational excellence covers the operations processes that deploy an application and keep it running in production. For more information, see [Overview of the operational excellence pillar](#).

The primary considerations regarding manageability of Site Recovery-based disaster recovery of Azure Stack Hub VMs include:

- Implementation of Site Recovery on Azure Stack Hub
- Failover and failback procedures
- Delegation of roles and responsibilities
- DevOps

## Implementation of Site Recovery on Azure Stack Hub

To implement Site Recovery on Azure Stack Hub in a small to medium sized single-tenant environment, you can follow the manual provisioning process driven by the graphical interface of Recovery Services Vault in the Azure portal. For multi-tenant implementations, you might want to consider automating parts of the implementation process, because you will typically need to set up a separate configuration server VM and a separate Recovery Services vault for each tenant. You also have the option to automate deployment of the

mobility agent by following the procedure described in [Prepare source machine for push installation of mobility agent](#).

## Failover and failback procedures

To simplify the management of failover, consider implementing recovery plans for all protected workloads. For more information, refer to the [Reliability](#) section earlier in this article. You will also find recommendations for optimizing the management of the failback procedure.

## Delegation of roles and responsibilities

Planning for and implementing disaster recovery of Azure Stack Hub VM-based workloads by using Site Recovery typically involves interaction of stakeholders:

- Azure Stack Hub operators manage Azure Stack Hub infrastructure, ensuring that there are sufficient compute, storage, and network resources necessary for implementing a comprehensive disaster recovery solution and making these resources available to tenants. They also collaborate with application and data owners to help determine the optimal approach to deploying their workloads to Azure Stack Hub.
- Azure administrators manage Azure resources necessary to implement hybrid disaster recovery solutions.
- Microsoft Entra administrators manage Microsoft Entra resources, including user and group objects that are used to provision, configure, and manage Azure resources.
- Azure Stack Hub tenant IT staff designs, implements, and manages Site Recovery, including failover and failback.
- Azure Stack Hub users need to provide RPO and RTO requirements and submit requests to implement disaster recovery for their workloads.

Make sure there's a clear understanding of the roles and responsibilities attributed to application owners and operators in the context of protection and recovery. Users are responsible for protecting VMs. Operators are responsible for the operational status of the Azure Stack Hub infrastructure.

### Note

For guidance regarding fine-grained delegation of permissions in Site Recovery scenarios, refer to [Manage Site Recovery access with Azure role-based access control \(Azure RBAC\)](#).

While configuring VM-level recovery by using Site Recovery is primarily a responsibility of IT operations, there are some DevOps-specific considerations that should be incorporated into a comprehensive disaster recovery strategy. Azure Stack Hub facilitates implementing Infrastructure-as-Code (IaC), which incorporates the automated deployment of a variety of workloads, including VM-based applications and services. You can use this capability to streamline the provisioning of Site Recovery-based disaster recovery scenarios, which simplifies the initial setup in multiple tenant scenarios.

For example, you can use the same Azure Resource Manager templates to provision all of the network resources necessary to accommodate VM-based workloads in an Azure Stack Hub stamp for your application in a single, coordinated operation. You can use the same template to provision a matching set of resources in Azure to provision a disaster recovery site. To account for any differences between the two environments, you can simply specify different values of template parameters in each case.

## Performance efficiency

Performance efficiency is the ability of your workload to scale to meet the demands placed on it by users in an efficient manner. For more information, see [Performance efficiency pillar overview](#).

When planning to deploy Site Recovery on Azure Stack Hub, you need to consider the amount of processing, storage, and network resources allocated to the configuration and process servers. You might need to adjust the estimated sizing of the Azure Stack Hub VM hosting the Site Recovery components post deployment to accommodate changes in processing or storage requirements. You have three basic options to adjust the sizing:

- Implement vertical scaling. This involves modifying the amount and type of processor, memory, and disk resources of the Azure Stack Hub VM hosting the configuration server including the process server. To estimate resource requirements, you can use the information in the following table:

*Table 1: Configuration and process server sizing requirements*

 [Expand table](#)

CPU	Memory	Cache disk	Data change rate	Protected machines
8 vCPUs 2 sockets * 4 cores @ 2.5 GHz	16GB	300 GB	500 GB or less	< 100 machines

CPU	Memory	Cache disk	Data change rate	Protected machines
12 vCPUs 2 sockets * 6 cores @ 2.5 GHz	18 GB	600 GB	500 GB-1 TB	100 to 150 machines
16 vCPUs 2 sockets * 8 cores @ 2.5 GHz	32 GB	1 TB	1-2 TB	150-200 machines

- Implement horizontal scaling. This involves provisioning or deprovisioning Azure Stack Hub VMs with the process server installed to match processing demands of protected Azure Stack Hub VMs. In general, if you have to scale your deployment to more than 200 source machines, or you have a total daily churn rate of more than two terabytes (TB), you need additional process servers to handle replication traffic. To estimate the number and configuration of additional process servers, refer to [Size recommendations for the process server](#).
- Modify replication policy. This involves changing parameters of the replication policy, with focus on app-consistent snapshots.

From the networking standpoint, there are several different methods to adjust bandwidth available for replication traffic:

- Modify VM size. The size of Azure Stack Hub VMs determines the maximum network bandwidth. However, it's important to note that there are no bandwidth guarantees. Instead, VMs can utilize the amount of available bandwidth up to the limit determined by their size.
- Replace uplink switches. Azure Stack Hub systems support a range of hardware switches, offering several choices of uplink speeds. Each Azure Stack Hub cluster node has two uplinks to the top of rack switches for fault tolerance. The system allocates half of the uplink capacity for critical infrastructure. The remainder is shared capacity for Azure Stack Hub services and all user traffic. Systems deployed with faster speeds have more bandwidth available for replication traffic.

 **Note**

While it's possible to segregate network traffic by attaching a second network adapter to a server, with Azure Stack Hub VMs, all VM traffic to the internet shares the same uplink. A second, virtual network adapter won't segregate traffic at the physical transport level.

- Modify throughput of the network connection to Azure. To accommodate larger volumes of replication traffic, you might consider using Azure ExpressRoute with Microsoft peering for connections between Azure Stack Hub virtual networks and Azure Storage. Azure ExpressRoute extends on-premises networks into the Microsoft cloud over a private connection supplied by a connectivity provider. You can buy ExpressRoute circuits for a wide range of bandwidths, from 50 megabits per second (Mbps) to 100 gigabits per second.

 **Note**

For details regarding implementing Azure ExpressRoute in Azure Stack Hub scenarios, refer to [Connect Azure Stack Hub to Azure using Azure ExpressRoute](#).

- Modify throttling of replication traffic on the process server. You can control how much bandwidth is used by the replication traffic on the VMs that are hosting process servers from the graphical interface of the Microsoft Azure Recovery Services agent. The supported capabilities include setting the limits for work and non-work hours, with the bandwidth values ranging from 512 kilobits per second to 1,023 Mbps. Alternatively, you can apply the same configuration by using the **Set-OBMachineSetting** PowerShell cmdlet.
- Modify network bandwidth allocated per protected VM on the process server. To accomplish this, modify the value of **UploadThreadsPerVM** entry within the **HKEY\_LOCAL\_MACHINE\SOFTWARE\Microsoft\Windows Azure Backup\Replication** key. By default, the value is set to 4, but you can increase it to 32 if there's enough network bandwidth available.

## Deploy this scenario

### Prerequisites

Implementing the recommended solution is contingent on satisfying the following prerequisites:

- Access to an Azure subscription, with permissions sufficient to provision and manage all cloud components of the Site Recovery components, including:
  - Azure Recovery Services vault in the Azure region designated as the disaster recovery site for the Azure Stack Hub production environment.
  - An Azure Storage account hosting content of replicated disks of Azure Stack Hub VMs.
  - An Azure virtual network representing the disaster recovery environment to which hydrated Azure VMs will be connected following a planned or unplanned failover.

- An isolated Azure virtual network representing the test environment to which hydrated Azure VMs will be connected following a test failover.
- An Azure ExpressRoute-based connectivity between the on-premises environment, Azure virtual networks, and the Azure storage account used for hosting copies of VHD files with content replicated from disks of protected Azure Stack Hub VMs.
- An Azure Stack Hub user subscription. All Azure Stack Hub VMs protected by an individual Site Recovery configuration server must belong to the same Azure Stack Hub user subscription.
- An Azure Stack Hub virtual network. All protected VMs must have direct connectivity to the VMs hosting the process server component (by default this is the configuration server VM).
- An Azure Stack Hub Windows Server VM that will host the configuration server and a process server. The VM must belong to the same subscription and be attached to the same virtual network as the Azure Stack Hub VMs that need to be protected. In addition, the VM needs to:
  - Comply with [Site Recovery configuration server software and hardware requirements](#)
  - Satisfy external connectivity [network requirements](#) documentation.

 **Note**

Additional storage and performance considerations for the configuration and process servers are described in more detail later in this architecture.

- Satisfy internal connectivity requirements. In particular, Azure Stack Hub VMs that you want to protect need to be able to communicate with:
  - The configuration server via TCP port **443** (HTTPS) inbound for replication management
  - The process server via TCP port **9443** to deliver replication data.

 **Note**

You can change the port used by the process server for both external and internal connectivity as part of its configuration when running Site Recovery Unified Setup.

- Azure Stack Hub VMs to be protected, running any of the [supported operating systems](#). To protect Azure Stack Hub VMs that are running Windows Server operating systems, you must:

- Create a Windows account with administrative rights. You can specify this account when you enable Site Recovery on these VMs. The process server uses this account to install the Site Recovery Mobility service. In a workgroup environment, make sure to disable Remote User Access control on target Windows Server operating systems by setting the value of the **LocalAccountTokenFilterPolicy** **DWORD** registry entry in the **HKEY\_LOCAL\_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System** key to 1.
- Enable File and Printer Sharing and Windows Management Instrumentation rules in Microsoft Defender firewall.
- To protect Azure Stack Hub VMs that are running Linux operating systems, you must:
  - Create a root user account. You can specify this account when you enable Site Recovery on these VMs. The process server uses this account to install the Site Recovery Mobility service.
  - Install the latest openssh, openssh-server, and openssl packages.
  - Enable and run Secure Shell (SSH) port 22.
  - Enable Secure FTP subsystem and password authentication.

## High-level implementation steps

At a high level, the implementation of Site Recovery-based disaster recovery on Azure Stack Hub consists of the following stages:

1. Prepare Azure Stack Hub VMs to be protected by Site Recovery. Ensure that the VMs satisfy Site Recovery prerequisites listed in the previous section.
2. Create and configure an Azure Recovery Services vault. Set up an Azure Recovery Services vault and specify what you want to replicate. Site Recovery components and activities are configured and managed by using the vault.
3. Set up the source replication environment. Provision a Site Recovery configuration server and process server by installing Site Recovery Unified Setup binaries and register it with the vault.

 **Note**

You can rerun the Site Recovery Unified Setup to implement additional process servers on Azure Stack Hub VMs.

4. Set up the target replication environment. Create or select an existing Azure storage account and an Azure virtual network in the Azure region that will host the disaster recovery site. During replication, the content of the disks for the protected Azure Stack

Hub VMs is copied to the Azure Storage account. During failover, Site Recovery automatically provisions Azure VMs serving as replicas of protected Azure Stack Hub VMs and connects them to the Azure virtual network.

5. Enable replication. Configure replication setting and enable replication for Azure Stack Hub VMs. The mobility service is installed automatically on each Azure Stack Hub VM for which replication is enabled. Site Recovery initiates replication of each Azure Stack Hub VM, according to the policy settings you defined.
6. Perform a test failover. After replication is established, verify that failover will work as expected by performing a test failover.
7. Perform a planned or unplanned failover. Following a successful test failover, you are ready to conduct either a planned or unplanned failover to Azure. You have the option to designate which Azure Stack Hub VMs to include in the failover.
8. Perform a failback. When you are ready to fail back, stop the Azure VMs corresponding to the Azure Stack Hub VMs you failed, download their disk files to on-premises storage, upload them into Azure Stack Hub, and attach them to an existing or new VM.

## Summary

In conclusion, Azure Stack Hub is a unique offering, which differs in many aspects from other virtualization platforms. As such, it warrants special considerations in regard to business continuity strategy for its workloads. By using Azure services, you can simplify designing and implementing this strategy. In this architecture reference article, we explored the use of Microsoft Site Recovery for protecting Azure Stack Hub VM-based workloads in the connected deployment model. This approach allows customers to benefit from resiliency and manageability of Azure Stack Hub and from the hyperscale and global presence of the Azure cloud.

It's important to note that the disaster recovery solution described here focused exclusively on VM-based workloads of Azure Stack Hub. This is only part of an overall business continuity strategy that should account for other Azure Stack Hub workload types and scenarios that affect their availability.

## Next steps

Product documentation:

- [About Site Recovery](#)
- [Azure Stack Hub overview](#)
- [What is Microsoft Entra ID?](#)
- [What is Azure Blob storage?](#)

- [What is Azure ExpressRoute?](#)
- [What is Azure Virtual Network?](#)

Microsoft Learn modules:

- [Azure Stack Hub](#)
- [Configure Microsoft Entra ID](#)
- [Configure storage accounts](#)
- [Configure virtual networks](#)
- [Design and implement Azure ExpressRoute](#)
- [Design your site recovery solution in Azure](#)

## Related resources

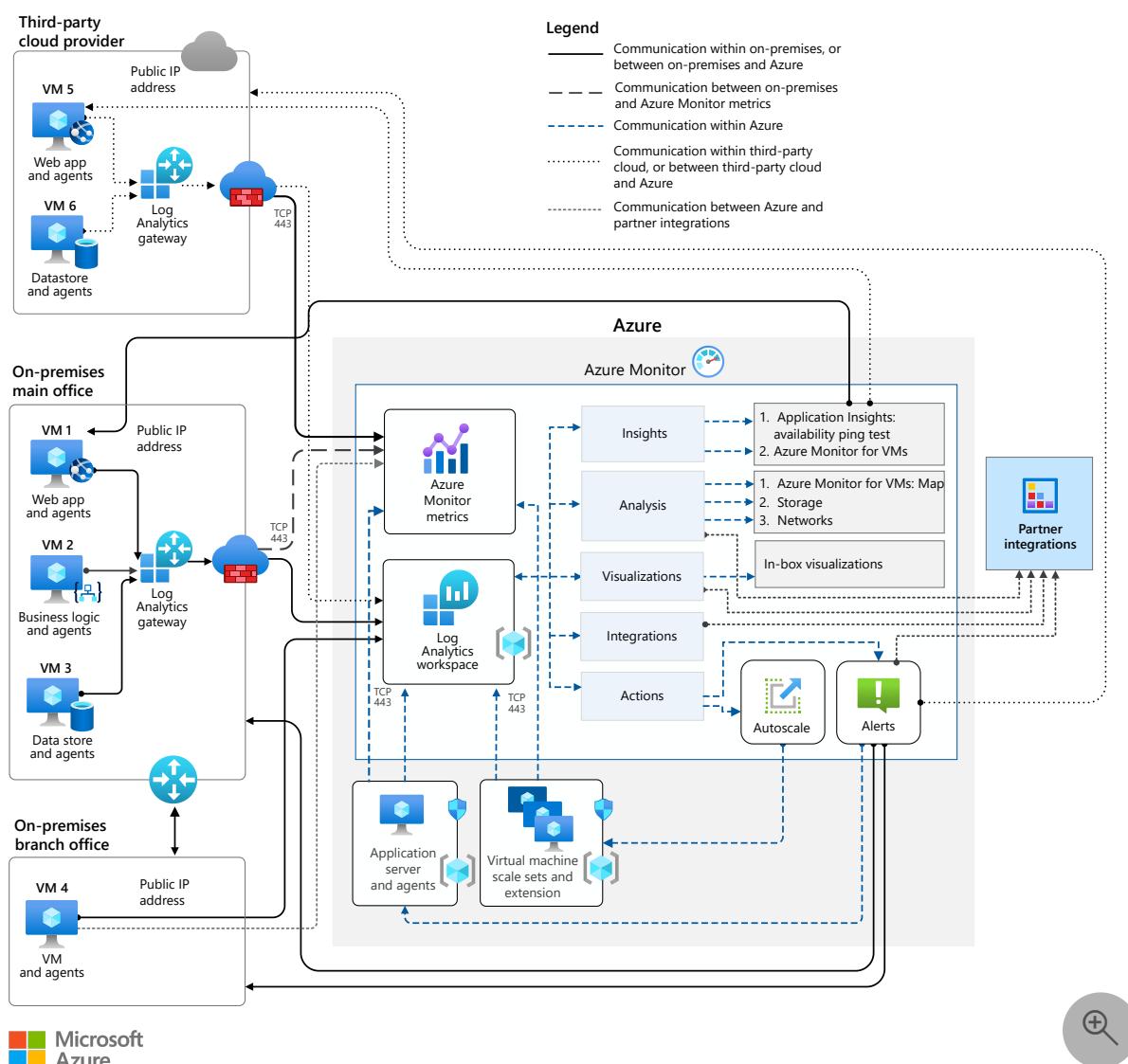
- [Hybrid architecture design](#)
- [Back up files and applications on Azure Stack Hub](#)
- [Hybrid connections](#)
- [Hybrid file share with disaster recovery for remote and local branch workers](#)

# Hybrid availability and performance monitoring

Azure Event Hubs   Azure Log Analytics   Azure Monitor   Azure Storage   Azure Virtual Machines

This reference architecture shows how to use Azure Monitor to monitor the performance and availability of operating system (OS) workloads that run in virtual machines (VMs). The VMs can be in Microsoft Azure, in on-premises environments, or in non-Azure clouds.

## Architecture



Download a [Visio file](#) of this architecture.

# Workflow

- **On-premises main office - VM 1.** This component is a web application with internet access and a public-facing webpage, and both Log Analytics and Dependency agents installed. For information about agents, refer to [Log Analytics agent overview](#) and [Overview of Azure Monitor agents, Dependency agent](#).
- **On-premises main office - VM 2.** This business-logic environment doesn't have internet access. It does, however, have Log Analytics and Dependency agents installed.
- **On-premises main office - VM 3.** This component is a datastore without internet access, but with Log Analytics and Dependency agents installed.
- **On-premises main office - Log Analytics gateway.** The Log Analytics gateway collects log and metric data from the three on-premises VMs, and delivers them into the *Log Analytics workspace* over Transmission Control Protocol (TCP) on port 443.
- **On-premises main office - Firewall.** Traffic to and from the on-premises environment is routed through the firewall.
- **Gateway.** The gateway provides connectivity to the branch office.
- **On-premises branch office - VM 4.** This component is the business application that's running without internet access, but with Log Analytics and Dependency agents installed. The Log Analytics agent installed on the VM is configured to transfer data directly to the Log Analytics workspace without the need for a Log Analytics gateway.
- **On-premises branch office - Gateway.** This gateway connects the branch office to the on-premises main office via a Virtual Private Network (VPN).
- **Third-party cloud provider - VM 5.** This component is a web application with internet access, a public-facing webpage, and both Log Analytics and Dependency agents installed.
- **Third-party cloud provider - VM 6.** This component is a datastore environment without internet access, but with both Log Analytics and Dependency agents installed. There is no direct connectivity from the third-party cloud provider environments to the on-premises environments.
- **Azure - VMSS.** This is a scale set that's created by using Azure Virtual Machine Scale Sets. It runs a business application with the log analytics and diagnostic agents installed.
- **Azure - Application server.** This server has a single VM running a business application, with Log Analytics and diagnostic agents installed.
- **Azure Monitor metrics.** Data collected by Azure Monitor metrics is stored in a time series database that's optimized for analyzing timestamped data. It also stores metrics sent from on-premises VMs and Azure VMs.

- **Azure Monitor - Log Analytics workspace.** This workspace stores logs sent from on-premises VMs, Azure VMs, and VMs on third-party cloud providers. The workspace is an Azure resource where data is aggregated, and it serves as an administrative boundary for accessing this data. Other Azure Monitor services then connect to the Log Analytics workspace and use the data for various purposes. For more information, see [Designing your Azure Monitor Logs deployment](#).
- **Azure Monitor - Insights - Application Insights.** Application Insights provides analyses of applications and insights into their use. In this example architecture, an availability ping test checks the availability of the on-premises web application. Alert rules are enabled to provide notification of a failed test. For more information, see [What is Application Insights?](#) and [Monitor the availability of any website](#).
- **Azure Monitor - Insights - Azure Monitor for VMs.** Azure Monitor for VMs monitors the performance and health of your virtual machines and virtual machine scale sets. The monitoring includes their running processes and dependencies on other resources. In this scenario, the Azure Monitor for VMs will provide insights into your virtual machines. For more information, see [What is Azure Monitor for VMs?](#).
- **Azure Monitor - Analysis.** Log and metric data from the VMs is queried within Azure Monitor metrics and the Log Analytics workspace using the Kusto Query Language (KQL). The results provide insights into the infrastructure, topology, and resources. For more information, see [Kusto: Overview](#) and [Azure Monitor log query examples](#).
- **Azure Monitor - Visualizations.** Azure Monitor uses visualization tools to review application and infrastructure components and communications between services in Azure Monitor. Visualization tools include [Application Map in Azure Application Insight](#), the [Map feature of Azure Monitor for VMs](#), [Azure Monitor Workbooks](#), and various dashboard views available within Azure Monitor. For more information, see [Use the Map feature of Azure Monitor for VMs to understand application components](#), [Create and share dashboards of Log Analytics data](#), and [Azure Monitor Workbooks](#).
- **Azure Monitor - Integrations.** Azure Monitor integrates with a range of partner and third-party tools and extensions. These tools and extensions enhance and build upon existing Azure Monitor functionality, such as analysis and visualizations.
- **Azure Monitor - Actions - Alerts.** Variations in metric and log data can indicate the occurrence of events. Rules define the data variations that trigger alerts, provide notifications, and initiate remediation responses. In this architecture, when an alert is triggered, automation runbooks automatically remediate the on-premises VMs and Azure VMs. Webhook actions, Service Management integration, and other action types are also available. For more information, see [Create, view, and manage](#)

metric alerts using Azure Monitor and Create, view, and manage log alerts using Azure Monitor.

- **Azure Monitor - Actions - Autoscale.** Autoscale adds or removes VM instances according to demand, which maintains performance and increases cost effectiveness. In this architecture, Autoscale has conditions defined around average CPU load (in percentage). When conditions are met, Azure Monitor Autoscale will adjust the scale set according to demand. For more information, see [Overview of autoscale in Microsoft Azure](#).

## Components

The architecture consists of the following components:

- [Azure Virtual Machines](#)
- [Azure Monitor](#)
- [Azure Policy](#)
- [Azure Event Hubs](#)
- [Azure Storage](#)

## Recommendations

The following best practices are recommendations that apply for most scenarios. Follow these practices unless you have a specific requirement that overrides them.

## Log Analytics workspace

Consider the following recommendations when designing the Log Analytics workspace:

- Place the workspace and resources in the same Azure region, if latency is an important factor.
- Start with a single Log Analytics workspace, and increase the number of workspaces as the requirements change.
- If you have geographically dispersed teams and resources, you might need one workspace per region.
- Your workspace doesn't need to be in the same subscription as the resources you're running.

## Alerts

For simpler scenarios, you can use metrics to flag alerts rather than logs. Metrics:

- Provide a count, or *numerical value*, for events such as CPU usage, available memory, or logical disk space.
- Have low latency.
- Offer greater granularity, for example per-second or per-minute intervals.
- Notify you about an issue quickly.

To collect custom performance indicators or business-specific metrics to provide deeper insights, use custom metrics. For more information, see [Custom metrics in Azure Monitor \(Preview\)](#).

Metrics alerts are not the answer in all situations. You might still want to use log-based alerts when you require more customization or more powerful correlations.

## Analysis and Diagnostics

Consider the following recommendations for analysis and diagnostics:

- Use logs for deeper analysis. Logs can:
  - Provide verbose detail about events (compared to metrics).
  - Happen intermittently.
  - Facilitate deeper diagnostics after an initial metric flag.
- Customize log data collection (which is similar to metrics) using the HTTP Data Collector API to send log data to a Log Analytics workspace. For more information, see [Send log data to Azure Monitor with the HTTP Data Collector API \(public preview\)](#).
- Analyze your applications proactively with the **smart detection** feature of Application Insight. Smart detection applies the machine learning capabilities of Azure and statistical analysis to detect issues such as performance or failure anomalies, memory leaks, or general application degradation. For more information, see [Smart Detection in Application Insights](#).
- Use **Azure Monitor for VMs - Map** to review connections between servers, processes, inbound and outbound connection latency, and ports across any TCP-connected architecture. No configuration is required other than installing an agent. With **Azure Monitor for VMs - Map**, you can interact and engage with your servers as interconnected systems.

## Log Analytics queries

Query the data within a **Log Analytics workspace** by using KQL to search for terms, specific events, or states to identify trends and analyze patterns. Use the **Query explorer**

to browse and select pre-written queries, modify them, or create your own. You can run, save, share, and export queries from within a workspace, and pin your favorite queries to a dashboard for reuse.

## Agent installation

Install agents automatically and at scale, rather than individually, by using automation options such as Azure Policy, Azure PowerShell, Resource Manager templates, or Desired State Configuration (DSC). For more information, see [Enable Azure Monitor for VMs by using Azure Policy](#), [Enable Azure Monitor for VMs using Azure PowerShell](#), and [Enable Azure Monitor for VMs for a hybrid virtual machine - Desired State Configuration](#).

## Dashboard

For critical applications, create an **Azure Dashboard** view. Share or make your dashboard available on a shared screen, in real time, to people who need critical application data.

## Considerations

These considerations implement the pillars of the Azure Well-Architected Framework, which is a set of guiding tenets that can be used to improve the quality of a workload. For more information, see [Microsoft Azure Well-Architected Framework](#).

## Reliability

Reliability ensures that your application can meet the commitments that you make to your customers. For more information, see [Overview of the reliability pillar](#).

The following considerations help to ensure availability in your environment.

- Availability tests. The URL ping test used in this architecture is the simplest *outside-in* availability test. However, other options are available, such as:
  - Multi-step web test. Plays back recordings of sequenced web requests to test complex scenarios. Multiple-step web tests are created in Microsoft Visual Studio Enterprise, and then uploaded to the portal for execution.
  - Custom track availability tests. Use the `TrackAvailability()` method to send test results to Application Insights.
- Alerts. When you create an availability test in Application Insights, event alert notifications are enabled by default. You can edit the alert rules by specifying the notification type and details, from **Azure Monitor > Alerts**.

# Security

Security provides assurances against deliberate attacks and the abuse of your valuable data and systems. For more information, see [Overview of the security pillar](#).

The following items are considerations for making your environment more secure.

- Log Analytics workspace. Access modes are defined as one of the following contexts:
  - Workspace context. All logs that the workspace has permission to access can be queried. This is a vertical access approach. For example, a security team might need access to all resource data from the top down.
  - Resource context. Only logs for specific resources can be queried. For example, an application team can be granted access to logs for the particular resource they're working on.
- Secure data in transit to Log Analytics. Data in transit is secured using minimum Transport Layer Security (TLS) 1.2. You don't need to enable this feature explicitly. For more information, see [Log Analytics data security](#).
- Secure data at rest in Log Analytics. Data at rest in Log Analytics is secured, as per Azure Storage, using 256-bit Advanced Encryption Standard (AES) encryption by default.
- Smart Detection. Use Smart Detection in Application Insights to analyze the telemetry generated by your application, and to detect security issues. For more information, see [Application security detection pack \(preview\)](#).
- Integrate Azure Monitor with Security Information and Event Management (SIEM) tools. Route your monitoring data to an event hub with Azure Monitor to integrate external SIEM and monitoring tools. For more information, see [Stream Azure monitoring data to an event hub or external partner](#).

# Cost optimization

Cost optimization is about looking at ways to reduce unnecessary expenses and improve operational efficiencies. For more information, see [Overview of the cost optimization pillar](#).

The following items are considerations for controlling and managing costs in your environment.

- Azure Monitor. Azure Monitor costs are consumption-based, often referred to as *pay as you go*.
- Log Analytics. You pay for **data ingestion** and **data retention**. You can estimate and forecast the number of VMs, and the amount of data (in gigabytes) you expect

to collect from each VM. A typical Azure VM consumes between 1 gigabyte (GB) and 3 GB of data each month. If you're evaluating data usage with Azure Monitor logs, use the data statistics from your own environment and obtain a discount with **Capacity reservations**.

- Application Insights. This component is billed according to the volume of telemetry data your application sends, and the number of web tests you run.
- Metric queries. Metric queries are billed by the number of calls made.
- Alerts. Alerts are billed based on the type, and number, of signals monitored.
- Notifications. Notifications are billed according to the type, and number, of notifications you send.
- Azure Monitor. The **Usage and estimated costs** section of Azure Monitor estimates your monthly costs based on the previous 31 days of usage.
- For more information, see [Azure Monitor pricing](#) and [Pricing calculator](#).

## Operational excellence

Operational excellence covers the operations processes that deploy an application and keep it running in production. For more information, see [Overview of the operational excellence pillar](#).

## Manageability

The following are considerations for making your environment more manageable.

- Azure Workbooks. Use workbooks to help perform further analysis, and create rich reports. Workbooks combine text, log queries, metrics, and parameters into interactive reports. Team members with access to the same Azure resources can edit workbooks. For more information, see [Create interactive reports Azure Monitor for VMs with workbooks](#).
- Partner integrations. Integrate Azure Monitor with partner and third-party tools to assist with analysis, visualization, alerts, or Service Management and Azure Pipelines. For more information, see [Azure Monitor partner integrations](#).
- Integrate Azure Monitor with Microsoft System Center. Integrate Azure Monitor with the System Center product suite. For more information, see [Connect Operations Manager to Azure Monitor](#).
- Send data to Azure Event Hubs. For integrating Azure Monitor with visualization and external monitoring tools, refer to [Stream Azure monitoring data to an event hub or external partner](#).
- Log Analytics gateway. For smaller environments such as the branch office, use the agent to transfer data into the Log Analytics workspace, rather than into a gateway. For more information, see [Establish connectivity to Azure Log Analytics](#).

## DevOps

The following are considerations for integrating your environment with DevOps processes and solutions.

- Application Insights. Integrate Application Insights into Azure Pipelines to help make performance and usability improvements. Application Insights can detect performance anomalies automatically. It connects to various development tools, such as Azure DevOps Services and GitHub.
- Application Instrumentation. *Instrument* applications by modifying application code to enable telemetry with Application Insights. The following methods are ways to instrument applications:
  - At runtime. Instrumenting your web application on the server at runtime is ideal for applications that are deployed already, as it avoids having to update code. Suitable scenarios include:
    - Microsoft ASP.NET or ASP.NET Core applications hosted on Azure Web Apps
    - ASP.NET applications hosted in Microsoft Internet Information Services (IIS) on a virtual machine or virtual machine scale set
    - ASP.NET applications hosted in IIS on an on-premises VM
    - Java-based Azure Functions
    - Node.JS apps on Linux App Services
    - Microservices hosted on AKS
  - At development time. Add Application Insights to your code to customize telemetry collection and send more data. Supported languages and platforms include:
    - ASP.NET applications
    - ASP.NET Core applications
    - .NET Console applications
    - Java
    - Node.js
    - Python
- Use IT Service Management Connector (ITSMC) to connect to external IT Service Management (ITSM) tools. ITSMC connects Azure to supported ITSM products and services, where issue-related work items typically reside. For more information, see [Connect Azure to ITSM tools using IT Service Management Connector](#).

## Performance efficiency

Performance efficiency is the ability of your workload to scale in an efficient manner to meet the demands that your users place on it. For more information, see [Performance efficiency pillar overview](#).

The following are considerations for scaling your environment.

- Automate installation and configuration of your resources and applications.
- Large-scale geographically dispersed applications. Use **Distributed Tracing** within Application Insights to track dependencies and calls across multiple application components, backend resources, and microservices environments. With **Distributed Tracing** you can debug applications that call across process boundaries, outside the local stack. (You don't need to enable **Distributed Tracing**, it's available automatically as part of App Insights.)
  - Two options for consuming distributed trace data are:
    - Transaction diagnostics experience. This experience is similar to a call stack with an added time dimension. The transaction diagnostics experience provides visibility into one single transaction/request. It's helpful for finding the root cause of reliability issues and performance bottlenecks on a per-request basis. For more information, see [What is Distributed Tracing?](#)
    - Application map experience. This aggregates many transactions to demonstrate how systems interact topologically, and provide average performance and error rates. For more information, see [Application Map: Triage Distributed Applications](#).

## Next steps

Learn more about the component technologies:

- [Azure Event Hubs](#) — A big data streaming platform and event ingestion service
- [Azure Monitor](#) overview
- [Overview of Log Analytics in Azure Monitor](#)
- [What are virtual machine scale sets?](#)
- [Overview of autoscale in Microsoft Azure](#)
- [What is Application Insights?](#)

## Related resources

Explore related architectures:

- [Serverless event processing](#)
- [Azure Data Explorer monitoring](#)
- [Unified logging for microservices applications](#)
- [Microservices architecture on Azure Service Fabric](#)

# Manage configurations for Azure Arc-enabled servers

Azure Arc

Azure Monitor

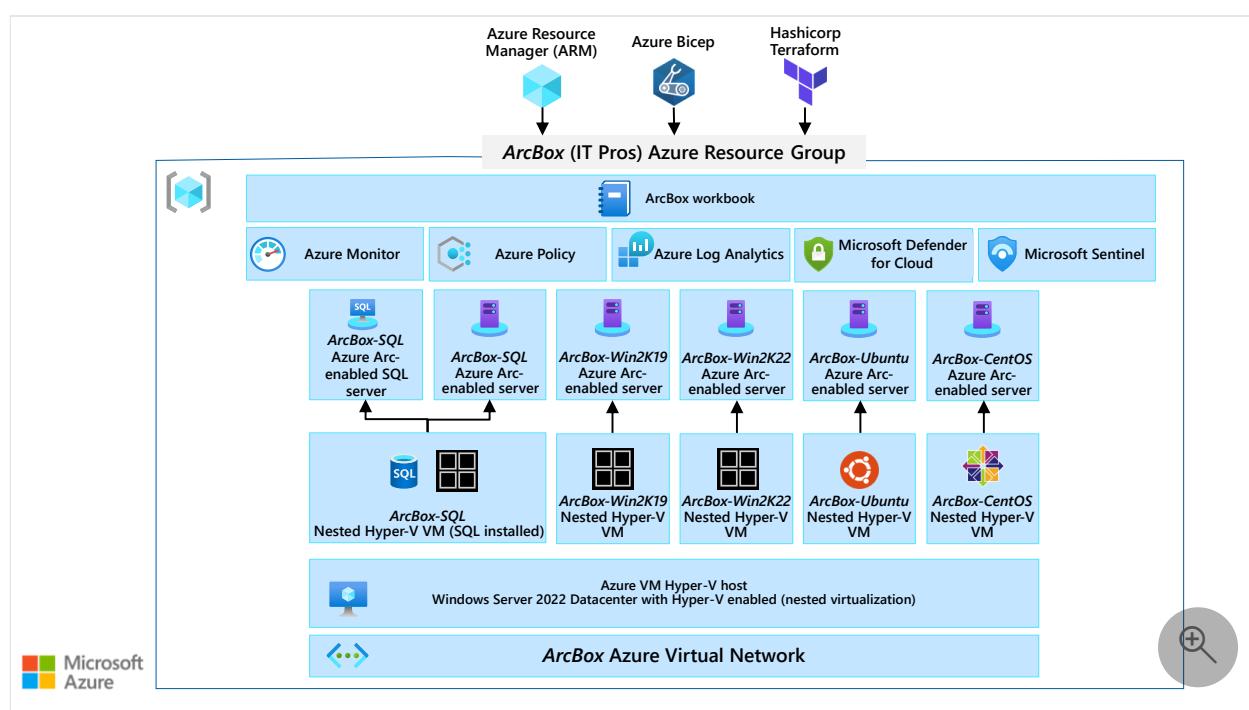
Azure Policy

Azure Resource Manager

Azure Virtual Machines

This reference architecture illustrates how Azure Arc enables you to manage, govern, and secure servers across on-premises, multicloud, and edge scenarios, and is based on the Azure Arc Jumpstart [ArcBox for IT Pros](#) implementation. ArcBox is a solution that provides an easy to deploy sandbox for all things Azure Arc. ArcBox for IT Pros is a version of ArcBox that is intended for users who want to experience Azure Arc-enabled servers capabilities in a sandbox environment.

## Architecture



Download a [PowerPoint file](#) of this architecture.

## Components

The architecture consists of the following components:

- An **Azure Resource Group** is a container that holds related resources for an Azure solution. The resource group can include all the resources for the solution, or only those resources that you want to manage as a group.

- [ArcBox workbook](#) is an Azure Monitor workbook, which provides a single pane of glass for monitoring and reporting on ArcBox resources. The workbook acts as a flexible canvas for data analysis and visualization in the Azure portal, gathering information from several data sources from across ArcBox and combining them into an integrated interactive experience.
- [Azure Monitor](#) enables you to track performance and events for systems running in Azure, on-premises, or in other clouds.
- [Azure Policy guest configuration](#) can audit operating systems and machine configuration both for machines running in Azure and Arc-enabled servers running on-premises or in other clouds.
- [Azure Log Analytics](#) is a tool in the Azure portal to edit and run log queries from data collected by Azure Monitor Logs and interactively analyze their results. You can use Log Analytics queries to retrieve records that match particular criteria, identify trends, analyze patterns, and provide various insights into your data.
- [Microsoft Defender for Cloud](#) is a cloud security posture management (CSPM) and cloud workload protection (CWP) solution. Microsoft Defender for Cloud finds weak spots across your cloud configuration, helps strengthen the overall security posture of your environment, and can protect workloads across multicloud and hybrid environments from evolving threats.
- [Microsoft Sentinel](#) is a scalable, cloud-native, security information and event management (SIEM) and security orchestration, automation, and response (SOAR) solution. Microsoft Sentinel delivers intelligent security analytics and threat intelligence across the enterprise, providing a single solution for attack detection, threat visibility, proactive hunting, and threat response.
- [Azure Arc-enabled servers](#) enables you to connect Azure to your Windows and Linux machines hosted outside of Azure on your corporate network. When a server is connected to Azure, it becomes an Arc-enabled server and is treated as a resource in Azure. Each Arc-enabled server has a Resource ID, a managed system identity, and is managed as part of a resource group inside a subscription. Arc-enabled servers benefit from standard Azure constructs such as inventory, policy, tags, and Azure Lighthouse.
- [Hyper-V nested virtualization](#) is used by Jumpstart ArcBox for IT Pros to host Windows Server virtual machines inside of an Azure virtual machine. This provides the same experience as using physical Windows Server machines, but without the hardware requirements.
- [Azure Virtual Network](#) provides a private network that enables components within the Azure Resource Group to communicate, such as the virtual machines.

## Scenario details

# Potential use cases

Typical uses for this architecture include:

- Organize, govern, and inventory large groups of virtual machines (VMs) and servers across multiple environments.
- Enforce organization standards and assess compliance at scale for all your resources anywhere with Azure Policy.
- Easily deploy supported VM extensions to Arc enabled servers.
- Configure and enforce Azure Policy for VMs and servers hosted across multiple environments.

## Recommendations

The following recommendations apply for most scenarios. Follow these recommendations unless you have a specific requirement that overrides them.

## Configure Azure Arc Connected Machine agent

You can connect any other physical or virtual machine running Windows or Linux to Azure Arc. Before onboarding machines, be sure to complete the [Connected machine agent prerequisites](#), which includes registering the Azure resource providers for Azure Arc-enabled servers. To use Azure Arc to connect the machine to Azure, you need to install the Azure Connected Machine agent on each machine that you plan to connect using Azure Arc. For more information, see [Overview of Azure Arc-enabled servers agent](#).

Once configured, the Connected Machine agent sends a regular heartbeat message every five minutes to Azure. When the heartbeat isn't received, Azure assigns the machine Offline status, which is reflected in the portal within 15 to 30 minutes. Upon receiving a subsequent heartbeat message from the Connected Machine agent, its status will automatically change to Connected.

There are several options available in Azure to connect your Windows and Linux machines:

- Manual installation: Azure Arc-enabled servers can be enabled for one or a few Windows or Linux machines in your environment by using the Windows Admin Center tool set or by performing a set of steps manually.
- Script-based installation: You can perform automated agent installation by running a template script that you download from the Azure portal.

- Connect machines at scale using a service principal: To onboard at scale, use a service principal and deploy via your organization's existing automation.
- Installation using Windows PowerShell DSC

Consult the [Azure Connected Machine agent deployment options](#) for comprehensive documentation on the various deployment options available.

## Enable Azure Policy guest configuration

Azure Arc-enabled servers support [Azure Policy](#) at the Azure resource management layer, and also within the individual server machine using [guest configuration policies](#). Azure Policy guest configuration can audit settings inside a machine, both for machines running in Azure and Arc-enabled servers. For example, you can audit settings such as:

- Operating system configuration
- Application configuration or presence
- Environment settings

There are several [Azure Policy built-in definitions for Azure Arc](#). These policies provide auditing and configuration settings for both Windows and Linux-based machines.

## Enable Azure Update Management

Update Management. You can perform update management for Arc-enabled servers. [Update management](#) in Azure Automation enables you to manage operating system updates and quickly assess the status of available updates on all agent machines. You can also manage the process of installing required updates for servers.

Change Tracking and Inventory. [Azure Automation Change Tracking and Inventory](#) for Arc-enabled servers allows you to determine what software is installed in your environment. You can collect and observe inventory for software, files, Linux daemons, Windows services, and Windows Registry keys. Tracking the configurations of your machines can help you pinpoint operational issues across your environment and better understand the state of your machines.

## Monitor Azure Arc-enabled servers

You can use Azure Monitor to monitor your VMs, virtual machine scale sets, and Azure Arc machines at scale. Azure Monitor analyzes the performance and health of your Windows and Linux VMs, and monitors their processes and dependencies on other resources and external processes. It includes support for monitoring performance and

application dependencies for VMs that are hosted on-premises or in another cloud provider.

The Azure Monitor agents should be automatically deployed to Azure Arc-enabled Windows and Linux servers, through [Azure Policy](#). Review and understand how the [Log Analytics agent](#) operates and collects data before deployment.

Design and plan your Log Analytics workspace deployment. It will be the container where data is collected, aggregated, and later analyzed. A Log Analytics workspace represents a geographical location of your data, data isolation, and scope for configurations like data retention. Use a single Azure Monitor Log Analytics workspace as described in the [management and monitoring best practices](#) of Cloud Adoption Framework.

## Secure Azure Arc-enabled servers

Use Azure RBAC to control and manage the permission for Azure Arc-enabled servers managed identities and perform periodic access reviews for these identities. Control privileged user roles to avoid system-managed identities being misused to gain unauthorized access to Azure resources.

Consider using [Azure Key Vault](#) to manage certificates on your Azure Arc-enabled servers. The key vault VM extension allows you to manage the certificate lifecycle on Windows and Linux machines.

[Connect Azure Arc-enabled servers to Microsoft Defender for Cloud](#). This helps you start collecting security-related configurations and event logs so you can recommend actions and improve your overall Azure security posture.

[Connect Azure Arc-enabled servers to Microsoft Sentinel](#). This enables you to start collecting security-related events and start correlating them with other data sources.

## Validate network topology

The Connected Machine agent for Linux and Windows communicates outbound securely to Azure Arc over TCP port 443. The Connected Machine agent can connect to the Azure control plane using the following methods:

- [Direct connection to Azure public endpoints](#), optionally from behind a firewall or a proxy server.
- [Azure Private Link](#) using a Private Link Scope model to allow multiple servers or machines to communicate with their Azure Arc resources using a single private endpoint.

Consult [Network topology and connectivity for Azure Arc-enabled servers](#) for comprehensive networking guidance for your Arc-enabled servers implementation.

# Considerations

These considerations implement the pillars of the Azure Well-Architected Framework, which is a set of guiding tenets that can be used to improve the quality of a workload. For more information, see [Microsoft Azure Well-Architected Framework](#).

## Reliability

- In most cases, the location you select when you create the installation script should be the Azure region geographically closest to your machine's location. The rest of the data will be stored within the Azure geography containing the region you specify, which might also affect your choice of region if you have data residency requirements. If an outage affects the Azure region to which your machine is connected, the outage won't affect the Arc-enabled server. However, management operations using Azure might not be available.
- If you have multiple locations that provide a geographical-redundant service, it's best to connect the machines in each location to a different Azure region for resilience in the event of a regional outage.
- If the Azure connected machine agent stops sending heartbeats to Azure, or goes offline, you will not be able to perform operational tasks on it. Hence, it's necessary to [develop a plan for notifications and responses](#).
- Set up [resource health alerts](#) to get notified in near real-time when resources have a change in their health status. And define a monitoring and alerting policy in [Azure Policy](#) that identifies unhealthy Azure Arc-enabled servers.
- Extend your current backup solution to Azure, or easily configure our application-aware replication and application-consistent backup that scales based on your business needs. The centralized management interface for [Azure Backup](#) and [Azure Site Recovery](#) makes it simple to define policies to natively protect, monitor, and manage your Arc-enabled Windows and Linux servers.
- Review the [business continuity and disaster recovery](#) guidance to determine whether your enterprise requirements are met.
- Other reliability considerations for your solution are described in the [reliability design principles](#) section in the Microsoft Azure Well-Architected Framework.

## Security

- Appropriate Azure role-based access control (Azure RBAC) should be managed for Arc-enabled servers. To onboard machines, you must be a member of the **Azure Connected Machine Onboarding** role. To read, modify, re-onboard, and delete a machine, you must be a member of the **Azure Connected Machine Resource Administrator** role.
- Microsoft Defender for Cloud can monitor on-premises systems, Azure VMs, Azure Monitor resources, and even VMs hosted by other cloud providers. Enable Microsoft Defender for servers for all subscriptions containing Azure Arc-enabled servers for security baseline monitoring, security posture management, and threat protection.
- Microsoft Sentinel can help simplify data collection across different sources, including Azure, on-premises solutions, and across clouds using built-in connectors.
- You can use Azure Policy to manage security policies across your Arc-enabled servers, including implementing security policies in Microsoft Defender for Cloud. A security policy defines the desired configuration of your workloads and helps ensure you're complying with the security requirements of your company or regulators. Defender for Cloud policies are based on policy initiatives created in Azure Policy.
- To limit which extensions can be installed on your Arc-enabled server, you can configure the lists of extensions you wish to allow and block on the server. The extension manager will evaluate all requests to install, update, or upgrade extensions against the allowlist and blocklist to determine if the extension can be installed on the server.
- [Azure Private Link](#) allows you to securely link Azure PaaS services to your virtual network using private endpoints. You can connect your on-premises or multicloud servers with Azure Arc and send all traffic over an Azure ExpressRoute or site-to-site VPN connection instead of using public networks. You can use a Private Link Scope model to allow multiple servers or machines to communicate with their Azure Arc resources using a single private endpoint.
- Consult [Azure Arc-enabled servers security overview](#) for a comprehensive overview of the security features in Azure Arc-enabled server.
- Other security considerations for your solution are described in the [security design principles](#) section in the Microsoft Azure Well-Architected Framework.

## Cost optimization

- Azure Arc control plane functionality is provided at no extra cost. This includes support for resource organization through Azure management groups and tags, and access control through Azure role-based access control (RBAC). Azure services

used in conjunction to Azure Arc-enabled servers incur costs according to their usage.

- Consult [Cost governance for Azure Arc-enabled servers](#) for additional Azure Arc cost optimization guidance.
- Other cost optimization considerations for your solution are described in the [Principles of cost optimization](#) section in the Microsoft Azure Well-Architected Framework.
- Use the [Azure pricing calculator](#) to estimate costs.
- When deploying the Jumpstart ArcBox for IT Pros reference implementation for this architecture, keep in mind ArcBox resources generate Azure Consumption charges from the underlying Azure resources. These resources include core compute, storage, networking and auxiliary services.

## Operational excellence

- Automate the deployment of your Arc-enabled servers environment. The [reference implementation](#) of this architecture is fully automated using a combination of Azure ARM templates, VM extensions, Azure Policy configurations, and PowerShell scripts. You can also reuse these artifacts for your own deployments. Consult [Automation disciplines for Azure Arc-enabled servers](#) for additional Arc-enabled servers automation guidance in the Cloud Adoption Framework (CAF).
- There are several options available in Azure to automate the [onboarding of Arc-enabled servers](#). To onboard at scale, use a service principal and deploy via your organizations existing automation platform.
- VM extensions can be deployed to Arc-enabled servers to simplify the management of hybrid servers throughout their lifecycle. Consider automating the deployment of VM extensions via Azure Policy when managing servers at scale.
- Enable patch and Update Management in your onboarded Azure Arc-enabled servers to ease OS lifecycle management.
- Review [Azure Arc Jumpstart Unified Operations Use Cases](#) to learn about additional operational excellence scenarios for Azure Arc-enabled servers.
- Other operational excellence considerations for your solution are described in the [Operational excellence design principles](#) section in the Microsoft Azure Well-Architected Framework.

## Performance efficiency

- Before configuring your machines with Azure Arc-enabled servers, you should review the Azure Resource Manager [subscription limits](#) and [resource group limits](#) to plan for the number of machines to be connected.

- A phased deployment approach as described in the [deployment guide](#) can help you determine the resource capacity requirements for your implementation.
- Use Azure Monitor to collect data directly from your Azure Arc-enabled servers into a Log Analytics workspace for detailed analysis and correlation. Review the [deployment options](#) for the Azure Monitor agents.
- Additional performance efficiency considerations for your solution are described in the [Performance efficiency principles](#) section in the Microsoft Azure Well-Architected Framework.

## Deploy this scenario

The reference implementation of this architecture can be found in the [Jumpstart ArcBox for IT Pros](#), included as part of the [Arc Jumpstart](#) project. ArcBox is designed to be completely self-contained within a single Azure subscription and resource group. ArcBox makes it easy for a user to get hands-on experience with all available Azure Arc technology with nothing more than an available Azure subscription.

To deploy the reference implementation, follow the steps in the GitHub repo by selecting the **Jumpstart ArcBox for IT Pros** button below.

[Jumpstart ArcBox for IT Pros](#)

## Contributors

*This article is maintained by Microsoft. It was originally written by the following contributors.*

Principal author:

- [Pieter de Bruin](#) | Senior Program Manager

*To see non-public LinkedIn profiles, sign in to LinkedIn.*

## Next steps

- [Learn more about Azure Arc](#)
- [Learn more about Azure Arc-enabled servers](#)
- [Azure Arc learning path](#)
- [Review Azure Arc Jumpstart scenarios](#) in the Arc Jumpstart
- [Review Arc-enabled servers landing zone accelerator](#) in CAF

# Related resources

Explore related architectures:

- [Manage configurations for Azure Arc-enabled servers](#)
- [Azure Arc hybrid management and deployment for Kubernetes clusters](#)

# Manage hybrid Azure workloads using Windows Admin Center

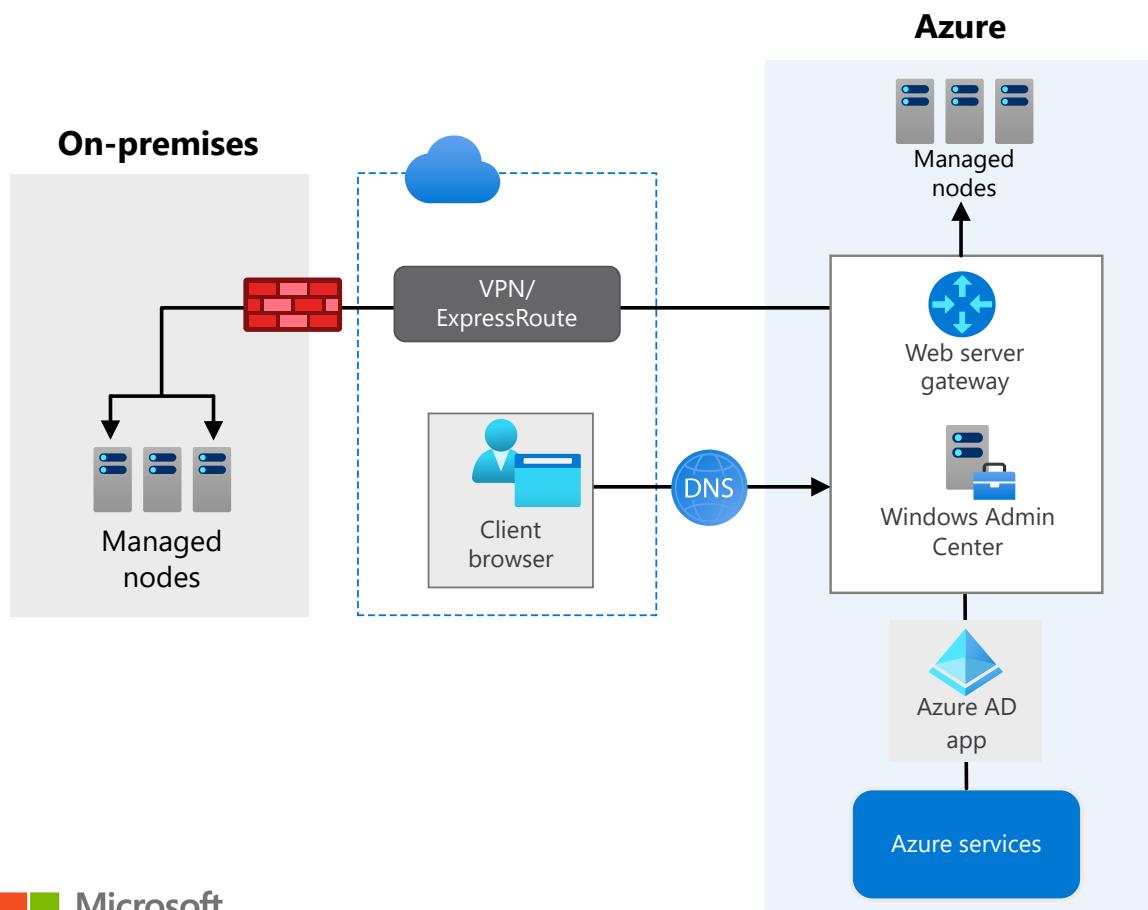
[Microsoft Entra ID](#) [Azure Key Vault](#) [Azure Portal](#) [Azure Virtual Machines](#)

This reference architecture illustrates how to design a hybrid Windows Admin Center solution to manage workloads that are hosted on-premises and in Microsoft Azure. This architecture includes two scenarios:

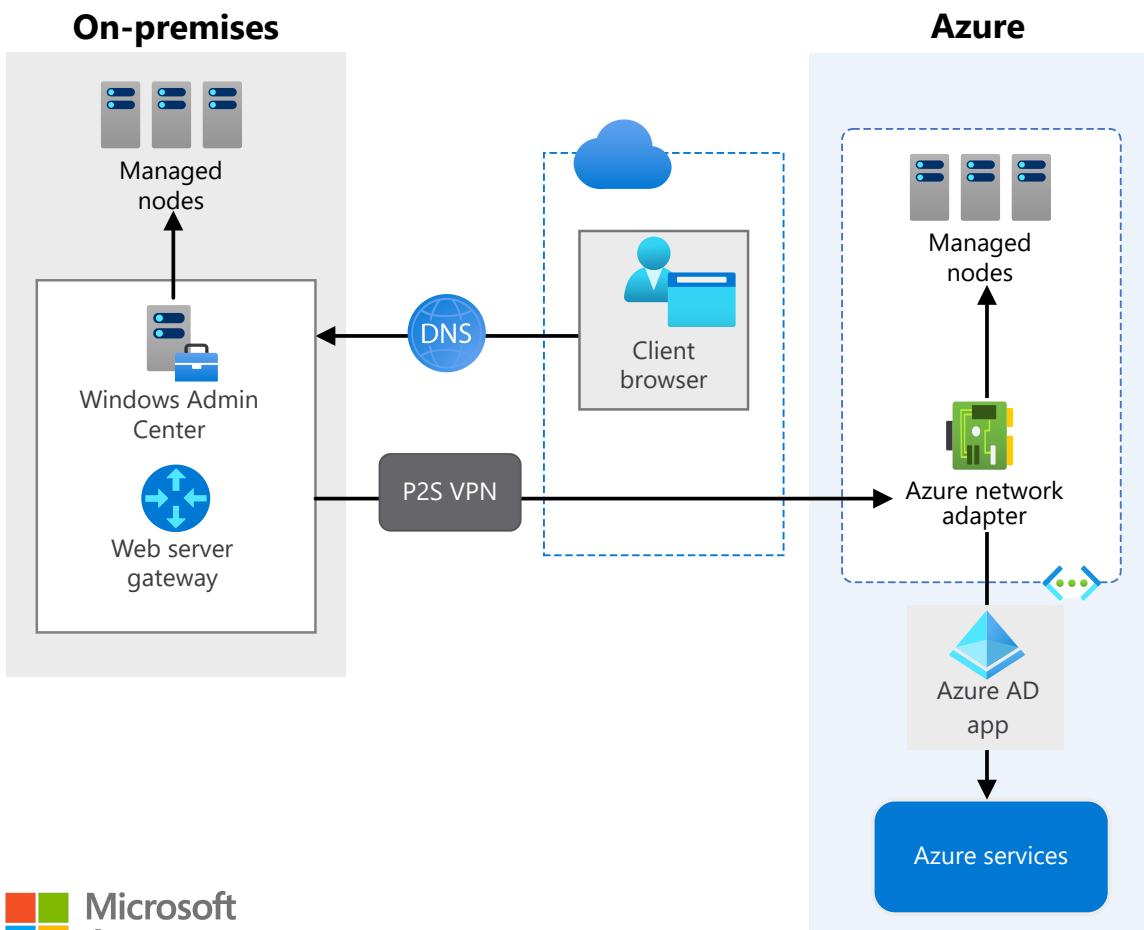
- Windows Admin Center deployed to a virtual machine (VM) in Azure.
- Windows Admin Center deployed to a server (physical or virtual) on-premises.

## Architecture

The first diagram illustrates Windows Admin Center deployed to a VM in Azure.



The second diagram illustrates Windows Admin Center deployed on-premises.



Download a [Visio file](#) of all architectures diagrams in this article.

## Workflow

The architecture consists of the following:

- **On-premises corporate network.** A private local area network that runs within an organization.
- **On-premises corporate firewall.** An organizational firewall configured to allow users access to the Windows Admin Center gateway when the gateway is deployed on-premises.
- **Windows VM.** A Windows VM installed with the Windows Admin Center gateway that hosts web services that users can connect to.
- **Microsoft Entra app.** An app used for all points of Azure integration in Windows Admin Center, including Microsoft Entra authentication to the gateway.
- **Managed nodes.** Nodes managed by Windows Admin Center, which might include physical servers running Azure Stack, or Windows Server or virtual machines running Windows Server.
- **VPN or ExpressRoute.** A Site-to-Site (S2S) VPN or ExpressRoute to manage nodes on-premises when Windows Admin Center is deployed in Azure.

- **Azure network adapter.** An adapter for a Point-to-Site (P2S) VPN to Azure when Windows Admin Center is deployed on-premises. Windows Admin Center can automatically deploy the adapter and also create an Azure gateway.
- **Domain Name System (DNS).** A DNS record that users reference to connect to the Windows Admin Center gateway.

## Components

- [Windows Admin Center](#) is a browser-based management tool set that gives you full control over all aspects of your server infrastructure and is particularly useful for managing servers on private networks that are not connected to the Internet. It accesses servers through the Windows Admin Center gateway that's installed on Windows Server or on domain-joined Windows 10. The gateway can be installed on-premises or on an Azure VM that runs Windows.
- [Azure ExpressRoute](#) creates private connections between Azure datacenters and infrastructure on premises or in a colocation environment.
- [Azure Virtual Network](#) provides secure network infrastructure in the cloud.
- [Azure Virtual Machines](#) provides Linux and Windows virtual machines. In this solution, you can use it to provide a Windows VM to run Windows Admin Center.

## Scenario details

### Potential use cases

Typical uses for this architecture include:

- Organizations that want to manage individual Windows Server instances, Hyper-Converged Infrastructure or Hyper-V VMs that run on-premises and in Microsoft Azure.
- Organizations that want to manage Windows Server instances hosted by other cloud providers.

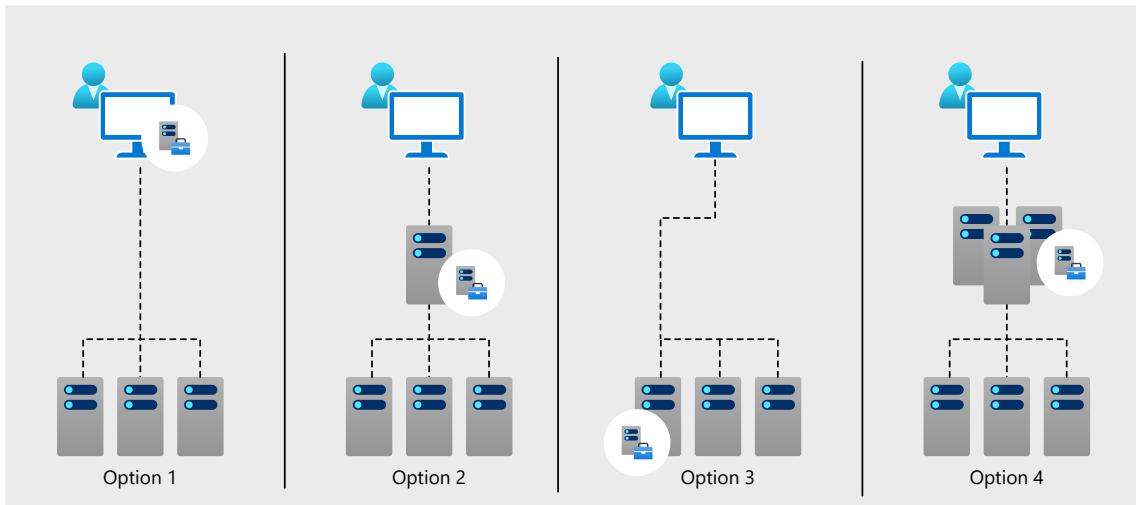
## Recommendations

The following recommendations apply for most scenarios. Follow these recommendations unless you have a specific requirement that overrides them.

### Installation types

You have multiple options when deploying Windows Admin Center, including installing on a Windows 10 PC, a Windows server, or an Azure VM. The deployment type you choose will depend on your business requirements.

The on-premises installation types are:



- **Option 1: Local Client.** Deploy Windows Admin Center on a Windows 10 client. This is ideal for quick deployments, testing, and improvised or small-scale scenarios.
- **Option 2: Gateway Server.** Install on a designated gateway server running Windows Server 2016, Windows Server 2019, or later, and access from any client browser with connectivity to the gateway server.
- **Option 3: Managed Server.** Install directly on a managed server running Windows Server 2016, Windows Server 2019, or later to enable the server to manage itself or a cluster in which the managed server is a member node. This is great for distributed branch office scenarios.
- **Option 4: Failover Cluster.** Deploy in a failover cluster to enable high availability of the gateway service. This is great for production environments to ensure management service resiliency.

Because Windows Admin Center has no cloud service dependencies, some organizations might choose to deploy Windows Admin Center to an on-premises system for managing on-premises computers and then enable hybrid services over time. However, many organizations prefer to take advantage of service offerings in Azure and other cloud platforms that are integrated with Windows Admin Center.

Deploying Windows Admin Center on an Azure VM provides gateway functionality similar to Windows Server 2016 and Windows Server 2019. However, Azure also enables

additional features for Azure VMs. Refer to [Reliability](#) for more information about the benefits of deploying Windows Admin Center to Azure VMs.

## Automated deployment

Microsoft provides an .msi file that you use to deploy Windows Admin Center to an on-premises VM. The .msi file installs Windows Admin Center and automatically generates a self-signed certificate or associates an existing certificate based on your preference.

### 💡 Tip

For more information about using the .msi file to deploy Windows Admin Center, refer to [Install Windows Admin Center](#).

When implementing Windows Admin Center on an Azure VM, Microsoft provides the **Deploy-Windows Admin CenterAzVM.ps1** script to automatically deploy all required resources. In this scenario, the script deploys a new Azure VM with Windows Admin Center installed and opens port 443 on the public IP address. The script can also deploy Windows Admin Center to an existing Azure VM. When running the script, you can use variables to specify the resource group, virtual network name, VM name, location, and many other options.

### ⓘ Important

Prior to deployment, upload the certificate to an Azure Key Vault. Alternatively, you can use the Azure portal to generate a certificate.

### 💡 Tip

For more information about using the **Deploy-Windows Admin CenterAzVM.ps1** script to deploy Windows Admin Center to an Azure VM, refer to [Deploy Windows Admin Center in Azure](#).

### 💡 Tip

For more information about the manual steps required to deploy Windows Admin Center to an Azure VM, refer to [Deploy manually on an existing Azure virtual machine](#).

## Topology recommendations

While you can implement Windows Admin Center on any existing or new VM that's on-premises or hosted in Azure, Microsoft recommends deploying Windows Admin Center gateway on a Windows Server 2019 Azure VM. The automated deployment option discussed earlier allows you to implement Windows Admin Center on an Azure VM more quickly while still safeguarding your environment. Instead of deploying Windows Admin Center on an on-premises VM, you can minimize the configuration changes required for an on-premises firewall and network.

**ⓘ Important**

Managing nodes on-premises requires a connection (such as S2S VPN or ExpressRoute) from Azure to on-premises.

When deployed on-premises, you can use Windows Admin Center to build the hybrid connection to Azure. The hybrid connection, called an *Azure network adapter*, is a P2S VPN to Azure that allows Windows Admin Center to access hybrid cloud features. This process also automatically creates an Azure gateway for the VPN connection. For more information, refer to [About Point-to-Site VPN](#).

**ⓘ Important**

You must have an Azure virtual network in Azure before creating an Azure network adapter.

You should also configure the Windows Admin Center gateway for high availability. This will help ensure that management of systems and services remain available during unexpected failures. Azure provides multiple service offerings for these scenarios regardless of whether Windows Admin Center gateway is deployed to an on-premises or Azure VM. Refer to *Availability Considerations* for more information.

## Certificate recommendations

Windows Admin Center gateway is a web-based tool that uses a Secure Sockets Layer (SSL) certificate to encrypt web traffic for administrators who are using the gateway. Windows Admin Center allows you to use an SSL certificate that a trusted third-party certification authority (CA) created from a self-signed certificate. You will receive a warning when trying to connect from a browser if you choose the latter option. As a result, we recommend that you only use self-signed certificates for test environments.

## 💡 Tip

To learn how other Microsoft customers have used Windows Admin Center to improve their productivity and reduce costs, refer to [Windows Admin Center Case Studies](#).

## Administrative recommendations

Deploying Windows Admin Center on a local Windows 10 client is great for quick-start tests, and ad-hoc or small-scale scenarios. However, for production scenarios with multiple administrators, we recommend Windows Server. When deployed on Windows Server, Windows Admin Center provides a centralized management hub for your server environment with additional capabilities such as smart card authentication, conditional access, and multifactor authentication. For more information about controlling access to Windows Admin Center, refer to [User Access Options with Windows Admin Center](#).

## Considerations

These considerations implement the pillars of the Azure Well-Architected Framework, which is a set of guiding tenets that can be used to improve the quality of a workload. For more information, see [Microsoft Azure Well-Architected Framework](#).

## Reliability

Reliability ensures that your application can meet the commitments that you make to your customers. For more information, see [Overview of the reliability pillar](#).

- You can help ensure high availability of the Windows Admin Center gateway service by deploying it in an active/passive model on a failover cluster. In this scenario, only one instance of the Windows Admin Center gateway service is active. If one of the nodes in the cluster fails, the Windows Admin Center gateway service seamlessly fails over to another node.

### ✖ Caution

Deploying Windows Admin Center on a Windows 10 client computer doesn't provide high availability because gateway functionality isn't included with the Windows 10 deployment. Deploy the Windows Admin Center gateway to Windows Server 2016 or Windows Server 2019.

- To help ensure high availability of Windows Admin Center data in case of a node failure, configure a Cluster Shared Volume (CSV) into which Windows Admin Center will store persistent data that all the nodes in the cluster access. You can deploy the failover cluster for Windows Admin Center gateway by using an automated deployment script. The `Install-WindowsAdminCenterHA.ps1` script automatically deploys the failover cluster, configures IP addresses for the cluster service, installs Windows Admin Center gateway, configures the port number for the gateway service, and configures the web service with the appropriate certificate.

 **Tip**

For more information about using the automated deployment script, refer to [Deploy Windows Admin Center with high availability](#).

- Deploying Windows Admin Center gateway to an Azure VM provides additional high-availability options. For example, by using availability sets you can provide VM redundancy and availability within an Azure datacenter by distributing the VMs across multiple hardware nodes. You can also use availability zones in Azure, which provide datacenter fault tolerance. Availability zones are unique physical locations that span datacenters within an Azure region. This helps ensure that Azure VMs have independent power, cooling, and networking. For more information on high availability, refer to [Availability options for virtual machines in Azure](#) and [High availability and disaster recovery for IaaS apps](#).

## Security

Security provides assurances against deliberate attacks and the abuse of your valuable data and systems. For more information, see [Overview of the security pillar](#).

- Deploying Windows Admin Center provides your organization with a centralized management interface for your server environment. By controlling access to Windows Admin Center, you can improve the security of your management landscape.
- Windows Admin Center provides multiple features to help secure your management platform. For starters, Windows Admin Center gateway authentication can use local groups, Active Directory Domain Services (AD DS), and cloud-based Microsoft Entra ID. You can also enforce smart card authentication by specifying an additional required group for smart card-based security groups. And, by requiring Microsoft Entra authentication for the gateway, you can use other

Microsoft Entra security features such as conditional access and Microsoft Entra multifactor authentication.

- Access to the Windows Admin Center gateway doesn't imply access to the target servers that are made visible by the gateway. To manage a target server, users must connect with credentials that grant administrator privileges on the servers they want to manage. However, some users might not need administrative access to perform their responsibilities. In this scenario, you can use role-based access control (RBAC) in Windows Admin Center to provide these users with limited access to servers rather than granting them full administrative access.

 **Note**

If you deployed Local Administrator Password Solution (LAPS) in your environment, use LAPS credentials through Windows Admin Center to authenticate with a target server.

- In Windows Admin Center, RBAC works by configuring each managed server with a Windows PowerShell *Just Enough Administration* endpoint. This endpoint defines the roles, including which parts of the system each role can manage and which users are assigned to the roles. When a user connects to the endpoint, RBAC creates a temporary local administrator account to manage the system on their behalf and automatically removes the account when the user stops managing the server through the Windows Admin Center. For more information, refer to [Just Enough Administration](#).
- Windows Admin Center also provides visibility into the management actions performed in your environment. Windows Admin Center records events by logging actions to the **Microsoft-ServerManagementExperience** event channel in the event log of the managed server. This allows you to audit actions that administrators perform, and helps you troubleshoot Windows Admin Center issues and review usage metrics. For more information on auditing, refer to [Use event logging in Windows Admin Center to gain insight into management activities and track gateway usage](#).

## Cost optimization

Cost optimization is about looking at ways to reduce unnecessary expenses and improve operational efficiencies. For more information, see [Overview of the cost optimization pillar](#).

- For on-premises deployments, Windows Admin Center costs nothing beyond the cost of the Windows operating system, which requires valid Windows Server or

Windows 10 licenses.

- For Azure deployments, plan for costs associated with deploying Windows Admin Center to an Azure VM. Some of these costs might include the VM, storage, static or public IP addresses (if enabled), and any networking components required for integration with on-premises environments. In addition, you might need to plan for the cost of purchasing non-Microsoft CA certificates.

## Operational excellence

Operational excellence covers the operations processes that deploy an application and keep it running in production. For more information, see [Overview of the operational excellence pillar](#).

## Manageability

- Access to the Windows Admin Center management tool set depends on the installation type. For example, in a local client scenario, you would open the Windows Admin Center from the **Start** menu and connect to it from a client web browser by accessing `https://localhost:6516`. In other scenarios where you deploy the Windows Admin Center gateway to a Window Server, you can connect to the Windows Admin Center gateway from a client web browser by accessing the server name URL, such as `https://servername.contoso.com`, or the URL of the cluster name.
- When deployed on Windows Server, Windows Admin Center provides a centralized point of management for your server environment. Windows Admin Center provides management tools for many common scenarios and tools, including:
  - Certificate management
  - Event Viewer
  - File Explorer
  - Network settings
  - Remote Desktop Connection
  - Windows PowerShell
  - Firewall management
  - Registry editing
  - Enabling and disabling roles and features
  - Managing installed apps and devices
  - Managing scheduled tasks
  - Configuring local users and groups
  - Managing Windows services
  - Managing storage

- Managing Windows Update

For a complete list of server management capabilities, refer to [Manage Servers with Windows Admin Center](#).

 **Important**

Windows Admin Center doesn't replace all Remote Server Administration Tools (RSAT), such as Internet Information Services (IIS), because there is no equivalent management capability for it in Windows Admin Center.

 **Note**

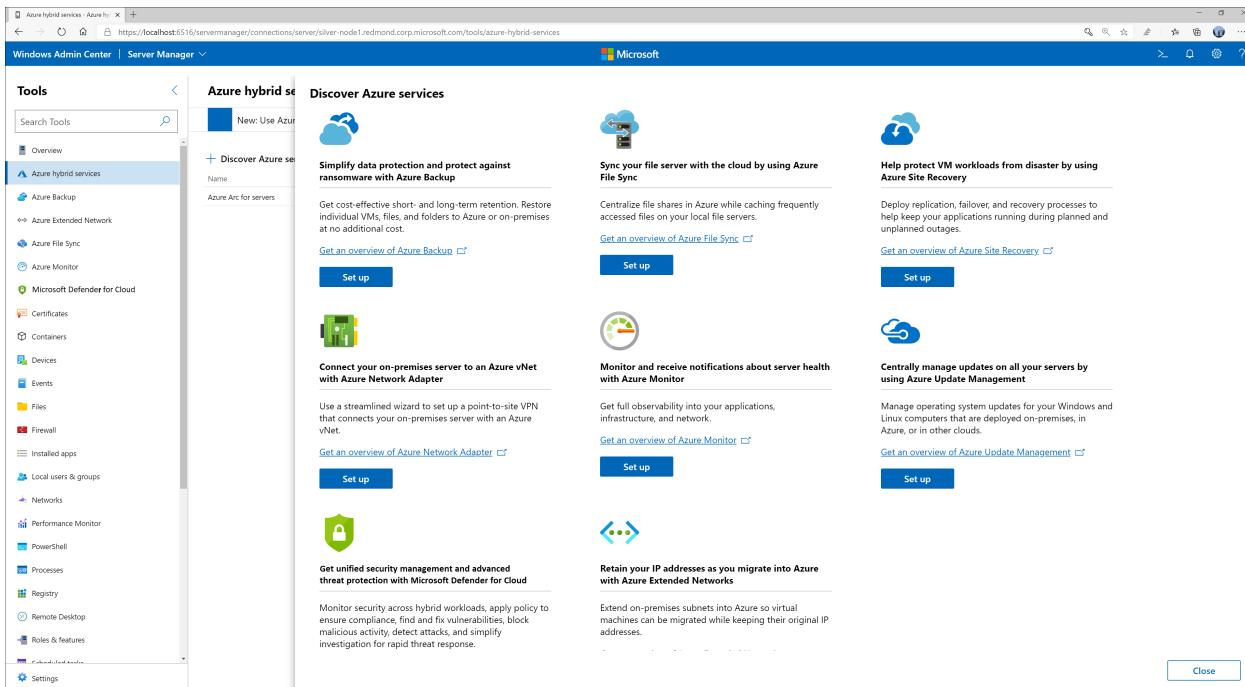
Windows Admin Center provides a subset of Server Manager features for managing Windows 10 client PCs.

- Windows Admin Center provides support for managing failover clusters and cluster resources, managing Hyper-V VMs and virtual switches, and managing Windows Server Storage Replica. In addition to using Windows Admin Center to manage and monitor an existing Hyper-Converged Infrastructure, you can also use Windows Admin Center to deploy a new Hyper-Converged Infrastructure. By using a multistage workflow, Windows Admin Center guides you through installing features, configuring networking, creating a cluster, and deploying Storage Spaces Direct and Software-Defined Networking. For more information, refer to [Manage Hyper-Converged Infrastructure with Windows Admin Center](#).

 **Note**

You can use Windows Admin Center to manage Microsoft Hyper-V Server 2016 or Microsoft Hyper-V Server 2019 (the free Microsoft virtualization product).

- The Azure hybrid services tool in Windows Admin Center provides a centralized location for all the integrated Azure services. You can use Azure hybrid services to protect VMs in Azure and on-premises with cloud-based backup and disaster recovery. You can also extend on-premises capacity with storage and compute in Azure, and you can simplify network connectivity to Azure. With the help of cloud-intelligent Azure management services, you can centralize monitoring, governance, configuration, and security across your applications, network, and infrastructure.



## 💡 Tip

For more information on Azure integration, refer to [Connecting Windows Server to Azure hybrid services](#).

## ⓘ Important

The Microsoft Entra app requires Azure integration in Windows Admin Center.

## DevOps

- Windows Admin Center was built for extensibility so that Microsoft and other developers can build tools and solutions beyond the current offerings. Windows Admin Center provides an extensible platform in which each connection type and tool is an extension that you can install, uninstall, and update individually. Microsoft offers a software development kit (SDK) that enables developers to build their own tools for Windows Admin Center.
- You can build Windows Admin Center extensions using modern web technologies—including HTML5, CSS, Angular, TypeScript, and jQuery—to manage target servers via Windows PowerShell or Windows Management Instrumentation. You can also manage target servers, services, and devices over different protocols such as Representational State Transfer (REST) by building a Windows Admin Center gateway plugin.

## 💡 Tip

For more information about using the SDK to develop extensions for Windows Admin Center, refer to [Extensions for Windows Admin Center](#).

# Contributors

*This article is maintained by Microsoft. It was originally written by the following contributors.*

Principal author:

- [Mike Martin](#) | Senior Cloud Solution Architect

*To see non-public LinkedIn profiles, sign in to LinkedIn.*

# Next steps

More about Azure Automation:

- [Install Windows Admin Center](#)
- [Preview: Use Windows Admin Center in the Azure portal to manage a Windows Server VM](#)
- [Manually deploy Windows Admin Center in Azure for managing multiple servers](#)
- [Connect hybrid machines to Azure from Windows Admin Center](#)

# Related resources

- [Connect standalone servers by using Azure Network Adapter](#)
- [Use Azure Stack HCI stretched clusters for disaster recovery](#)
- [Manage configurations for Azure Arc-enabled servers](#)
- [Use Azure Stack HCI switchless interconnect and lightweight quorum for remote office or branch office](#)

# Deploy a line-of-business application using Azure App Service Environment v3

Azure App Service

Azure DNS

Azure Monitor

Azure Log Analytics

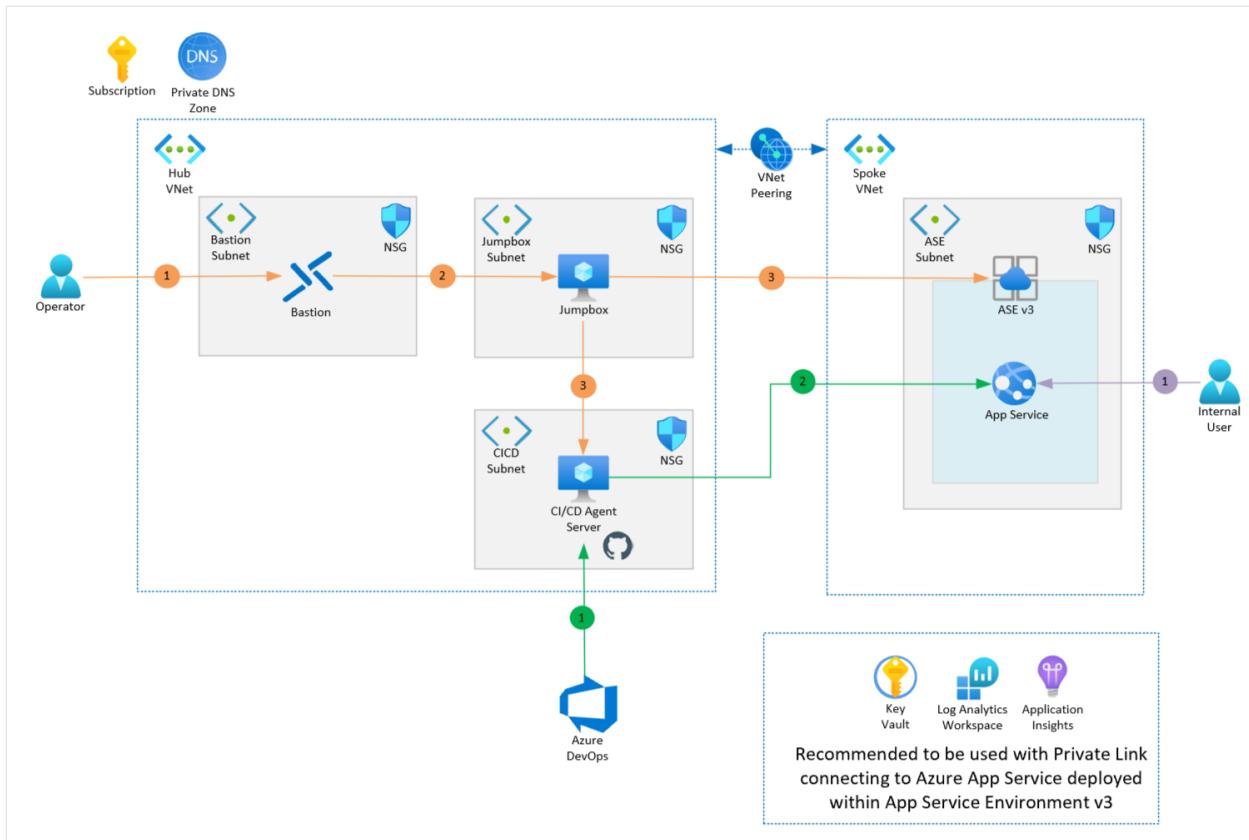
Azure Key Vault

For customers in segments that are tightly governed and restricted by compliance, it's important to have an isolated and dedicated environment, especially for line-of-business applications. While security is front and center, these critical applications also require the ability to scale and perform under scenarios of high memory utilization or high requests per second. This solution provides an example for how you can host line-of-business applications. You can use Azure App Service Environment to ensure that both security and performance can be addressed simultaneously. When deploying this solution, you'll have the flexibility to use existing resources in your [Azure landing zone](#), which represents your resources in the hub VNet. Or, you can deploy this solution as a self-contained workload.

## Note

This article provides a deployable architecture that aligns to our [Landing zone accelerator for App Service](#).

## Architecture



Download a [Visio file](#) of this architecture.

## Workflow

There are three flows with callouts in this architecture: Operations (orange), Deployment (green) and User (purple).

## Operations

1. Operators or administrators will want to perform administration tasks on the continuous integration/continuous deployment (CI/CD) server, or on the Kudu endpoint for the App Service Environment (ASE). First, they'll need to connect to the Azure Bastion host.
2. By using the Bastion host, the operator or administrator can then use Remote Desktop Protocol (RDP) to access the jumpbox server.
3. From the jumpbox server, the operator or administrator can RDP into the CI/CD server and perform the required tasks, such as agent upgrades, OS upgrades, and so on. The operator or administrator can also connect from the jumpbox server to the Kudu endpoint of the ASE instance, to perform administrative tasks or to perform advanced troubleshooting.

## Deployment

1. Deployment of the solution is performed via the CI/CD agent server. The DevOps agent on this server will connect with Azure Pipelines when a new deployment is executed.
2. The artifacts will then be deployed to the App Service by connecting to the App Service Environment (ASE) over the VNet peering.

## User

1. Users can connect to the deployed App Service over the company's network. They can use Azure ExpressRoute or a VPN if needed, and/or over any applicable Azure VNet peering.

## Components

The solution uses the following Azure services:

- **Azure App Service Environment v3 (ASEv3)** is a feature of [Azure App Service](#) and is a single-tenant service for customers that require high scale, network isolation, security, and/or high memory utilization. Apps are hosted in [App Service plans](#) that are created in ASEv3, with options of using different tiers within an Isolated v2 service plan. Compared to an earlier version of ASE, numerous improvements have been made including, but not limited to, network dependency, scale time, and the removal of the stamp fee. This solution uses an App Service Environment v3 that's configured for internal access.
- **Azure Private DNS** allows you to manage and resolve domain names within a virtual network, without needing to implement a custom DNS solution. An [Azure Private DNS zone](#) can be aligned to one or more virtual networks through [virtual network links](#). Due to the internal nature of the ASEv3 that this reference architecture uses, a private DNS zone is required to resolve the domain names of applications that are hosted on the App Service Environment.
- **Azure Application Insights** is a feature of [Azure Monitor](#) that helps developers detect anomalies, diagnose issues, and understand usage patterns. Application Insights features extensible application performance management and monitoring for live web apps. Various platforms are supported, including .NET, Node.js, Java, and Python. It supports apps that are hosted in Azure, on-premises, in a hybrid environment, or in other public clouds. Application Insights is included as part of this reference architecture, to monitor the behaviors of the deployed application.
- **Azure Log Analytics** is a feature of [Azure Monitor](#) that allows you to edit and run log queries with data in Azure Monitor Logs, optionally from within the Azure

portal. Developers can run simple queries for a set of records or use Log Analytics to perform an advanced analysis. They can then visualize the results. Log Analytics is configured as part of this reference architecture, to aggregate all the monitoring logs for analysis and reporting.

- [Azure Virtual Machines](#) is an on-demand, scalable computing resource that can be used to host several different workloads. In this reference architecture, virtual machines are used to provide a management jumpbox server, and to provide a host for the DevOps agent or GitHub runner.
- [Azure Key Vault](#) is a cloud service that securely stores and accesses secrets, which range from API keys and passwords to certificates and cryptographic keys. An Azure Key Vault is deployed as part of this architecture's infrastructure, to facilitate secret management for future code deployments.
- [Azure Bastion](#) is a platform-as-a-service that's provisioned within the developer's virtual network. It provides secure RDP/SSH connectivity to the developer's virtual machines over TLS, from the Azure portal. With Azure Bastion, virtual machines no longer require a public IP address to connect via RDP/SSH. This reference architecture uses Azure Bastion to access the DevOps agent or GitHub runner server or the management jumpbox server.

## Alternatives

Consider adding an [Azure Application Gateway](#) before the App Service instance, to provide Web Application Firewall (WAF) functionality to protect web applications from common exploits and vulnerabilities.

A [self-hosted GitHub runner](#) can be used in place of the Azure DevOps self-hosted agent.

## Considerations

These considerations implement the pillars of the Azure Well-Architected Framework, which is a set of guiding tenets that can be used to improve the quality of a workload. For more information, see [Microsoft Azure Well-Architected Framework](#).

## Reliability

Reliability ensures your application can meet the commitments you make to your customers. For more information, see [Overview of the reliability pillar](#).

- Consider your requirements for zone redundancy in this reference implementation, as well as the zone redundancy capabilities of any other Azure Services in your solution. ASEv3 supports zone redundancy by spreading instances to all three zones in the target region. This configuration can only be set at the time of the ASE creation, and it might not be available in all regions. See more information, see [Availability zone support for App Service Environment](#). This reference implementation implements zone redundancy, but you can change it by cloning this repo and setting the `zoneRedundant` property to `false`.

## Security

Security provides assurances against deliberate attacks and the abuse of your valuable data and systems. For more information, see [Overview of the security pillar](#).

- Employ the appropriate use of access restrictions, so that the app service is only reachable from valid locations. For example, if the app service is hosting APIs, and it's fronted by APIM, you can set up an access restriction so that the app service is only accessible from APIM.
- Since this reference implementation deploys an ASE into a virtual network (referred to as an internal ASE), all applications deployed to the ASE are inherently network-isolated, at the scope of the virtual network.
- Store application secrets (database credentials, API tokens, and private keys) in Azure Key Vault. Configure your App Service app to access them securely with a managed identity. Determine when to use [Azure Key Vault vs Azure App Configuration](#).

## Cost optimization

Cost optimization is about looking at ways to reduce unnecessary expenses and improve operational efficiencies. For more information, see [Overview of the cost optimization pillar](#).

- Although there's no stamp fee for an ASEv3 instance, there's a charge that's levied when no App Service Plans are configured within the ASEv3 instance. This charge is levied at the same rate as one instance of a Windows 11v2 instance, for the region in which the ASEv3 instance is deployed.
- When configured to be zone redundant, the charging model is adjusted to account for the underlying infrastructure that's deployed in this configuration. You might be liable for additional instances, as per [ASEv3 Pricing](#).
- For ASEv3 App Service plans (known as Isolated v2 App Service plans), use [Azure Reservations](#) and [Azure savings plan for compute](#) with a one-year or three-year

contract and receive significant savings off pay-as-you-go prices. For more information, see [How reservation discounts apply to Isolated v2 instances](#).

## Operational excellence

Operational excellence covers the operations processes that deploy an application and keep it running in production. For more information, see [Overview of the operational excellence pillar](#).

- Use Application Insights or another application performance management solution to monitor and learn how your application behaves in different environments.
  - There are two ways to enable [Application Insights](#). For different environments, collect telemetry data into different Application Insights instances.
  - If your application has multiple components separated into different services, you might want to examine their behavior together. Collect their telemetry data into the same Application Insights instance, but label them with different cloud role names.
  - Export the Application Insights data to an [Azure Log Analytics](#) workspace. We recommend you use a single workspace for the organization.
  - Include operational dashboards in application and feature design, to ensure the solution can be supported in production.
  - Implement health checks for your endpoints, and then use them for health probes, dependency checks, and availability tests.
- Consider using prefixes and suffixes with well-defined conventions, to uniquely identify every deployed resource. These naming conventions avoid conflicts, when you deploy solutions next to each other and improve the overall team agility and throughput.
- Depending on the network configuration, App Service might not be reachable from the public internet, and the use of public hosted agents won't work for deployments. Use [self-hosted agents](#) in that scenario.

## Deploy this scenario

To get started and better understand the specifics of this implementation, review the reference implementation resources, at [User Guide for Reference Implementation Deployment](#).

- We recommend that you clone this repo and modify the reference implementation resources to suit your requirements and your organization's specific landing zone guidelines.

- Before deploying, ensure that the service principal that's used to deploy the solution has the required permissions to create the resource types that we listed above.
- Consider the CI/CD service that you'll use to deploy the reference implementation. As this reference implementation is an internal ASE, you'll need a self-hosted agent to execute the deployment pipelines. You have the choice to use either a DevOps agent or a GitHub runner. Refer to the [user guide](#) on the specific configuration values that are required.
- Consider the region(s) to which you intend to deploy this reference implementation. Consult the [ASEv3 Regions list](#) to ensure the selected region(s) are enabled for deployment.

## Contributors

*This article is maintained by Microsoft. It was originally written by the following contributors.*

Principal authors:

- [Pete Messina](#) | Senior Cloud Solution Architect
- [Nabeel Prior](#) | Senior Cloud Solution Architect

*To see non-public LinkedIn profiles, sign in to LinkedIn.*

## Next steps

- [Security in Azure App Service](#)
- [Networking for App Service](#)
- [Landing zone accelerator for App Service](#)

Learn more about these key services:

- [Azure App Service Environment v3 \(ASEv3\)](#)
- [Azure Private DNS Zones](#)
- [Azure Application Insights](#)
- [Azure Log Analytics](#)
- [Azure Virtual Machines overview](#)
- [Azure Key Vault concepts](#)
- [Azure Bastion](#)

## Related resources

- High availability enterprise deployment using App Services Environment
- Enterprise deployment using App Service Environment
- High availability enterprise deployment using App Service Environment
- E-commerce website running in secured App Service Environment

# Archive on-premises data to the cloud

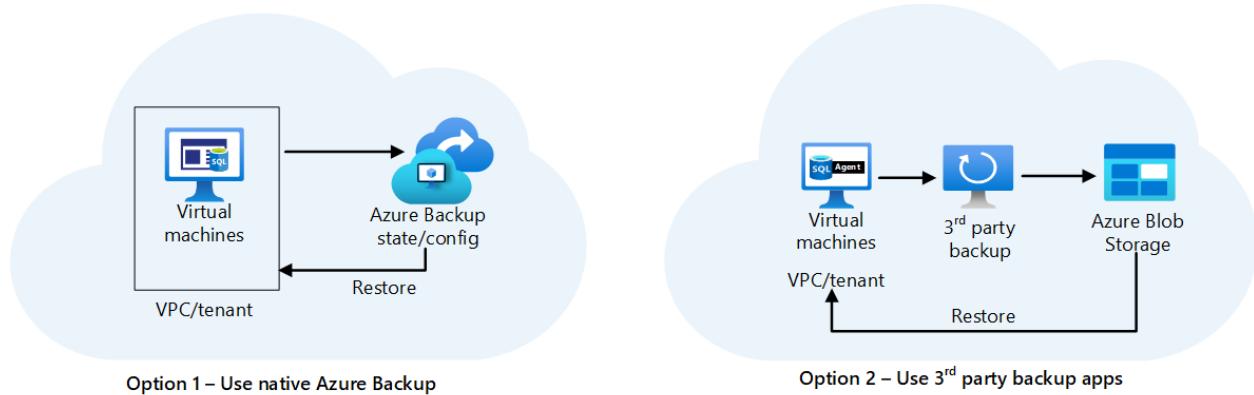
Azure Blob Storage   Azure StorSimple   Azure Virtual Machines

## 💡 Solution ideas

This article is a solution idea. If you'd like us to expand the content with more information, such as potential use cases, alternative services, implementation considerations, or pricing guidance, let us know by providing [GitHub feedback](#).

This architecture shows you how to archive your on-premises data to Azure Blob storage.

## Architecture



*Download an [SVG](#) of this architecture.*

## Components

- Azure [StorSimple](#) appliance running on-premises that can tier data to Azure Blob storage (both hot and cool tier). [StorSimple](#) can be used to archive data from on-premises to Azure.
- [Blob Storage](#): A cool or archive tier on Azure Blob storage is used to back up data that's less frequently accessed, while a hot tier is used to store data that's frequently accessed.

# Alternatives

For new solutions, you might also consider using [Azure File Sync](#). File Sync is a service that allows you to cache several Azure file shares on an on-premises Windows Server or cloud virtual machine (VM).

## Scenario details

This solution is built on the Azure managed services: [StorSimple](#) and [Blob Storage](#). These services run in a high-availability environment, patched and supported, allowing you to focus on your solution instead of the environment they run in.

## Potential use cases

Organizations can leverage Azure Blob storage to:

- Store structured and unstructured logs, images, videos, and other file types.
- Create cost effective solutions for petabytes of data.

## Next steps

- [Learning path for StorSimple](#)
- [Azure Blob Storage: Hot, cool, and Archive storage tiers](#)

# Back up on-premises applications and data to the cloud

Azure Backup   Azure Blob Storage   Azure Virtual Machines

## 💡 Solution ideas

This article is a solution idea. If you'd like us to expand the content with more information, such as potential use cases, alternative services, implementation considerations, or pricing guidance, let us know by providing [GitHub feedback](#).

Back up data and applications from an on-premises system to Azure using Azure Backup or a partner solution. An internet connection to Azure is used to connect to Azure Backup or to Azure Blob storage. Azure Backup Server can write backups directly to Azure Backup. Alternatively, a partner solution such as Commvault Simpana or Veeam Availability Suite, hosted on-premises, can write backups to Blob storage directly or via a cloud endpoint such as Veeam Cloud Connect.

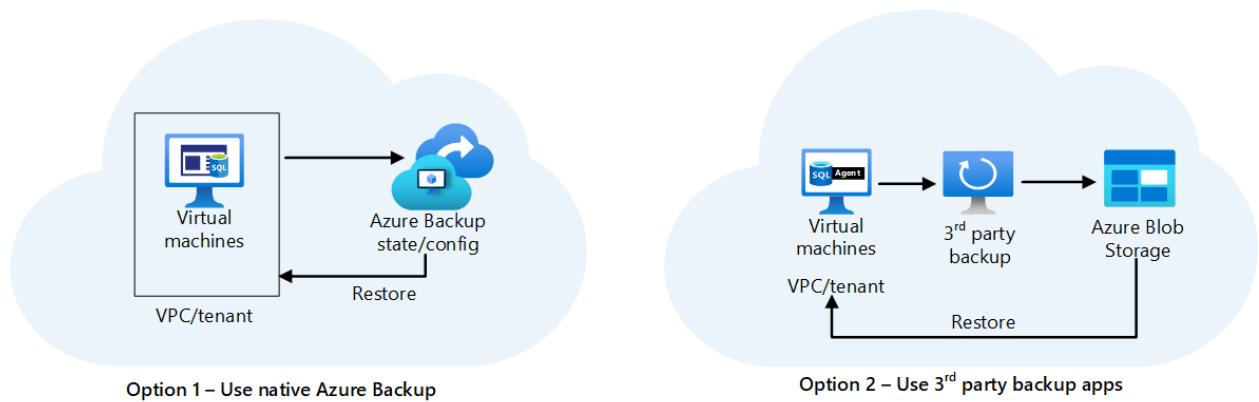
This solution is built on the Azure managed services: [Backup Server](#), [Azure Backup](#), and [Blob Storage](#). These services run in a high-availability environment, patched and supported, allowing you to focus on your solution instead of the environment they run in.

## Potential use case

Your organization's backup strategy will differ depending on the workload you need to protect, and Azure Backup can assist you with a wide variety of backup types.

- Backup and restore your files and folders, which can be great for storing application configuration changes or other business materials.
- “Typical” Windows or Linux machines to fine-grained protection for Exchange, SQL, or SharePoint services.
- Hyper-V, VMware or even capture system state and do a bare-metal recovery if needed.
- Create backups of your Azure VMs directly from the portal.

## Architecture



*Download a [Visio file](#) of this architecture.*

## Components

- Azure [Backup Server](#) orchestrates the backup of machines and manages the configuration of the restore procedures. It also has two days of backup data for operational recovery.
- Azure [Backup](#) service runs on the cloud and holds the recovery points, enforces policies, and enables you to manage data and application protection. You don't need to create or manage an Azure Blob storage account when using [Azure Backup](#).
- [Blob Storage](#): Blob storage that partner solutions such as Commvault connect to for backing up data and applications. You must create and manage Azure Blob storage when using partner solutions.

## Next steps

- [Back up workloads using Azure Backup Server](#)
- [Back up files and folders using Azure Backup](#)
- [Store backed up files in Blob storage](#)

# Centralized app configuration and security

Microsoft Entra ID

Azure App Configuration

Azure Key Vault

## 💡 Solution ideas

This article is a solution idea. If you'd like us to expand the content with more information, such as potential use cases, alternative services, implementation considerations, or pricing guidance, let us know by providing [GitHub feedback](#).

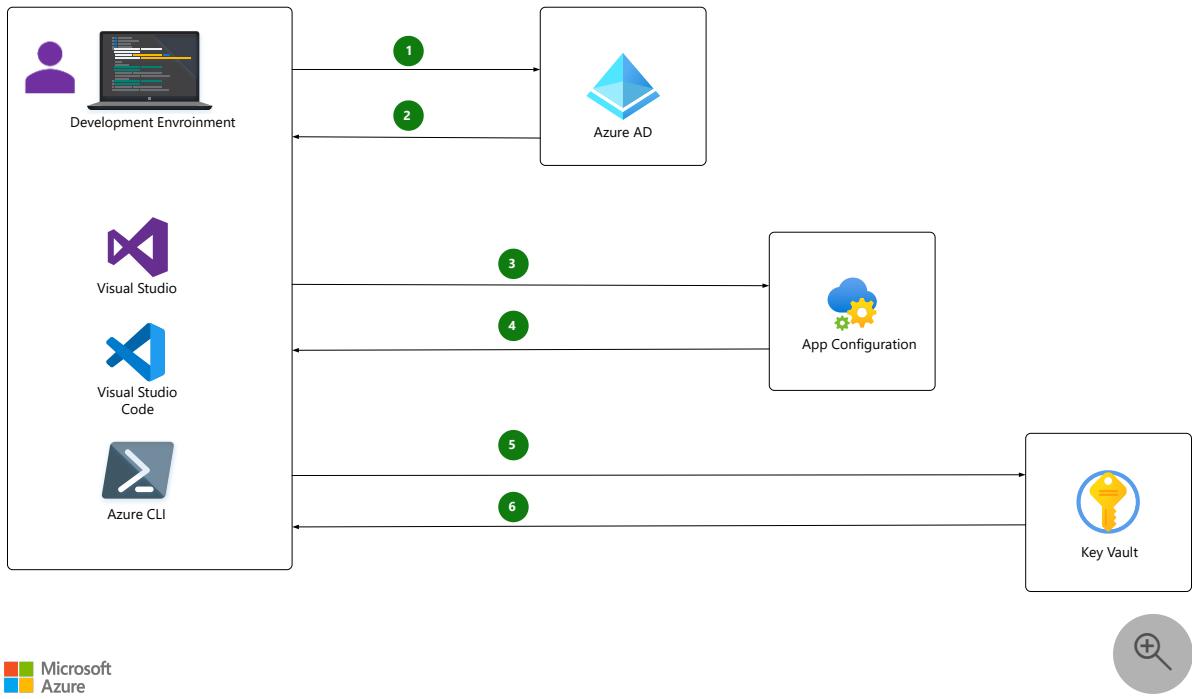
This article outlines a solution for creating a robust and scalable application in a distributed environment. The solution uses Azure App Configuration and Azure Key Vault to manage and store app configuration settings, feature flags, and secure access settings in one place.

## Architecture

The following diagrams show how App Configuration and Key Vault can work together to manage and secure apps in **development** and **Azure** environments.

## Development environment

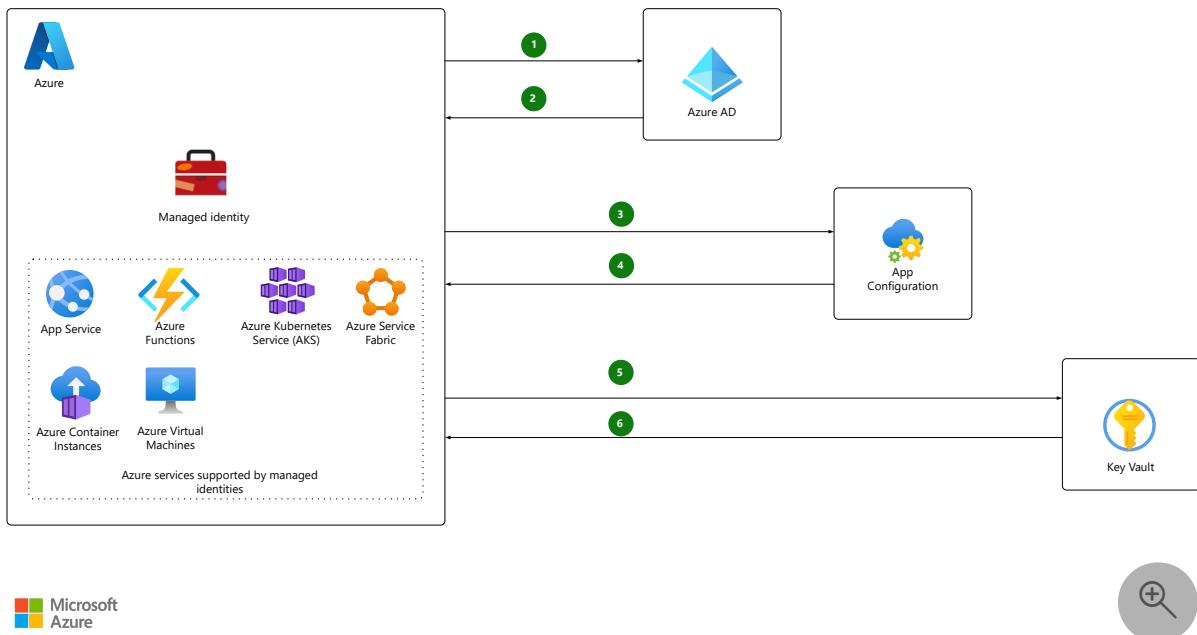
In the development environment, the app uses an identity via Visual Studio or version 2.0 of the Azure CLI to sign in and send an authentication request to Microsoft Entra ID.



[Download a Visio file](#) of this architecture.

## Azure staging or production environment

The Azure staging and production environments use a [managed identity](#) for sign-in and authentication.



[Download a Visio file](#) of this architecture.

## Dataflow

1. The application sends an authentication request during debugging in Visual Studio, or authenticates via the MSI in Azure.
2. Upon successful authentication, Microsoft Entra ID returns an access token.
3. The App Configuration SDK sends a request with the access token to read the app's App Configuration Key Vault **secretURI** value for the app's key vault.
4. Upon successful authorization, App Configuration sends the configuration value.
5. Utilizing the sign-in identity, the app sends a request to Key Vault to retrieve the application secret for the **secretURI** that App Configuration sent.
6. Upon successful authorization, Key Vault returns the secret value.

## Components

- [Microsoft Entra ID](#) is a universal platform for managing and securing identities.
- [App Configuration](#) provides a way to store configurations for all your Azure apps in a universal, hosted location.
- [Managed identities](#) provide an identity for applications to use when connecting to resources that support Microsoft Entra authentication.
- [Key Vault](#) safeguards cryptographic keys and other secrets that are used by cloud apps and services.

## Scenario details

Cloud-based applications often run on multiple virtual machines or containers in multiple regions, and use multiple external services. Creating a robust and scalable application in a distributed environment presents a significant challenge.

By using App Configuration, you can manage and store all your app's configuration settings, feature flags, and secure access settings in one place. App Configuration works seamlessly with Key Vault, which stores passwords, keys, and secrets for secure access.

## Potential use cases

Any application can use App Configuration, but the following types of applications benefit most from it:

- Microservices that are based on Azure Kubernetes Service (AKS), Azure Service Fabric, or other containerized apps that are deployed in one or more regions.
- Serverless apps, which include Azure Functions or other event-driven stateless compute apps.
- Apps that use a continuous deployment (CD) pipeline.

# Considerations

These considerations implement the pillars of the Azure Well-Architected Framework, which is a set of guiding tenets that can be used to improve the quality of a workload. For more information, see [Microsoft Azure Well-Architected Framework](#).

- It's best to use a different key vault for each application in each environment: development, Azure pre-production, and Azure production. Using different vaults helps prevent sharing secrets across environments, and reduces threats in the event of a breach.
- To use these scenarios, the sign-in identity must have the **App Configuration Data Reader** role in the App Configuration resource, and have explicit **access policies** for retrieving the secrets in Key Vault.

## Contributors

*This article is maintained by Microsoft. It was originally written by the following contributors.*

Principal author:

- [Sowmyan Soman](#) | Principal Cloud Solution Architect

## Next steps

Learn more about the component technologies:

- [Azure App Configuration](#)
- [Azure Key Vault](#)
- [Use Key Vault references for App Service and Azure Functions](#)
- [Use managed identities to access App Configuration](#)
- [Local development and security](#)

## Related resources

- [Security architecture design](#)
- [Microservices architecture on Azure Kubernetes Service](#)
- [Microservices architecture on Azure Service Fabric](#)
- [External Configuration Store pattern](#)

# Deploy AKS and API Management with mTLS

Microsoft Entra ID   Azure Kubernetes Service (AKS)   Azure API Management   Azure Container Registry

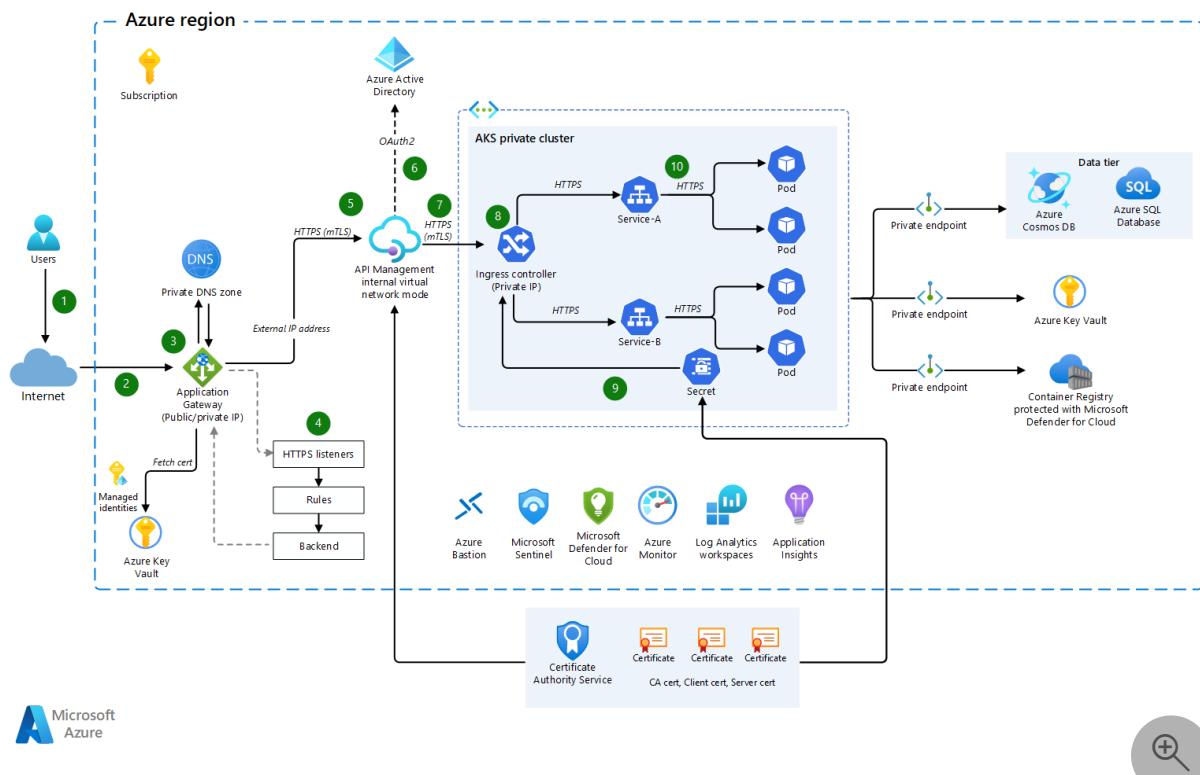
Microsoft Defender for Cloud

## 💡 Solution ideas

This article is a solution idea. If you'd like us to expand the content with more information, such as potential use cases, alternative services, implementation considerations, or pricing guidance, let us know by providing [GitHub feedback](#).

This solution demonstrates how to integrate Azure Kubernetes Service (AKS) and Azure API Management via mutual TLS (mTLS) in an architecture that provides end-to-end encryption.

## Architecture



Download a [Visio file](#) of this architecture.

## Dataflow

1. A user makes a request to the application endpoint from the internet.
2. Azure Application Gateway receives traffic as HTTPS and presents a PFX certificate previously loaded from Azure Key Vault to the user.
3. Application Gateway uses private keys to decrypt traffic (SSL offload), performs web application firewall inspections, and re-encrypts traffic by using public keys (end-to-end encryption).
4. Application Gateway applies rules and backend settings based on the backend pool and sends traffic to the API Management backend pool over HTTPS.
5. API Management is deployed in internal virtual network mode (Developer or Premium tier only) with a private IP address. It receives traffic as HTTPS with custom domain PFX certificates.
6. Microsoft Entra ID provides authentication and applies API Management policies via OAuth and client certificate validation. To receive and verify client certificates over HTTP/2 in API Management, you need to enable **Negotiate client certificate** on the **Custom domains** blade in API Management.
7. API Management sends traffic via HTTPS to an ingress controller for an AKS private cluster.
8. The AKS ingress controller receives the HTTPS traffic and verifies the PEM server certificate and private key. Most enterprise-level ingress controllers support mTLS. Examples include NGINX and AGIC.
9. The ingress controller processes TLS secrets (Kubernetes Secrets) by using cert.pem and key.pem. The ingress controller decrypts traffic by using a private key (offloaded). For enhanced-security secret management that's based on requirements, CSI driver integration with AKS is available.
10. The ingress controller re-encrypts traffic by using private keys and sends traffic over HTTPS to AKS pods. Depending on your requirements, you can configure AKS ingress as HTTPS backend or passthrough.

## Components

- [Application Gateway](#) . Application Gateway is a web traffic load balancer that you can use to manage traffic to web applications.
- [AKS](#) . AKS provides fully managed Kubernetes clusters for deployment, scaling, and management of containerized applications.
- [Azure Container Registry](#) . Container Registry is a managed, private Docker registry service on Azure. You can use Container Registry to store private Docker images, which are deployed to the cluster.

- [Microsoft Entra ID](#). When AKS is integrated with Microsoft Entra ID, you can use Microsoft Entra users, groups, or service principals as subjects in Kubernetes RBAC to manage AKS resources.
  - [Managed identities](#). Microsoft Entra managed identities eliminate the need to manage credentials like certificates, secrets, and keys.
- [Azure SQL Database](#). SQL Database is a fully managed and intelligent relational database service that's built for the cloud. You can use SQL Database to create a high-availability, high-performance data storage layer for your modern cloud applications.
- [Azure Cosmos DB](#). Azure Cosmos DB is a fully managed NoSQL database service for building and modernizing scalable, high-performance applications.
- [API Management](#). You can use API Management to publish APIs to your developers, partners, and employees.
- [Azure Private Link](#). Private Link provides access to PaaS services that are hosted on Azure, so you can keep your data on the Microsoft network.
- [Key Vault](#). Key Vault can provide enhanced security for keys and other secrets.
- [Defender for Cloud](#). Defender for Cloud is a solution for cloud security posture management and cloud workload protection. It finds weak spots across your cloud configuration, helps strengthen the security of your environment, and can protect workloads across multicloud and hybrid environments from evolving threats.
- [Azure Monitor](#). You can use Monitor to collect, analyze, and act on telemetry data from your Azure and on-premises environments. Monitor helps you maximize the performance and availability of your applications and proactively identify problems.
  - [Log Analytics](#). You can use Log Analytics to edit and run log queries with data in Azure Monitor logs.
  - [Application Insights](#). Application Insights is an extension of Azure Monitor. It provides application performance monitoring.
- [Microsoft Sentinel](#). Microsoft Sentinel is a cloud-native security information and event manager platform that uses built-in AI to help you analyze large volumes of data.
- [Azure Bastion](#). Azure Bastion is a fully managed service that provides RDP and SSH access to VMs without any exposure through public IP addresses. You can provision the service directly in your local or peered virtual network to get support for all VMs in that network.
- [Azure Private DNS](#). You can use Private DNS to manage and resolve domain names in a virtual network without adding a custom DNS solution.

## Scenario details

You can use this solution to integrate AKS and API Management via mTLS in an architecture that provides end-to-end encryption.

## Potential use cases

- AKS integration with API Management and Application Gateway, via mTLS.
- End-to-end mTLS between API Management and AKS.
- High security deployments for organizations that need end-to-end TLS. For example, organizations in the financial sector can benefit from this solution.

You can use this approach to manage the following scenarios:

- Deploy API Management in internal mode and expose APIs by using Application Gateway.
- Configure mTLS and end-to-end encryption for high security and traffic over HTTPS.
- Connect to Azure PaaS services by using an enhanced security private endpoint.
- Implement Defender for Containers security.

## Contributors

*This article is maintained by Microsoft. It was originally written by the following contributors.*

Principal author:

- [Saswat Mohanty](#) | Senior Cloud Solution Architect

Other contributors:

- [Mick Alberts](#) | Technical Writer
- [Arshad Azeem](#) | Senior Cloud Solution Architect
- [Raj Penchala](#) | Principal Cloud Solution Architect

*To see non-public LinkedIn profiles, sign in to LinkedIn.*

## Next steps

- [Application Gateway](#)
- [AKS Roadmap](#)
- [AKS documentation](#)
- [AKS learning path](#)

- [API Management learning path](#)
- [API Management landing zone accelerator ↗](#)
- [Microsoft Defender for Cloud Blog ↗](#)

## Related resources

- [AKS architecture design](#)
- [AKS cluster best practices](#)
- [Baseline architecture for an AKS cluster](#)

# Enterprise-scale disaster recovery

[Microsoft Entra ID](#)

[Azure Site Recovery](#)

[Azure Traffic Manager](#)

[Azure Virtual Network](#)

[Azure VPN Gateway](#)

## 💡 Solution ideas

This article is a solution idea. If you'd like us to expand the content with more information, such as potential use cases, alternative services, implementation considerations, or pricing guidance, let us know by providing [GitHub feedback](#).

A large enterprise architecture for SharePoint, Dynamics CRM, and Linux web servers hosted on an on-premises datacenter with failover to Azure infrastructure.

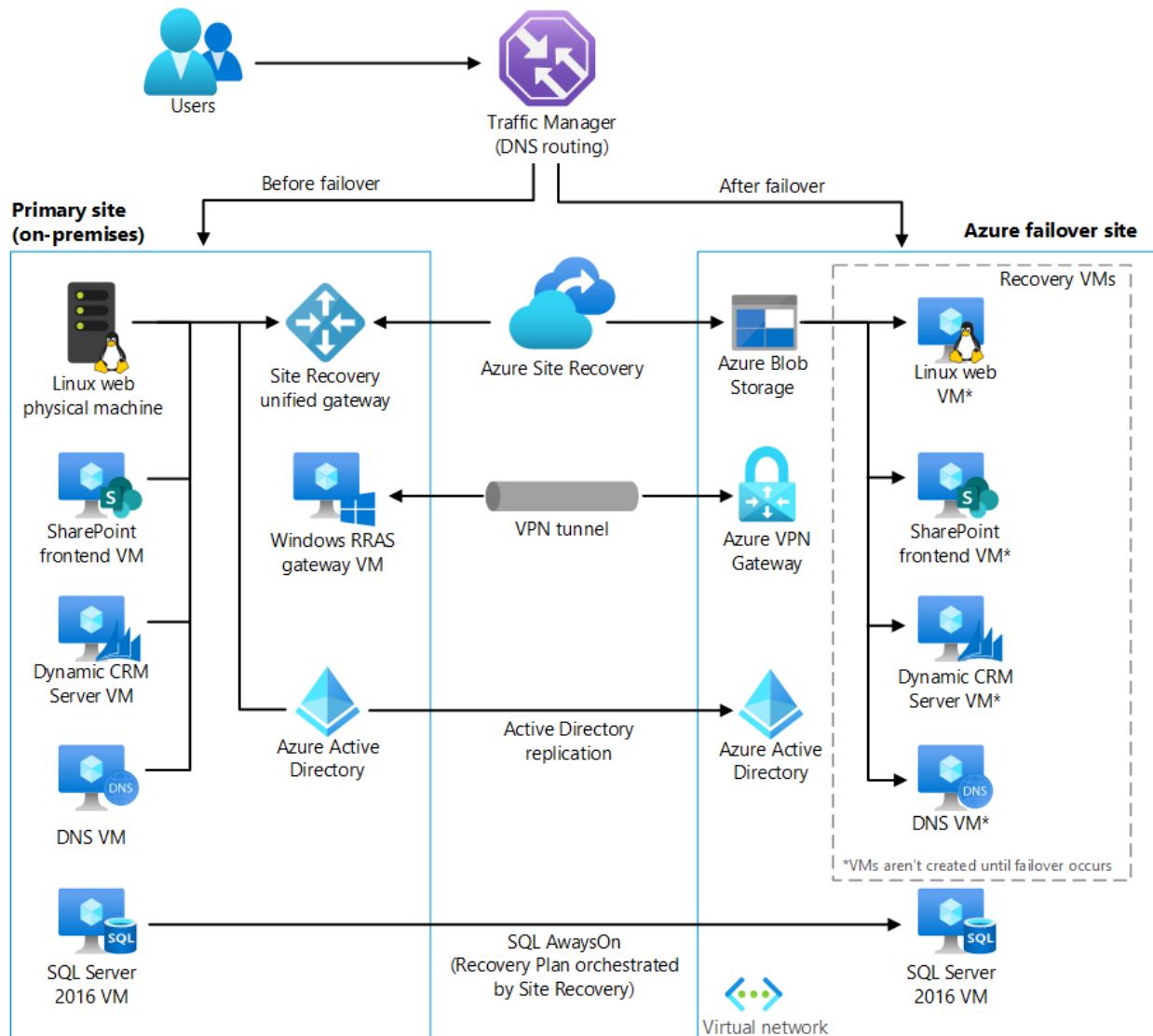
This solution is built on the Azure managed services: [Traffic Manager](#), [Azure Site Recovery](#), [Microsoft Entra ID](#), [VPN Gateway](#), and [Virtual Network](#). These services run in a high-availability environment, patched and supported, allowing you to focus on your solution instead of the environment they run in.

## Potential use cases

Organizations that utilize this service include:

- Hospitals (healthcare)
- Universities (education)
- Government (local, state, and federal)

## Architecture



Download a [Visio file](#) of this architecture.

# Components

- DNS traffic is routed via [Traffic Manager](#), which can easily move traffic from one site to another based on policies defined by your organization.
  - [Azure Site Recovery](#) orchestrates the replication of machines and manages the configuration of the fallback procedures.
  - [Blob storage](#) stores the replica images of all machines that are protected by Site Recovery.
  - [Microsoft Entra ID](#) is the replica of the on-premises [Microsoft Entra ID](#) services allowing cloud applications to be authenticated and authorized by your company.

- [VPN Gateway](#) : The VPN gateway maintains the communication between the on-premises network and the cloud network securely and privately.
- [Virtual Network](#) : The virtual network is where the failover site will be created when a disaster occurs.

## Contributors

*This article is maintained by Microsoft. It was originally written by the following contributors.*

Principal author:

- [Larry Claman](#) | Principal Technology Architect

## Next steps

- [Configure Failover routing method](#)
- [How does Azure Site Recovery work?](#)
- [Introduction to Microsoft Azure Storage](#)
- [Integrating your on-premises identities with Microsoft Entra ID](#)
- [Create a VNet with a Site-to-Site connection using the Azure portal](#)
- [Designing your network infrastructure for disaster recovery](#)

# Build high availability into your BCDR strategy

Azure Load Balancer

Azure SQL Database

Azure Virtual Machines

Azure Virtual Network

Azure App Service

## 💡 Solution ideas

This article is a solution idea. If you'd like us to expand the content with more information, such as potential use cases, alternative services, implementation considerations, or pricing guidance, let us know by providing [GitHub feedback](#).

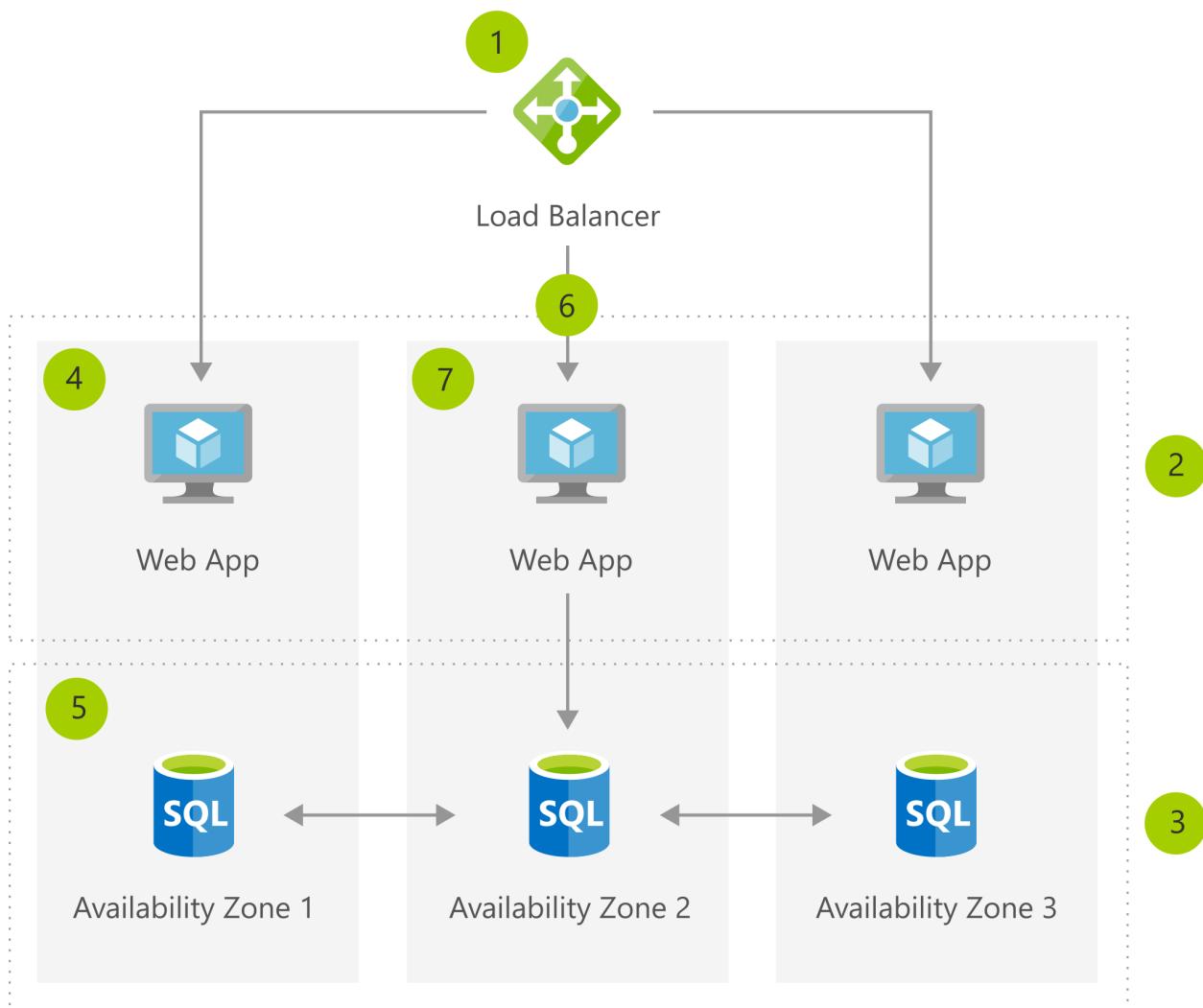
Virtual machines (VMs) are physically separated across zones, and a virtual network is created using load balancers at each site. These locations are close enough for high availability replication, so your applications stay running, despite any issues at the physical locations.

## Potential use cases

This solution is ideal for the healthcare industry and the following scenarios:

- Hospitals
- Data centers

## Architecture



*Download an [SVG](#) of this architecture.*

## Dataflow

1. Create a zone-redundant Load Balancer.
2. Create a front-end subnet.
3. Create a DB subnet.
4. Create VMs in three availability zones.
5. Configure a zone-redundant SQL DB.
6. Add VMs to the load balancer's back-end pool.
7. Deploy your application on VMs, for redundancy and high availability.

## Components

- [Virtual Machines](#): Provision Windows and Linux virtual machines in seconds
- [Azure SQL Database](#): Managed, intelligent SQL in the cloud
- [Load Balancer](#): Deliver high availability and network performance to your applications

# Next steps

- [Virtual Machines documentation](#)
- [SQL Database documentation](#)
- [Load Balancer documentation](#)

# HPC media rendering

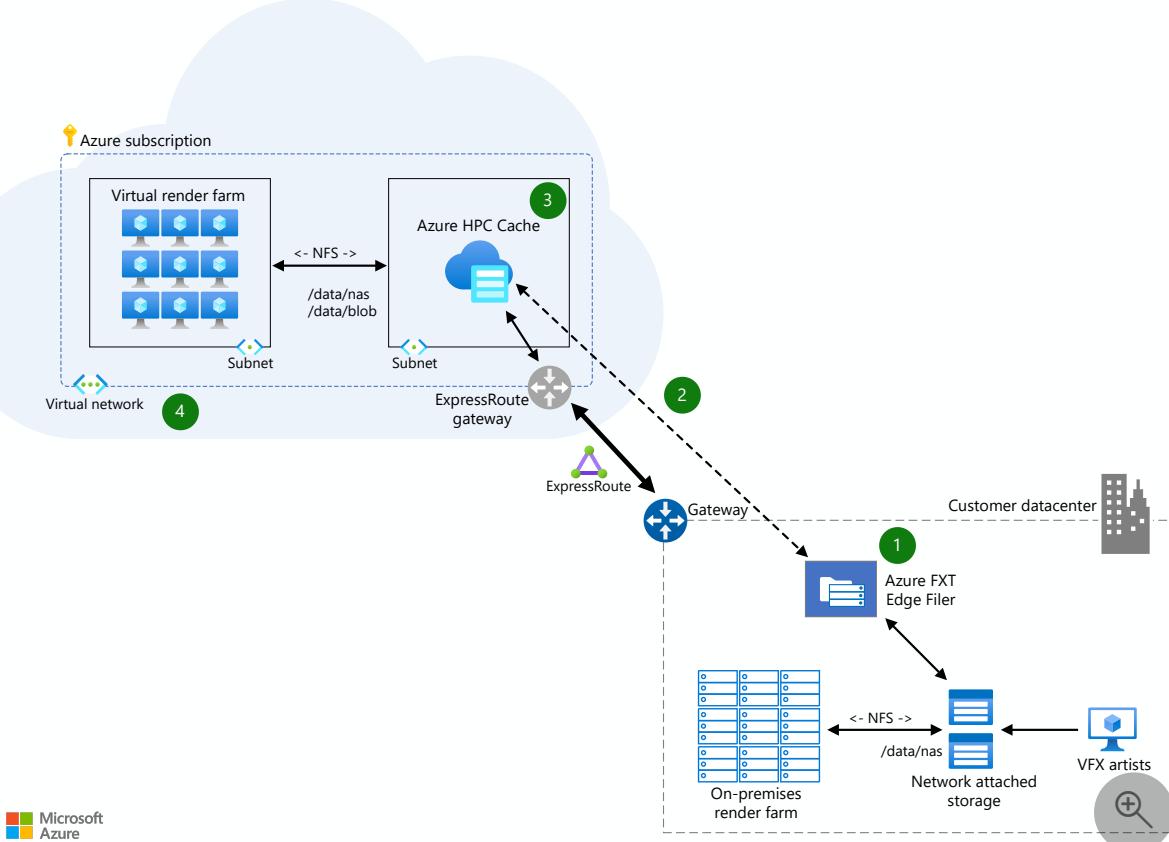
Azure Batch   Azure CycleCloud   Azure FXT Edge Filer   Azure Virtual Machine Scale Sets

## 💡 Solution ideas

This article is a solution idea. If you'd like us to expand the content with more information, such as potential use cases, alternative services, implementation considerations, or pricing guidance, let us know by providing [GitHub feedback](#).

This solution allows studios to leverage on-premises capacity to its fullest with the Azure FXT Edge Filer for NAS acceleration. When demand grows beyond on-premises capacity, burst render provides access to tens of thousands of cores using Azure Virtual Machine Scale Sets. An Express Route connection and HPC Cache minimize latency while studios securely manage storage in a single place without replication.

## Architecture



Download a [Visio file](#) of this architecture.

# Dataflow

1. Optimize access to NAS files and support remote artists with the Azure FXT Edge Filer connecting artists to low-latency storage.
2. Connecting on-premises storage resources to Azure via Azure Express Route providing a secure, private link to additional render cores.
3. Azure HPC Cache provides low-latency access to tens of thousands of compute cores with burst rendering. Azure SDK support in HPC Cache enables automation for easy infrastructure management and cost efficiencies.
4. A virtual render farm is available in Azure using Virtual Machine Scale Sets that grow as you need it and provides capacity to meet fluctuating demand.

## Components

- [N-Series VMs](#) : N-series virtual machines are ideal for compute and graphics-intensive workloads, helping customers to fuel innovation through scenarios like high-end remote visualization, deep learning, and predictive analytics.
- [H-Series VMs](#) : The H-series is a new family specifically designed to handle high performance computing workloads such as financial risk modeling, seismic and reservoir simulation, molecular modeling, and genomic research.
- Effectively manage common workloads with ease while creating and optimizing HPC clusters with Microsoft [Azure CycleCloud](#) .
- [Avere vFXT](#) : Faster, more accessible data storage for high-performance computing at the edge
- [Azure Batch](#) : Cloud-scale job scheduling and compute management

## Scenario details

### Potential use cases

Graphics designers, artists, and animation designers need high performance systems to make sure they deliver the best quality work and can accommodate change requests without waiting hours for the processing to finish. Areas that studios can see the benefits from high performance computing include:

- Animation and modeling.
- 3D Rendering.
- Compositing and color grading.

# Next steps

- [N-Series Virtual Machines Documentation](#)
- [H-Series Virtual Machines Documentation](#)
- [Azure CycleCloud Documentation](#)
- [Avere vFXT Documentation](#)
- [Azure Batch Documentation](#)

# Keyword search and speech-to-text

Azure Content Delivery Network   Azure AI Search   Azure Media Player   Azure Video Indexer

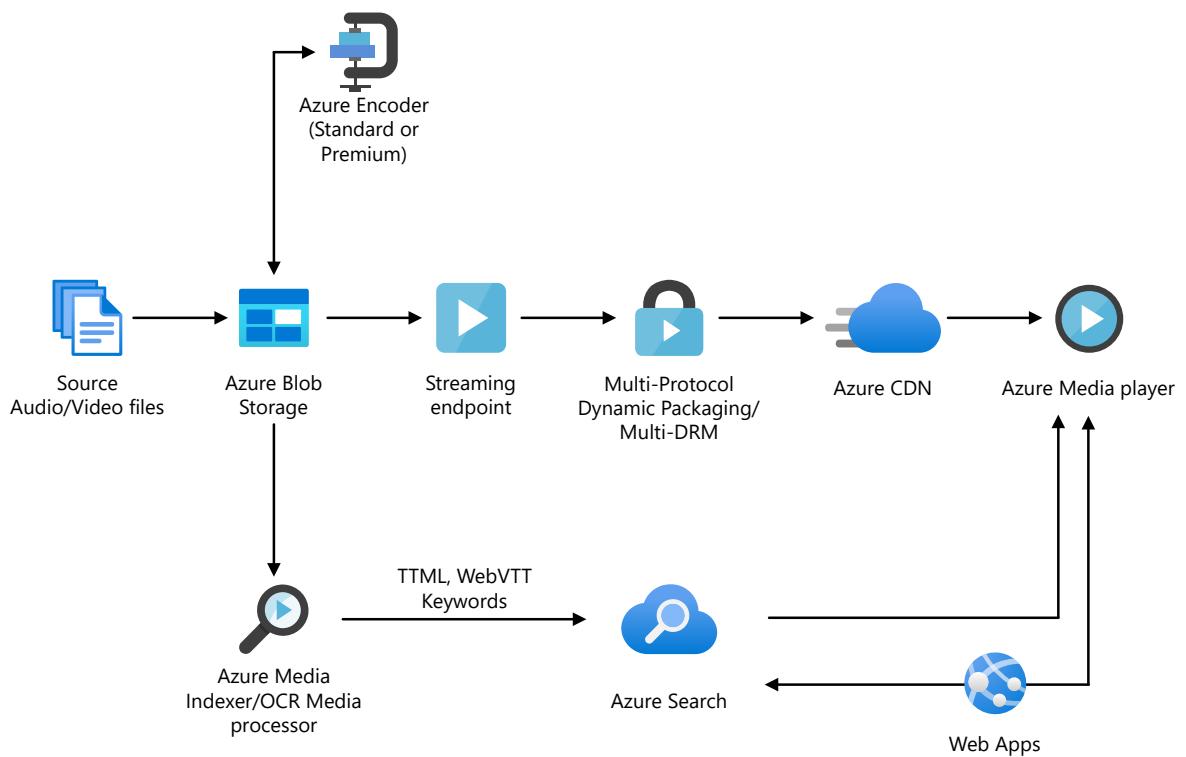
Azure App Service

## 💡 Solution ideas

This article is a solution idea. If you'd like us to expand the content with more information, such as potential use cases, alternative services, implementation considerations, or pricing guidance, let us know by providing [GitHub feedback](#).

This solution idea identifies speech in static video files to manage speech as standard content.

## Architecture



 Microsoft Azure



Download a [Visio file](#) of this architecture.

# Dataflow

- Azure Blob Storage stores large amounts of unstructured data that can be accessed from anywhere in the world via HTTP or HTTPS. You can use [Blob Storage](#) to expose data publicly to the world, or to store application data privately.
- [Azure Encoding](#) converts media files from one encoding to another.
- [Azure streaming endpoint](#) represents a streaming service that can deliver content directly to a client player application, or to a content delivery network (CDN) for further distribution.
- [Content Delivery Network](#) provides secure, reliable content delivery with broad global reach and a rich feature set.
- [Azure Media Player](#) uses industry standards, such as HTML5 (MSE/EME) to provide an enriched adaptive streaming experience. Regardless of the playback technology used, you have a unified JavaScript interface to access APIs.
- [Azure Cognitive Search](#) provides a ready-to-use service that gets populated with data and then used to add search functionality to a web or mobile application.
- [Web Apps](#) hosts the website or web application.
- [Azure Media Indexer](#) makes the content of your media files searchable and generates a full-text transcript for closed-captioning and keywords. Media files are processed individually or in batches.

# Components

- [Blob Storage](#) is a service that's part of [Azure Storage](#). Blob Storage offers optimized cloud object storage for large amounts of unstructured data.
- [Azure Media Services](#) is a cloud-based platform that you can use to stream video, enhance accessibility and distribution, and analyze video content.
- [Live and on-demand streaming](#) is a feature of Azure Media Services that delivers content to various devices at scale.
- [Azure Encoding](#) provides a way to convert files that contain digital video or audio from one standard format to another.
- [Azure Media Player](#) plays videos that are in various formats.
- [Azure Content Delivery Network](#) offers a global solution for rapidly delivering content. This service provides your users with fast, reliable, and secure access to your apps' static and dynamic web content.
- [Azure Cognitive Search](#) is a cloud search service that supplies infrastructure, APIs, and tools for searching. You can use Azure Cognitive Search to build search experiences over private, heterogeneous content in web, mobile, and enterprise applications.

- [App Service](#) provides a framework for building, deploying, and scaling web apps. The [Web Apps](#) feature is a service for hosting web applications, REST APIs, and mobile back ends.
- [Azure Media Indexer](#) provides a way to make content of your media files searchable. It can also generate a full-text transcript for closed captioning and keywords.

## Scenario details

A speech-to-text solution provides a way to identify speech in static video files so you can manage it as standard content. For instance, employees can use this technology to search within training videos for spoken words or phrases. Then they can navigate to the specific moment in the video that contains the word or phrase.

When you use this solution, you can upload static videos to an Azure website. The Azure Media Indexer uses the Speech API to index the speech within the videos and stores it in an Azure database. You can search for words or phrases by using the Web Apps feature of Azure App Service. Then you can retrieve a list of results. When you select a result, you can see the place in the video that mentions the word or phrase.

This solution is built on the Azure managed services [Content Delivery Network](#) and [Azure Cognitive Search](#).

## Potential use cases

This solution applies to scenarios that can benefit from the ability to search recorded speech. Examples include:

- Training and educational videos.
- Crime investigations.
- Customer service analysis.

## Next steps

- [How to use Azure Blob Storage](#)
- [How to encode an asset using Media Encoder](#)
- [How to manage streaming endpoints](#)
- [Using Azure Content Delivery Network](#)
- [Develop video player applications](#)
- [Create an Azure Cognitive Search service](#)
- [Run Web Apps in the cloud](#)

- Indexing media files

## Related resources

- Gridwich cloud media system
- Live stream digital media
- Video-on-demand digital media

# SMB disaster recovery with Azure Site Recovery

Azure Blob Storage

Azure Site Recovery

Azure Traffic Manager

Azure Virtual Network

SQL Server

## 💡 Solution ideas

This article is a solution idea. If you'd like us to expand the content with more information, such as potential use cases, alternative services, implementation considerations, or pricing guidance, let us know by providing [GitHub feedback](#).

Small and medium businesses can inexpensively implement disaster recovery to the cloud by using Azure Site Recovery or a partner solution like Double-Take DR.

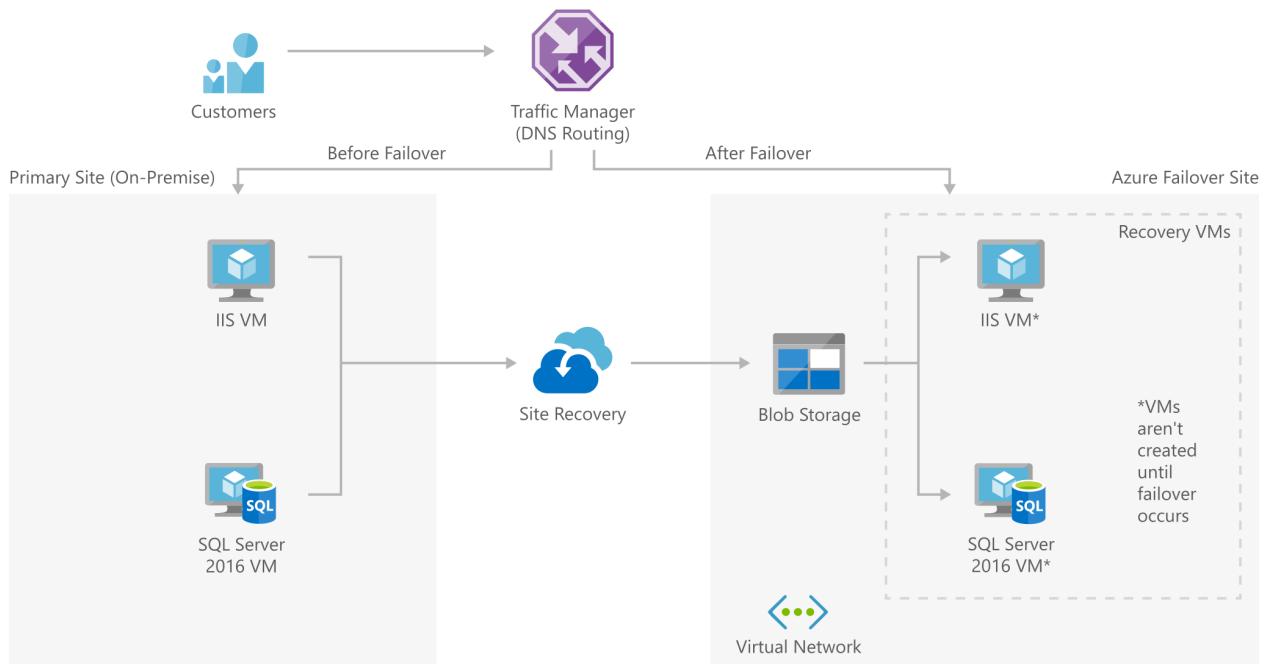
This solution is built on the Azure managed services: [Traffic Manager](#), [Azure Site Recovery](#), and [Virtual Network](#). These services run in a high-availability environment, patched and supported, allowing you to focus on your solution instead of the environment they run in.

## Potential use cases

Ideal industries for this solution include healthcare, travel and hospitality, and manufacturing. Industries that utilize this service include:

- Healthcare (portable clinics and pop-up virus testing centers)
- Restaurants (local and regional chains, in the travel and hospitality industries)
- Logistics (local and regional supply chains, in the manufacturing industry)

## Architecture



[Download an \*\*SVG\*\* of this architecture.](#)

## Components

- DNS traffic is routed via [Traffic Manager](#), which can easily move traffic from one site to another based on policies defined by your organization.
- [Azure Site Recovery](#) orchestrates the replication of machines and manages the configuration of the failback procedures.
- [Virtual Network](#): The virtual network is where the failover site will be created when a disaster occurs.
- [Blob storage](#) stores the replica images of all machines that are protected by Site Recovery.

## Next steps

- [Configure Failover routing method](#)
- [How does Azure Site Recovery work?](#)
- [Designing your network infrastructure for disaster recovery](#)
- [Introduction to Microsoft Azure Storage](#)

# SMB disaster recovery with Double-Take DR

Azure Traffic Manager

Azure Virtual Network

Azure VPN Gateway

SQL Server

## 💡 Solution ideas

This article is a solution idea. If you'd like us to expand the content with more information, such as potential use cases, alternative services, implementation considerations, or pricing guidance, let us know by providing [GitHub feedback](#).

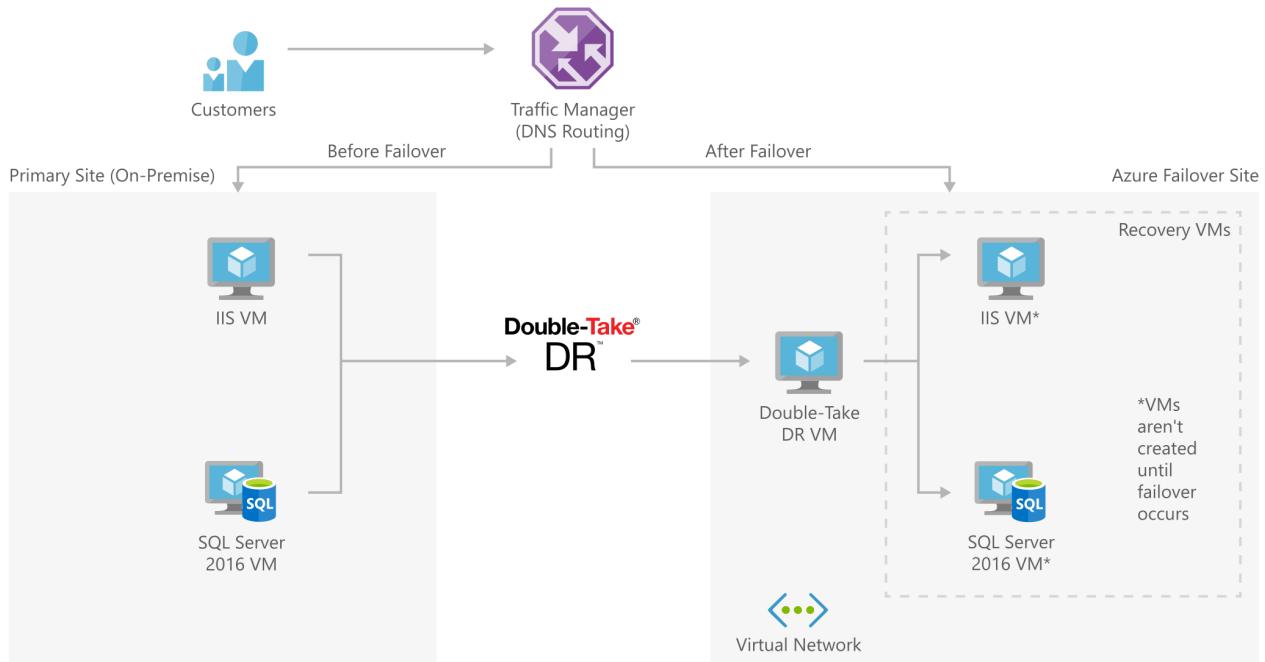
Small and medium businesses can inexpensively implement disaster recovery to the cloud by using a partner solution like Double-Take DR.

This solution is built on the Azure managed services: [Traffic Manager](#), [VPN Gateway](#), and [Virtual Network](#). These services run in a high-availability environment, patched and supported, allowing you to focus on your solution instead of the environment they run in.

## Potential use cases

Organizations that utilize double-take include SMEs to Fortune 500 companies.

## Architecture



[Download an \*\*SVG\*\* of this architecture.](#)

## Components

- DNS traffic is routed via [Traffic Manager](#), which can easily move traffic from one site to another based on policies defined by your organization.
- [VPN Gateway](#): The VPN gateway maintains the communication between the on-premises network and the cloud network securely and privately.
- [Virtual Network](#): The virtual network is where the failover site will be created when a disaster occurs.

## Next steps

- Configure Failover routing method
- Create a VNet with a Site-to-Site connection using the Azure portal

# Training and procedural guidance powered by mixed reality

Azure Media Services

Azure Spatial Anchors

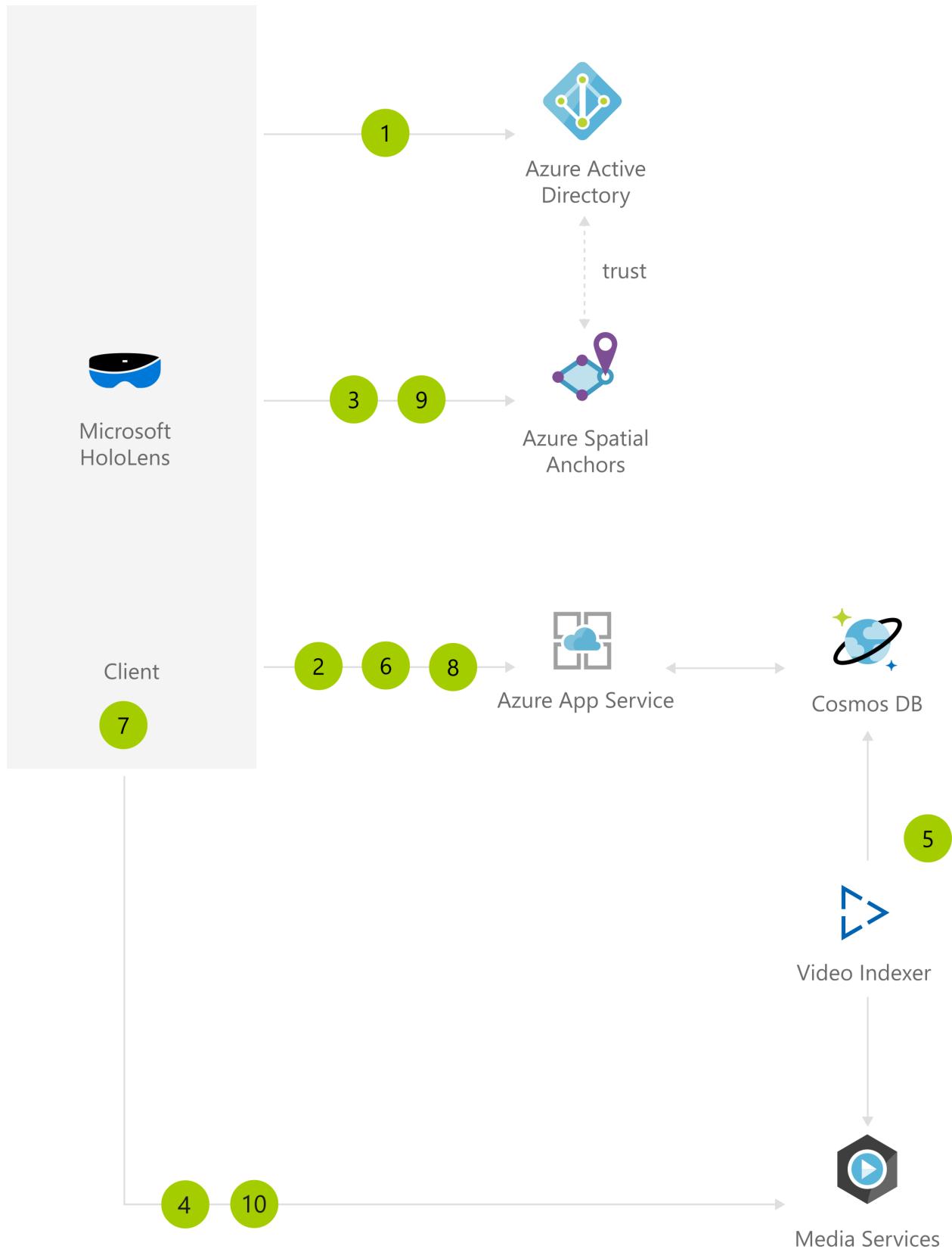
Azure Video Indexer

## 💡 Solution ideas

This article is a solution idea. If you'd like us to expand the content with more information, such as potential use cases, alternative services, implementation considerations, or pricing guidance, let us know by providing [GitHub feedback](#).

Organizations can use this solution to help employees learn processes and materials effectively by providing persistent holographic instructions mapped to precise locations in physical workspaces.

## Architecture



Download an [SVG file](#) of this architecture.

## Dataflow

1. The user creating the training session authenticates using their Microsoft Entra credentials from HoloLens.
2. The client application connects to its own web service to create a training session. Metadata about that training session is stored in Azure Cosmos DB.

3. The user scans the environment and places a first anchor where the first step of the procedure needs to happen. Azure Spatial Anchors validates that the user has sufficient permissions to create anchors via Microsoft Entra ID, and then stores the anchor.
4. The user records a video of the procedure on HoloLens and uploads it to Azure.
5. The video is encoded with Media Services and prepared for on-demand viewing, as well as processed with Video Indexer for better content search. Video Indexer stores the metadata on Azure Cosmos DB.
6. The app saves against its web service the anchor ID for that first step, alongside a link to the video.
7. The user, in the same session, then moves on to step 2, places an anchor there, and again records a video of the procedure and saves the resulting anchor ID and video link to its web service. That process is then repeated until all steps in the procedure are executed. As the user moves from step to step, previous anchors are still visible with their respective step number.
8. A trainee comes in, selects the training session, retrieves anchor IDs and links to videos that are part of the procedure.
9. The trainee scans the room to find the anchors indicating the real-world location of each step in the procedure. As soon as one is found, all anchors are retrieved and shown in the app.
10. The trainee can then retrace the exact steps of the expert who recorded the procedure, and view holographic videos of each step at the right location in the lab.

## Components

- [Spatial Anchors](#) : Create multi-user, spatially aware mixed reality experiences
- [Microsoft Entra ID](#) : Synchronize on-premises directories and enable single sign-on
- [Azure Cosmos DB](#) : Globally distributed, multi-model database for any scale
- [App Service](#) : Quickly create powerful cloud apps for web and mobile
- [Media Services](#) : Encode, store, and stream video and audio at scale
- [Video Indexer](#) : Make your media more discoverable and accessible

## Scenario details

This solution can help you enable your team and employees to learn new processes and materials faster, with fewer errors and greater confidence, by providing persistent holographic instructions mapped to precise locations in their physical workspace. Jumpstart employee comprehension with head-up, hands-free experiences using

HoloLens devices. And with Azure Spatial Anchors, you can place directions on the procedure's most important objects and return to this content over time.

## Potential use cases

Organizations use this solution to enable employees (new and current) with learning new skills, material and processes quicker than they can with traditional training programs. This solution is ideal for the education industry.

## Next steps

- Share Spatial Anchors across devices
- Create a new tenant in Microsoft Entra ID
- Build a .NET web app using Azure Cosmos DB for NoSQL and the Azure portal
- Authenticate and authorize users end-to-end in Azure App Service
- Upload, encode, and stream videos using .NET
- What is Video Indexer?

## Related resources

- [Solutions for the education industry](#)
- [Immersive education](#)
- [Training and simulation for enterprises](#)

# Video capture and analytics for retail

Azure IoT Edge

Azure IoT Hub

Azure Media Services

Azure Stack Edge

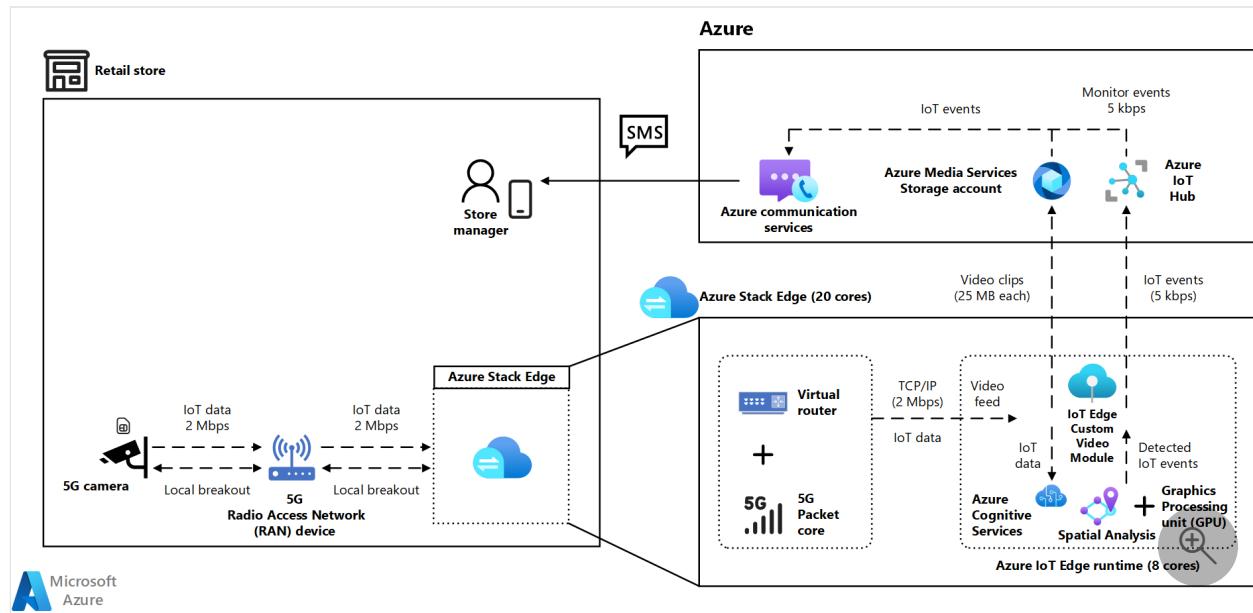
Azure App Service

## 💡 Solution ideas

This article is a solution idea. If you'd like us to expand the content with more information, such as potential use cases, alternative services, implementation considerations, or pricing guidance, let us know by providing [GitHub feedback](#).

This solution describes how retailers like grocery stores can monitor storefront events and take immediate actions to improve customer experience. In this solution, 5G-enabled internet protocol (IP) cameras capture real-time video of shelf inventory, curbside pickup, and cashier queues.

## Architecture



Download a [Visio file](#) of this architecture.

## Dataflow

1. 5G-enabled IP cameras capture video in real time, and send the video feed to a 5G Radio Access Network (RAN) device.

2. The 5G radios in the stores forward the data to the 5G packet core running on the Azure Stack Edge IoT Edge server.
3. The packet core authenticates the devices, applies Quality of Service (QoS) policies, and routes the video traffic to the target application.
4. The custom IoT Edge module also runs on the edge server, which provides the low latency necessary for transporting and processing the video feeds.
5. The custom module simplifies setting up a video streaming pipeline and pre-processing the video for spatial analysis.
6. The [Spatial Analysis](#) module on the edge server anonymously counts cars, goods on a shelf, or people in line. The module sends these event notifications to the Azure IoT Hub module in the cloud.
7. The IoT Hub module records the event notifications in a web app, and alerts store managers or stock keepers if certain thresholds are passed.
8. An Azure Media Services Storage account stores events for long-term trend analysis to help with resource planning.

## Components

This solution uses the following Azure components:

- [Azure Stack Edge](#) is a portfolio of devices that bring compute, storage, and intelligence to the IoT Edge. Azure Stack Edge acts as a cloud storage gateway that enables data transfers to Azure, while retaining local access to files.
- [Web Apps in Microsoft Azure App Service](#) creates and deploys mission-critical web applications that scale with your business.
- [Azure IoT Hub](#) is a cloud-based managed service for bidirectional communication between IoT devices and Azure.
- [Media Services Storage](#) uses Azure Storage to store large media files.
- [Azure Network Function Manager](#) enables the deployment of network functions to the IoT Edge using consistent Azure tools and interfaces.

## Scenario details

This solution describes how retailers like grocery stores can monitor storefront events and take immediate actions to improve customer experience. In this solution, 5G-enabled internet protocol (IP) cameras capture real-time video of shelf inventory, curbside pickup, and cashier queues. On-premises IoT Edge devices analyze the video

data in real time to detect the number of people in checkout queues, empty shelf space, or cars in the parking lot.

Metrics analysis can trigger anomaly events to alert the store manager or stock supervisors to take corrective actions. The solution stores summary video clips or events in the cloud for long-term trend analysis.

## Potential use cases

This solution is ideal for the retail, automotive, and facilities/real-estate industries. This approach includes the following scenarios:

- Monitor and maintain occupancy limits in an establishment.
- Stop unauthorized users from tailgating others into an office building.
- Prevent fraud at grocery store self-checkout stations.

## Contributors

*This article is maintained by Microsoft. It was originally written by the following contributors.*

Principal author:

- [Nikhil Ravi](#) | Product Management Leader

## Next steps

- [What is the Radio Access Network?](#)
- [Live Video Analytics on IoT Edge](#)
- [Azure Network Function Manager simplifies 5G deployments \(Video\)](#)
- [Introduction to Azure IoT Hub](#)
- [Introduction to Azure Stack](#)

## Related resources

- [IoT event routing](#)
- [Contactless IoT interfaces with Azure intelligent edge](#)

# Media architecture design

Article • 06/12/2023

Azure and Azure services can help you deliver high-quality video content to any device. Azure Media Services is a cloud-based platform that enables you to build solutions that achieve broadcast-quality video streaming, enhance accessibility and distribution, analyze content, and more. Gridwich, an event-processing framework for delivering media assets on Azure, was developed by Microsoft for a well-known entertainment conglomerate.

Select the following links to learn more about media services that are available on Azure:

- [Azure Media Services](#). Use high-definition video encoding and streaming services to reach your audiences on the devices they use. And enhance content discoverability and performance by using AI.
- [Azure Media Player](#). Use a single player for automatic playback on the most popular devices.
- [Azure Content Delivery Network](#). Deliver content on a fast, reliable network that has global reach.
- [Azure Content Protection](#). Use AES, PlayReady, Widevine, and FairPlay to deliver content with enhanced security.
- [Encoding](#). Implement studio-grade encoding at cloud scale.
- [Live and on-demand streaming](#). Deliver content to virtually any device, with scalable streaming.

## Path to production

For a brief overview, start with [Media Services terminology and concepts](#).

## Best practices

The Azure Security Benchmark provides recommendations on how you can improve the security of your Azure solutions. For information that's specific to Media Services, see [Azure security baseline for Azure Media Services](#).

## Media architectures and solutions

The following sections, organized by category, provide links to sample architectures and other articles.

# Gridwich media processing system

Gridwich is a stateless event-processing framework created by Microsoft. It embodies best practices for processing and delivering media assets on Azure. Gridwich pipelines ingest, process, store, and deliver media assets.

For more information, see the [Gridwich cloud media system](#) architecture.

## Gridwich concepts

- Clean monolith architecture
- Saga orchestration
- Project naming and namespaces
- CI/CD pipeline
- Content protection and DRM
- Media Services setup and scaling
- Gridwich Azure Storage Service
- Logging
- Gridwich request-response messages
- Variable flow with Terraform and Azure Pipelines

## Gridwich procedures

- Set up Azure DevOps
- Run Azure admin scripts
- Set up a local development environment
- Create a cloud environment
- Maintain and rotate keys
- Test Media Services v3 encoding

## Live streaming

- Live streaming with Azure Media Services v3
- Instant broadcasting with serverless code
- Live stream digital media

## Video on demand

- High availability with Media Services and video on demand
- Video-on-demand digital media

# Stay current with media workloads on Azure

Get the latest updates on [Azure media services and features](#).

## Additional resources

### Example solutions

Here are some additional articles about Azure media solutions:

- [Encoding video and audio with Media Services](#)
- [Monitor Media Services](#)
- [Content protection with dynamic encryption and key delivery](#)

### AWS professionals

- [AWS to Azure services comparison - Media Services](#)

# Media Services terminology and concepts

Article • 01/10/2023



Media Services API v3

## ⚠️ Warning

Azure Media Services will be retired June 30th, 2024. For more information, see the [AMS Retirement Guide](#).

This topic gives a brief overview of Azure Media Services terminology and concepts. The article also provides links to articles with an in-depth explanation of Media Services v3 concepts and functionality.

The fundamental concepts described in these topics should be reviewed before starting development.

## Media Services v3 terminology

Term	Description
Live Event	A <b>Live Event</b> represents a pipeline for ingesting, transcoding (optionally), and packaging live streams of video, audio, and real-time metadata.  For customers migrating from Media Services v2 APIs, the <b>Live Event</b> replaces the <b>Channel</b> entity in v2. For more information, see <a href="#">Migrating from v2 to v3</a> .
Streaming Endpoint/Packaging/Origin	A <b>Streaming Endpoint</b> represents a dynamic (just-in-time) packaging and origin service that can deliver your live and on-demand content directly to a client player application. It uses one of the common streaming media protocols (HLS or DASH). In addition, the <b>Streaming Endpoint</b> provides dynamic (just-in-time) encryption to industry-leading digital rights management systems (DRMs).  In the media streaming industry, this service is commonly referred to as a <b>Packager</b> or <b>Origin</b> . Other common terms in the industry for this capability include JITP (just-in-time-packager) or JITE (just-in-time-encryption).

# Media Services v3 concepts

Concepts	Description	Links
Assets and uploading content	<p>To start managing, encrypting, encoding, analyzing, and streaming media content in Azure, you need to create a Media Services account and upload your digital files into <b>Assets</b>.</p>	<a href="#">Cloud upload and storage</a>  <a href="#">Assets concept</a>
Encoding content	<p>Once you upload your high-quality digital media files into Assets, you can encode them into formats that can be played on a wide variety of browsers and devices.</p> <p>To encode with Media Services v3, you need to create <b>Transforms</b> and <b>Jobs</b>.</p>	<a href="#">Transforms and Jobs</a>  <a href="#">Encoding with Media Services</a>
Packaging and delivery	<p>Once your content is encoded, you can take advantage of <b>Dynamic Packaging</b>. In Media Services, a <b>Streaming Endpoint</b> is the dynamic packaging service used to deliver media content to client players. To make videos in the output asset available to clients for playback, you have to create a <b>Streaming Locator</b> and then build streaming URLs.</p> <p>When creating the <b>Streaming Locator</b>, in addition to the asset's name, you need to specify <b>Streaming Policy</b>. <b>Streaming Policies</b> enable you to define streaming protocols and encryption options (if any) for your <b>Streaming Locators</b>. Dynamic Packaging is used whether you stream your content live or on-demand.</p> <p>You can use Media Services <b>Dynamic Manifests</b> to stream only a specific rendition or subclips of your video. In addition, if you have pre-encoded content, or content that is already encoded by a 3rd party encoder you can stream the content with the AMS origin services. For an example of using a pre-encoded source file, see the sample - <a href="#">Streaming an existing Mp4</a></p>	<a href="#">Dynamic packaging</a>  <a href="#">Streaming Endpoints</a>  <a href="#">Streaming Locators</a>  <a href="#">Streaming Policies</a>  <a href="#">Dynamic manifests</a>  <a href="#">Filters</a>

Concepts	Description	Links
Content protection	<p>With Media Services, you can deliver your live and on-demand content encrypted dynamically with Advanced Encryption Standard (AES-128) or/and any of the three major DRM systems: Microsoft PlayReady, Google Widevine, and Apple FairPlay. Media Services also provides a service for delivering AES keys and DRM (PlayReady, Widevine, and Apple FairPlay Streaming) licenses to authorized clients.</p>	<a href="#">Content Key Policies</a> <a href="#">Content protection</a>
	<p>If specifying encryption options on your stream, create the <b>Content Key Policy</b> and associate it with your <b>Streaming Locator</b>. The <b>Content Key Policy</b> enables you to configure how the content key is delivered to end clients.</p>	
	<p>Try to reuse policies whenever the same options are needed.</p>	
Live streaming	<p>Media Services enables you to deliver live events to your customers on the Azure cloud. <b>Live Events</b> are responsible for ingesting and processing the live video feeds. When you create a <b>Live Event</b>, an input endpoint is created that you can use to send a live signal from a remote encoder. Once you have the stream flowing into the <b>Live Event</b>, you can begin the streaming event by creating an <b>Asset</b>, <b>Live Output</b>, and <b>Streaming Locator</b>. <b>Live Output</b> will archive the stream into the <b>Asset</b> and make it available to viewers through the <b>Streaming Endpoint</b>. A live event can be set to either a <i>pass-through</i> (an on-premises live encoder sends a multiple bitrate stream) or <i>live encoding</i> (an on-premises live encoder sends a single bitrate stream).</p>	<a href="#">Live streaming overview</a> <a href="#">Live Events and Live Outputs</a>
Monitoring with Event Grid	<p>To see the progress of the job, use <b>Event Grid</b>. Media Services also emits the live event types. With Event Grid, your apps can listen for and react to events from virtually all Azure services, as well as custom sources.</p>	<a href="#">Handling Event Grid events</a> <a href="#">Schemas</a>
Monitoring with Azure Monitor	<p>Monitor metrics and diagnostic logs that help you understand how your apps are performing with Azure Monitor.</p>	<a href="#">Media Services monitoring</a>
Player clients	<p>You can use any player framework that supports the HLS or DASH streaming protocol. There are many open source and commercial players available on the market (Shaka, Hls.js, Video.js, Theo Player, Bitmovin Player, etc.) as well as built-in native browser and OS level streaming support for HLS and DASH. The Azure Media Player is also available to play back media content streamed by Media Services on a wide variety of browsers. The Azure Media Player uses industry standards, such as HTML5, Media Source Extensions (MSE), and Encrypted Media Extensions (EME) to provide an adaptive streaming experience.</p>	<a href="#">List of media players</a>

# Get help and support

You can contact Media Services with questions or follow our updates by one of the following methods:

- [Q & A](#)
- [Stack Overflow](#). Tag questions with `azure-media-services`.
- [@MSFTAzureMedia](#) or use [@AzureSupport](#) to request support.
- Open a support ticket through the Azure portal.

# Encoding video and audio with Media Services

Article • 02/02/2023



Media Services API v3

## ⚠️ Warning

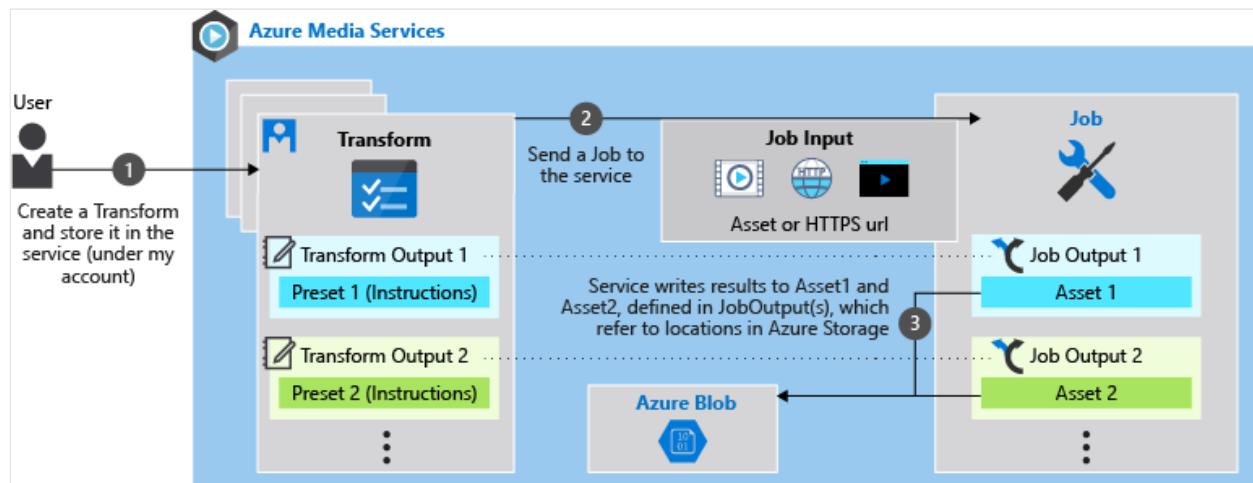
Azure Media Services will be retired June 30th, 2024. For more information, see the [AMS Retirement Guide](#).

## 💡 Tip

Want to generate thumbnails, stitch two videos together, subclip a video or rotate it (among other things)? You can find Media Services sample code on the [Samples](#) page.

The term encoding in Media Services applies to the process of converting files containing digital video and/or audio from one standard format to another, with the purpose of (a) reducing the size of the files, and/or (b) producing a format that's compatible with a broad range of devices and apps. This process is also referred to as video compression, or transcoding. See the [Data compression](#) and the [What Is Encoding and Transcoding?](#) for further discussion of the concepts.

Videos are typically delivered to devices and apps by [progressive download](#) or through [adaptive bitrate streaming](#).



## ⓘ Important

Media Services does not bill for canceled or jobs that throw errors. For example, a job that has reached 50% progress and is canceled is not billed at 50% of the job minutes. You are only charged for finished jobs.

- To deliver by progressive download, you can use Azure Media Services to convert a digital media file (mezzanine) into an [MP4](#) file, which contains video that's been encoded with the [H.264](#) codec, and audio that's been encoded with the [AAC](#) codec. This MP4 file is written to an Asset in your storage account. You can use the Azure Storage APIs or SDKs (for example, [Storage REST API](#) or [.NET SDK](#)) to download the file directly. If you created the output Asset with a specific container name in storage, use that location. Otherwise, you can use Media Services to [list the asset container URLs](#).
- To prepare content for delivery by adaptive bitrate streaming, the mezzanine file needs to be encoded at multiple bitrates (high to low). To ensure graceful transition of quality, the resolution of the video is lowered as the bitrate is lowered. This results in a so-called encoding ladder—a table of resolutions and bitrates (see [auto-generated adaptive bitrate ladder](#) or use the content aware encoding preset). You can use Media Services to encode your mezzanine files at multiple bitrates. In doing so, you'll get a set of MP4 files and associated streaming configuration files written to an Asset in your storage account. You can then use the [Dynamic Packaging](#) capability in Media Services to deliver the video via streaming protocols like [MPEG-DASH](#) and [HLS](#). This requires you to create a [Streaming Locator](#) and build streaming URLs corresponding to the supported protocols, which can then be handed off to devices/apps based on their capabilities.

## Transforms and jobs

To encode with Media Services v3, you need to create a [Transform](#) and a [Job](#). The transform defines a recipe for your encoding settings and outputs; the job is an instance of the recipe. For more information, see [Transforms and Jobs](#).

When encoding with Media Services, you use presets to tell the encoder how the input media files should be processed. In Media Services v3, you use Standard Encoder to encode your files. For example, you can specify the video resolution and/or the number of audio channels you want in the encoded content.

You can get started quickly with one of the built-in presets based on industry best practices or you can choose to build a custom preset to target your specific scenario or device requirements.

Starting with January 2019, when encoding with the Standard Encoder to produce MP4 file(s), a new .mpi file is generated and added to the output Asset. This MPI file is intended to improve performance for [dynamic packaging](#) and streaming scenarios.

### Note

You shouldn't modify or remove the MPI file, or take any dependency in your service on the existence (or not) of such a file.

## Built-in presets

Media Services supports the following built-in encoding presets:

### **BuiltInStandardEncoderPreset**

[BuiltInStandardEncoderPreset](#) is used to set a built-in preset for encoding the input video with the Standard Encoder.

The following built-in presets are currently supported:

- **EncoderNamedPreset.AACGoodQualityAudio:** Produces a single MP4 file containing only stereo audio encoded at 192 kbps.
- **EncoderNamedPreset.AdaptiveStreaming:** This supports H.264 adaptive bitrate encoding. For more information, see [auto-generating a bitrate ladder](#).
- **EncoderNamedPreset.H265AdaptiveStreaming :** Similar to the AdaptiveStreaming preset, but uses the HEVC (H.265) codec. Produces a set of GOP aligned MP4 files with H.265 video and stereo AAC audio. Auto-generates a bitrate ladder based on the input resolution, bitrate and frame rate. The auto-generated preset will never exceed the input resolution. For example, if the input is 720p, output will remain 720p at best.
- **EncoderNamedPreset.ContentAwareEncoding:** Exposes a preset for H.264 content-aware encoding. Produces a set of GOP-aligned MP4s by using content-aware encoding. Given any input content, the service performs an initial lightweight analysis of the input content, and uses the results to determine the optimal number of layers, appropriate bitrate and resolution settings for delivery by adaptive streaming. This preset is particularly effective for low and medium complexity videos, where the output files will be at lower bitrates but at a quality that still delivers a good experience to viewers. The output will contain MP4 files

with video and audio interleaved. This preset only produces output up to 1080P HD. If 4K output is required, you can configure the preset with [PresetConfigurations](#) by using the "maxHeight" property. For more information, see [content-aware encoding](#).

- **EncoderNamedPreset.H265ContentAwareEncoding:** Exposes a preset for HEVC (H.265) content-aware encoding. Produces a set of GOP-aligned MP4s by using content-aware encoding. Given any input content, the service performs an initial lightweight analysis of the input content, and uses the results to determine the optimal number of layers, appropriate bitrate and resolution settings for delivery by adaptive streaming. This preset is particularly effective for low and medium complexity videos, where the output files will be at lower bitrates but at a quality that still delivers a good experience to viewers. The output will contain MP4 files with video and audio interleaved. This preset produces output up to 4K HD. If 8K output is required, you can configure the preset with [PresetConfigurations](#) by using the "maxHeight" property.
- **EncoderNamedPreset.H264MultipleBitrate1080p:** produces a set of eight GOP-aligned MP4 files, ranging from 6000 kbps to 400 kbps, and stereo AAC audio. Resolution starts at 1080p and goes down to 360p.
- **EncoderNamedPreset.H264MultipleBitrate720p:** produces a set of six GOP-aligned MP4 files, ranging from 3400 kbps to 400 kbps, and stereo AAC audio. Resolution starts at 720p and goes down to 360p.
- **EncoderNamedPreset.H264MultipleBitrateSD:** produces a set of five GOP-aligned MP4 files, ranging from 1600 kbps to 400 kbps, and stereo AAC audio. Resolution starts at 480p and goes down to 360p.
- **EncoderNamedPreset.H264SingleBitrate1080p:** produces an MP4 file where the video is encoded with H.264 codec at 6750 kbps and a picture height of 1080 pixels, and the stereo audio is encoded with AAC-LC codec at 128 kbps. If you desire lower bitrates for audio, you can build a custom encoding preset in your transform and adjust the sampling rate or channel count to get down to lower values for AAC-LC.
- **EncoderNamedPreset.H264SingleBitrate720p:** produces an MP4 file where the video is encoded with H.264 codec at 4500 kbps and a picture height of 720 pixels, and the stereo audio is encoded with AAC-LC codec at 128 kbps. If you desire lower bitrates for audio, you can build a custom encoding preset in your transform and adjust the sampling rate or channel count to get down to lower values for AAC-LC.

- **EncoderNamedPreset.H264SingleBitrateSD**: produces an MP4 file where the video is encoded with H.264 codec at 2200 kbps and a picture height of 480 pixels, and the stereo audio is encoded with AAC-LC codec at 128 kbps. If you desire lower bitrates for audio, you can build a custom encoding preset in your transform and adjust the sampling rate or channel count to get down to lower values for AAC-LC.
- **EncoderNamedPreset.H265SingleBitrate720P**: produces an MP4 file where the video is encoded with HEVC (H.265) codec at 1800 kbps and a picture height of 720 pixels, and the stereo audio is encoded with AAC-LC codec at 128 kbps.
- **EncoderNamedPreset.H265SingleBitrate1080p**: produces an MP4 file where the video is encoded with HEVC (H.265) codec at 3500 kbps and a picture height of 1080 pixels, and the stereo audio is encoded with AAC-LC codec at 128 kbps.
- **EncoderNamedPreset.H265SingleBitrate4K**: produces an MP4 file where the video is encoded with HEVC (H.265) codec at 9500 kbps and a picture height of 2160 pixels, and the stereo audio is encoded with AAC-LC codec at 128 kbps.

To see the most up-to-date presets list, see [built-in presets to be used for encoding videos](#).

## Custom presets

Media Services fully supports customizing all values in presets to meet your specific encoding needs and requirements.

### StandardEncoderPreset

[StandardEncoderPreset](#) describes settings to be used when encoding the input video with the Standard Encoder. Use this preset when customizing Transform presets.

### Considerations

When creating custom presets, the following considerations apply:

- All values for height and width on AVC content must be a multiple of four.
- In Azure Media Services v3, all of the encoding bitrates are in bits per second. This is different from the presets with our v2 APIs, which used kilobits/second as the unit. For example, if the bitrate in v2 was specified as 128 (kilobits/second), in v3 it would be set to 128000 (bits/second).

# Preset schema

In Media Services v3, presets are strongly typed entities in the API itself. You can find the "schema" definition for these objects in [Open API Specification \(or Swagger\)](#). You can also view the preset definitions (like `StandardEncoderPreset`) in the [REST API](#), [.NET SDK](#) (or other Media Services v3 SDK reference documentation).

## Scaling encoding in v3

For accounts created with the **2020-05-01** or later version of the API or through the Azure portal, scaling and media reserved units are no longer required. Scaling will be automatic and handled by the service internally.

## Billing

Media Services does not bill for canceled or errored jobs. For example, a job that has reached 50% progress and is canceled is not billed at 50% of the job minutes. You are only charged for finished jobs.

For more information, see [pricing](#).

## Encoding samples

See the extensive list of [Encoding Samples](#).

## Get help and support

You can contact Media Services with questions or follow our updates by one of the following methods:

- [Q & A](#)
- [Stack Overflow](#). Tag questions with `azure-media-services`.
- [@MSFTAzureMedia](#) or use [@AzureSupport](#) to request support.
- Open a support ticket through the Azure portal.

# Live streaming with Azure Media Services v3

Article • 01/10/2023



Media Services API v3

## ⚠️ Warning

Azure Media Services will be retired June 30th, 2024. For more information, see the [AMS Retirement Guide](#).

Azure Media Services enables you to deliver live events to your customers on the Azure cloud. To stream your live events with Media Services, you'll need to set up a live video encoder that converts signals from a camera (or another device, like a laptop) into a contribution feed that is sent to Media Services. The contribution feed can include signals related to advertising, such as SCTE-35 markers. For a list of recommended live streaming encoders, see [live streaming encoders](#).

If you haven't used an on-premises encoder before, try the [Create an Azure Media Services live stream with OBS](#) quickstart.

## Dynamic packaging and delivery

With Media Services, you can take advantage of [dynamic packaging](#), which allows you to preview and broadcast your live streams in [MPEG DASH, HLS, and Smooth Streaming formats](#) from the contribution feed. Your viewers can play back the live stream with any HLS, DASH, or Smooth Streaming compatible players. See the [list of tested players](#) and try the [Media Services 3rd-party player samples](#).

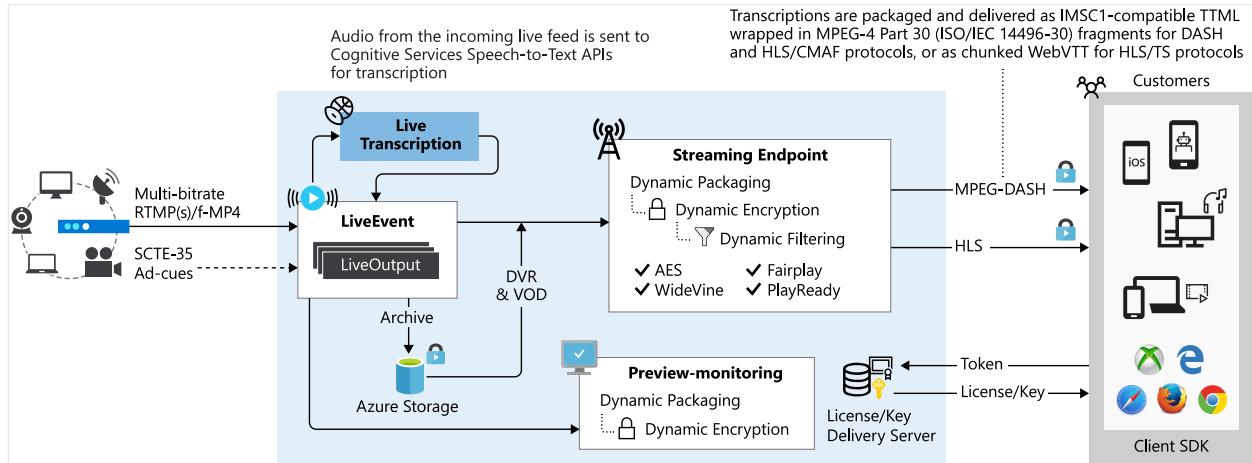
## Live event types

[Live events](#) are ingest and process live video feeds. A live event can be set to either:

- *pass-through* when an on-premises live encoder sends a multiple bitrate stream, or
- *live encoding* when an on-premises live encoder sends a single bitrate stream. For details about live outputs, see [Live events and live outputs](#).

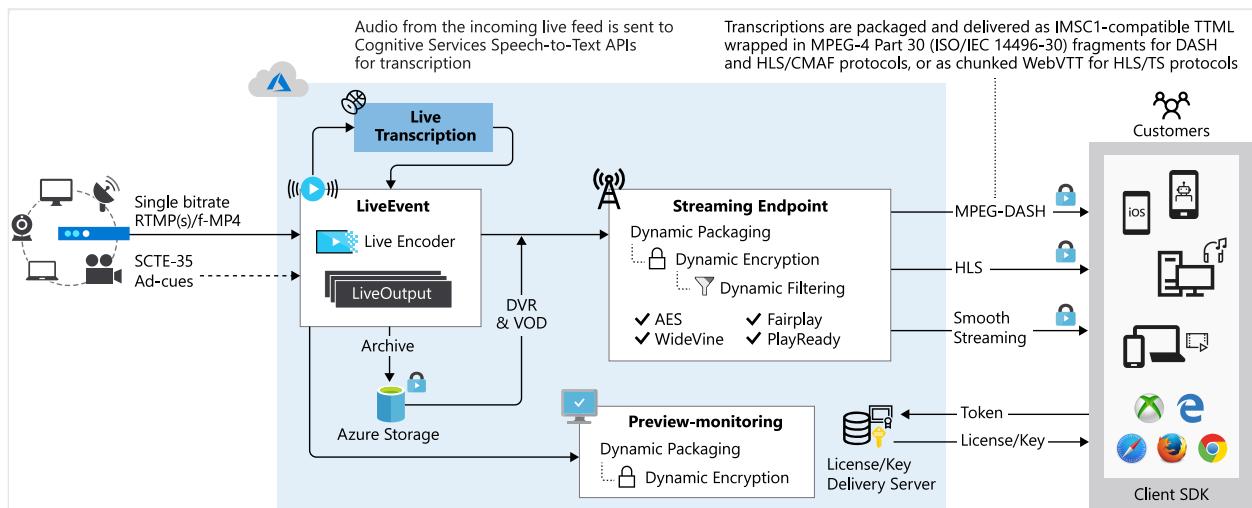
# Pass-through

When using the pass-through **Live Event** (basic or standard), you rely on your on-premises live encoder to generate a multiple bitrate video stream and send that as the contribution feed to the Live Event (using RTMP or fragmented-MP4 input protocol). The Live Event then passes the incoming video stream to the dynamic packager (Streaming Endpoint) without any further processing. A pass-through Live Event is optimized for long-running live events or 24x365 linear live streaming.



# Live encoding

To use live encoding, configure your on-premises live encoder to send a single bitrate video (up to 32Mbps aggregate) to the Live Event (using RTMP or fragmented-MP4 input protocol). The Live Event transcodes the incoming single bitrate stream into multiple bitrate video streams at varying resolutions. This improves delivery for playback devices with industry standard protocols like MPEG-DASH, Apple HTTP Live Streaming (HLS), and Microsoft Smooth Streaming.



# Live event options

## Dynamic encryption

Dynamic encryption enables you to dynamically encrypt your live or on-demand content with AES-128 or any of the three major digital rights management (DRM) systems: Microsoft PlayReady, Google Widevine, and Apple FairPlay. Media Services also provides a service for delivering AES keys and DRM (PlayReady, Widevine, and FairPlay) licenses to authorized clients. For more information, see [dynamic encryption](#).

Widevine is a service provided by Google Inc. and subject to the terms of service and Privacy Policy of Google, Inc.

## Dynamic filtering

Dynamic filtering is used to control the number of tracks, formats, bitrates, and presentation time windows that are sent out to the players. For more information, see [filters and dynamic manifests](#).

## Live transcription

Live transcription is a feature you can use with live events that are either pass-through or live encoding. For more information, see [live transcription](#). When this feature is enabled, the service uses the [Speech-To-Text](#) feature of Cognitive Services to transcribe the spoken words in the incoming audio into text. This text is then made available for delivery along with video and audio in MPEG-DASH and HLS protocols.

### **Important**

You *should* use GOP sizes of 2 seconds for live events. You **must** use GOP sizes of 4 seconds or below for passthrough live events with live transcriptions in order to get correct transcription data. If you choose to use higher GOP size, the transcription data might have defects, e.g. missing content.

## Security considerations for closed captions, subtitles, and timed-metadata delivery

The dynamic encryption and DRM features of Azure Media Services has limits to consider when attempting to secure content delivery that includes live transcriptions, captions, subtitles, or timed-metadata. The DRM subsystems, including PlayReady, FairPlay, and Widevine do not support the encryption and licensing of text tracks. The

lack of DRM encryption for text tracks limits your ability to secure the contents of live transcriptions, manual inserted captions, uploaded subtitles, or timed-metadata signals that may be inserted as separate tracks.

To secure your captions, subtitles, or timed-metadata tracks, follow these guidelines:

1. Use AES-128 Clear Key encryption. When enabling AES-128 clear key encryption, the text tracks can be configured to be encrypted using a full "envelope" encryption technique that follows the same encryption pattern as the audio and video segments. These segments can then be decrypted by a client application after requesting the decryption key from the Media Services Key Delivery service using an authenticated JWT token. This method is supported by the Azure Media Player, but may not be supported on all devices and can require some client-side development work to make sure it succeeds on all platforms.
2. Use CDN token authentication to protect the text (subtitle, captions, metadata) tracks being delivered with short form tokenized URLs that are restricted to geo, IP, or other configurable settings in the CDN portal. Enable the CDN security features using Verizon Premium CDN or other 3rd-party CDN configured to connect to your Media Services streaming endpoints.

### Warning

If you do not follow one of the guidelines above, your subtitles, captions, or timed-metadata text will be accessible as un-encrypted content that could be intercepted or shared outside of your intended client delivery path. This can result in leaked information. If you are concerned about the contents of the captions or subtitles being leaked in a secure delivery scenario, reach out to the Media Services support team for more information on the above guidelines for securing your content delivery.

## Live streaming workflow

To understand the live streaming workflow in Media Services v3, you have to first review and understand the following concepts:

- [Streaming endpoints](#)
- [Live events and live outputs](#)
- [Streaming locators](#)

## General steps

1. In your Media Services account, make sure the **streaming endpoint** (origin) is running.

2. Create a [live event](#).

When creating the event, you can specify to autostart it. Alternatively, you can start the event when you are ready to start streaming.

When autostart is set to true, the Live Event will be started right after creation. The billing starts as soon as the Live Event starts running. You must explicitly call Stop on the live event resource to halt further billing. For more information, see [live event states and billing](#).

3. Get the ingest URL(s) and configure your on-premises encoder to use the URL to send the contribution feed.

See [recommended live encoders](#).

4. Get the preview URL and use it to verify that the input from the encoder is actually being received.

5. Create a new **asset** object.

Each live output is associated with an asset, which it uses to record the video into the associated Azure blob storage container.

6. Create a **live output** and use the asset name that you created so that the stream can be archived into the asset.

Live Outputs start on creation and stop when deleted. When you delete the Live Output, you are not deleting the underlying asset and content in the asset.

7. Create a **streaming locator** with the [built-in streaming policy types](#).

To publish the live output, you must create a streaming locator for the associated asset.

8. List the paths on the **streaming locator** to get back the URLs to use (these are deterministic).

9. Get the hostname for the **streaming endpoint** (Origin) you wish to stream from.

10. Combine the URL from step 8 with the hostname in step 9 to get the full URL.

11. If you wish to stop making your **live event** viewable, you need to stop streaming the event and delete the **streaming locator**.

12. If you are done streaming events and want to clean up the resources provisioned earlier, follow the following procedure.

- Stop pushing the stream from the encoder.
- Stop the live event. Once the live event is stopped, it will not incur any charges. When you need to start it again, it will have the same ingest URL so you won't need to reconfigure your encoder.
- You can stop your streaming endpoint, unless you want to continue to provide the archive of your live event as an on-demand stream. If the live event is in stopped state, it will not incur any charges. However, if the streaming endpoint is still running, it will incur charges.

The asset that the live output is archiving to, automatically becomes an on-demand asset when the live output is deleted. You must delete all live outputs before a live event can be stopped. You can use an optional flag `removeOutputsOnStop` to automatically remove live outputs on stop.

### 💡 Tip

See [Live streaming tutorial](#), the article examines the code that implements the steps described above.

## Other important articles

- [Recommended live encoders](#)
- [Using a cloud DVR](#)
- [Live event types feature comparison](#)
- [States and billing](#)
- [Latency](#)
- [Quotas and limits](#)

## Live streaming FAQ

See the [live streaming questions in the FAQ](#).

## Get help and support

You can contact Media Services with questions or follow our updates by one of the following methods:

- [Q & A](#)
- [Stack Overflow](#). Tag questions with `azure-media-services`.
- [@MSFTAzureMedia](#) or use [@AzureSupport](#) to request support.

- Open a support ticket through the Azure portal.

# High Availability with Media Services and Video on Demand (VOD)

Article • 01/10/2023



Media Services API v3

## ⚠️ Warning

Azure Media Services will be retired June 30th, 2024. For more information, see the [AMS Retirement Guide](#).

## High availability for VOD

There is a high availability design pattern called [Geodes](#) in the Azure Architecture documentation. It describes how duplicate resources are deployed to different geographic regions to provide scalability and resiliency. You can use Azure services to create such an architecture to cover many high availability design considerations such as redundancy, health monitoring, load balancing, and data backup and recovery. One such architecture is described below with details on each service used in the solution as well as how the individual services can be used to create a high availability architecture for your VOD application.

## Sample

There is a sample available for you to use to become familiar with high availability with Media Services and Video on Demand (VOD). It also goes into more detail about how the services are used for a VOD scenario. The sample is not intended to be used in production in its current form. Carefully review the sample code and the readme, particularly the section on [Failure Modes](#) before integrating it into a production application. A production implementation of high availability for Video on Demand (VOD) should also carefully review their Content Delivery Network (CDN) strategy. Check out the [code on GitHub](#).

## Overview of services

The services used in this example architecture include:

Icon	Name	Description
	Media services account	<p><b>Description:</b>  A Media Services account is the starting point for managing, encrypting, encoding, analyzing, and streaming media content in Azure. It is associated with an Azure Storage account resource. The account and all associated storage must be in the same Azure subscription.</p> <p><b>VOD use:</b>  These are the services you use to encode and deliver your video and audio assets. For high availability, you would set up at least two Media Services accounts, each in a different region. <a href="#">Learn more about Azure Media Services</a>.</p>
	Storage account	<p><b>Description:</b>  An Azure storage account contains all your Azure Storage data objects: blobs, files, queues, tables, and disks. Data is accessible from anywhere in the world over HTTP or HTTPS.</p> <p>Each Media Services account, in each region would have a storage account in the same region.</p> <p><b>VOD use:</b>  You can store input and output data for VOD processing and streaming. <a href="#">Learn more about Azure Storage</a>.</p>
	Azure Storage Queue	<p><b>Description:</b>  Azure Queue storage is a service for storing large numbers of messages that can be accessed from anywhere in the world via authenticated calls using HTTP or HTTPS.</p> <p><b>VOD use:</b>  Queues can be used to send and receive messages to coordinate activities among different modules. The sample uses an Azure Storage Queue but Azure provides other types of queues such as Service Bus and Service Fabric Reliable Queues which may better fit your needs. <a href="#">Learn more about Azure Queue</a>.</p>
	Azure Cosmos DB	<p><b>Description:</b>  Azure Cosmos DB is Microsoft's globally distributed, multi-model database service that independently scales throughput and storage across any number of Azure regions worldwide.</p> <p><b>VOD use:</b>  Tables can be used to store job output status records and to track health state of each Media Services instance. You can also track/record the status of each call to the Media Services API. <a href="#">Learn more about Azure Cosmos DB</a>.</p>

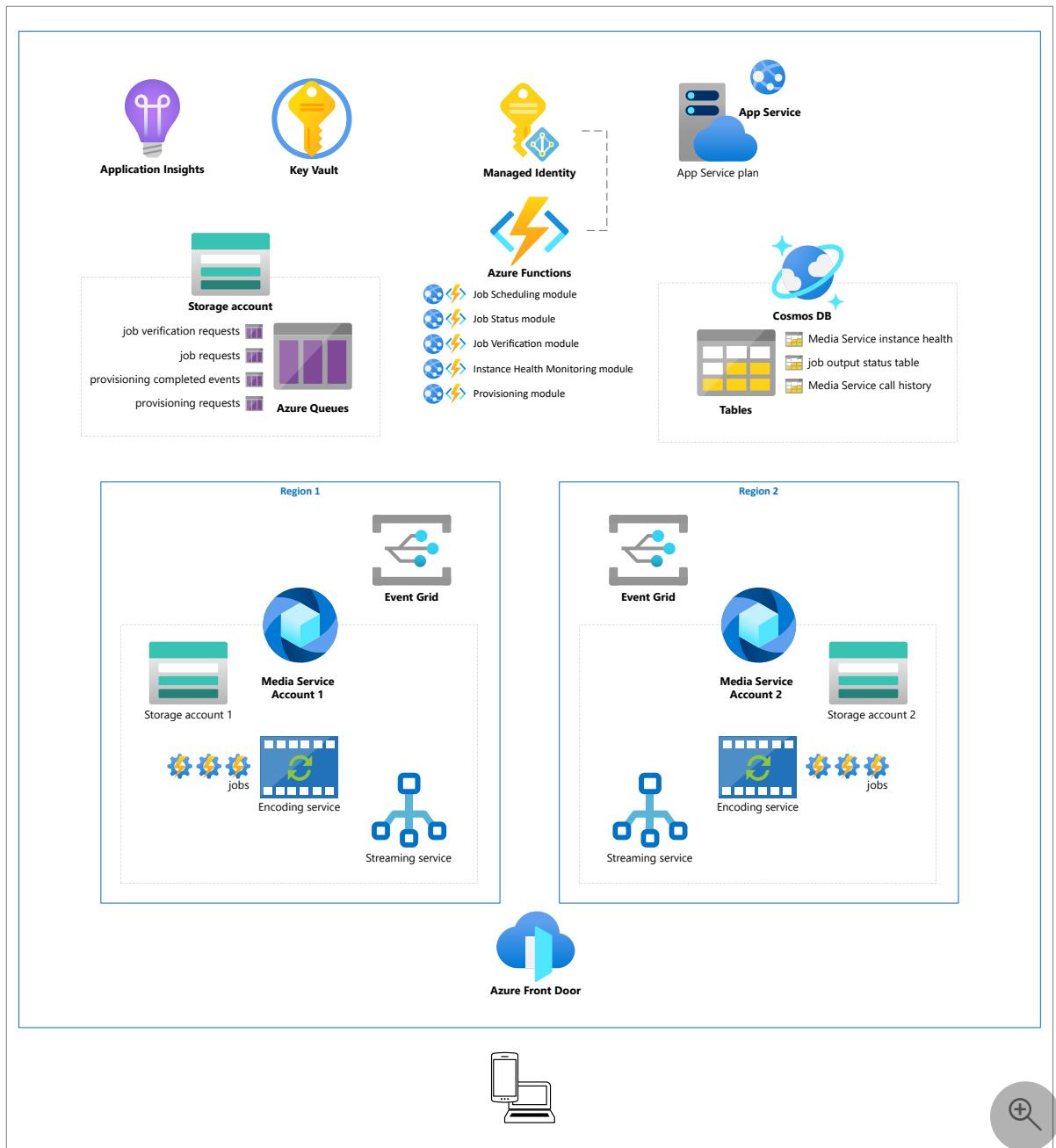
Icon	Name	Description
	Managed Identity	<p><b>Description:</b>  Managed identity is a feature of Azure AD that provides an automatically managed identity in Azure AD. It can be used to authenticate to any service that supports Azure AD authentication, including Key Vault, without storing credentials in code.</p> <p><b>VOD use:</b>  Azure Functions can use Managed Identity to authenticate to Media Services instances to connect to Key Vault. <a href="#">Learn more about Managed Identity</a>.</p>
	Key Vault	<p><b>Description:</b>  Azure Key Vault can be used to securely store and tightly control access to tokens, passwords, certificates, API keys, and other secrets. It can also be used as a Key Management solution. Azure Key Vault makes it easy to create and control the encryption keys used to encrypt your data. It can easily provision, manage, and deploy public and private Transport Layer Security/Secure Sockets Layer (TLS/SSL) certificates for use with Azure and internal connected resources. Secrets and keys can be protected either by software or FIPS 140-2 Level 2 validated HSMs.</p> <p><b>VOD use:</b>  Key Vault can be used to set up access policies for the service principal for your application. It can be used to store the connection string to storage accounts. We use Key Vault to store connection strings to storage accounts and cosmos db. You can also use Key Vault to store overall cluster configuration. For each Media Service Instance you can store subscription ID, resource group name and account name. For more details, see how it is used in the sample. <a href="#">Learn more about Key Vault</a>.</p>

Icon	Name	Description
 Azure Functions	<p><b>Description:</b> Run small pieces of code (called "functions") without worrying about application infrastructure with Azure Functions. <a href="#">Learn more about Azure Functions</a>.</p> <p><b>VOD use:</b> Azure Functions can be used to store host the modules of your VOD application. Modules for a VOD application could include:</p> <p><b>Job Scheduling Module</b> The job scheduling module would be for submitting new jobs to a Media Services cluster (two or more instances in different regions). It would track the health status of each Media Services instance and would submit a new job to the next healthy instance.</p> <p><b>Job Status Module</b> The job status module would be listening to job output status events coming from Azure Event Grid service. It would store events to event store to minimize the number of calls to Media Services APIs by rest of the modules.</p> <p><b>Instance Health Module</b> This module would track submitted jobs and determine the health status for each Media Services instance. It would track finished jobs, failed jobs and jobs that never finished.</p> <p><b>Provisioning Module</b> This module would provision processed assets. It would copy asset data to all Media Services instances and set up Azure Front Door service to ensure that assets could be streamed even if some Media Services instances weren't available. It would also set up streaming locators.</p> <p><b>Job Verification Module</b> This module would track each submitted job, resubmit failed jobs and perform job data clean up once a job is successfully finished.</p>	
 App Service (and plan)	<p><b>Description:</b> Azure App Service is an HTTP-based service for hosting web applications, REST APIs, and mobile back ends. It supports .NET, .NET Core, Java, Node.js, PHP, or Python. Applications run and scale on both Windows and Linux-based environments.</p> <p><b>VOD use:</b> Each module would be hosted by an App Service. <a href="#">Learn more about App Service</a>.</p>	

Icon	Name	Description
	Azure Front Door	<p><b>Description:</b> Azure Front Door is used to define, manage, and monitor the global routing of web traffic by optimizing for best performance and quick global failover for high availability.</p> <p><b>VOD use:</b> Azure Front Door could be used to route traffic to streaming endpoints. <a href="#">Learn more about Azure Front Door.</a></p>
	Azure Event Grid	<p><b>Description:</b> Created for event-based architectures, Event Grid has built-in support for events coming from Azure services, like storage blobs and resource groups. It also has support for custom topic events. Filters can be used to route specific events to different endpoints, multicast to multiple endpoints, and to make sure events are reliably delivered. It maximizes availability by natively spreading across multiple fault domains in every region, and across availability zones.</p> <p><b>VOD use:</b> Event Grid can be used to track all application events and store them to persist job status. <a href="#">Learn more about Azure Event Grid.</a></p>
	Application Insights	<p><b>Description:</b> Application Insights, a feature of Azure Monitor, is an extensible Application Performance Management (APM) service for developers and DevOps professionals. It is used to monitor live applications. It detects performance anomalies, and includes analytics tools to diagnose issues and to understand what users do with an app. It's designed to help you continuously improve performance and usability.</p> <p><b>VOD use:</b> All logs can be sent to Application Insights. It would be possible to see what instance processed each job by searching for successfully created job messages. It could contain all submitted job metadata including the unique identifier and instance name information. <a href="#">Learn more about Application Insights.</a></p>

## Architecture

This high-level diagram shows the architecture of the sample provided to get you started with high availability and media services.



## Best practices

### Regions

- Create two (or more) Azure Media Services accounts. The two accounts need to be in different regions. For more information, see [Regions in which the Azure Media Services service is deployed](#).
- Upload your media to the same region from which you are planning to submit the job.
- If you then need to resubmit the job to another region, you can use `JobInputHttp` or use `Copy-Blob` to copy the data from the source Asset container to an Asset container in the alternate region.

# Monitoring

- Subscribe for `JobStateChange` messages in each account via Azure Event Grid.
  - Use the [Microsoft.Azure.EventGrid SDK](#) (which supports Media Services events natively).
  - You can also consume Event Grid events via Azure Functions.

For more information:

- See the [Audio Analytics sample](#) which shows how to monitor a job with Azure Event Grid including adding a fallback in case the Azure Event Grid messages are delayed for some reason.
- When you create a [job](#):
  - Randomly select an account from the list of currently used accounts (this list will normally contain both accounts but if issues are detected it may contain only one account). If the list is empty, raise an alert so an operator can investigate.
  - Create a record to keep track of each inflight job and the region/account used.
- When your `JobStateChange` handler gets a notification that a job has reached the scheduled state, record the time it enters the scheduled state and the region/account used.
- When your `JobStateChange` handler gets a notification that a job has reached the processing state, mark the record for the job as processing and record the time it enters the processing state.
- When your `JobStateChange` handler gets a notification that a job has reached a final state (Finished/Errored/Canceled), mark the record for the job appropriately.
- Have a separate process that periodically looks at your records of the jobs
  - If you have jobs in the scheduled state that haven't advanced to the processing state in a reasonable amount of time for a given region, remove that region from your list of currently used accounts. Depending on your business requirements, you could decide to cancel those jobs right away and resubmit them to the other region. Or, you could give them some more time to move to the next state.
  - If a region was removed from the account list, monitor it for recovery before adding it back to the list. The regional health can be monitored via the existing jobs on the region (if they weren't canceled and resubmitted), by adding the account back to the list after a period of time, and by operators monitoring Azure communications about outages that may be affecting Azure Media Services.

# Get help and support

You can contact Media Services with questions or follow our updates by one of the following methods:

- [Q & A](#)
- [Stack Overflow](#). Tag questions with `azure-media-services`.
- [@MSFTAzureMedia](#) or use [@AzureSupport](#) to request support.
- Open a support ticket through the Azure portal.

# Offline audio dubbing

Azure AI services    Azure AI Speech

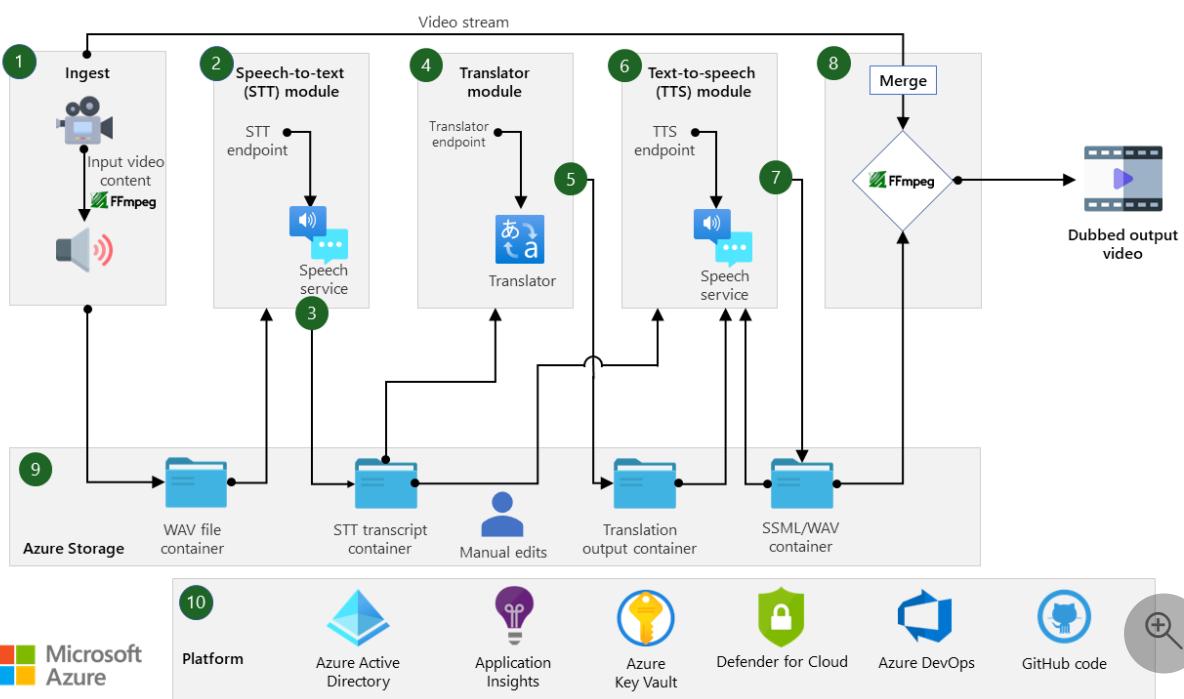
This guide describes how the AI tools in Azure Cognitive Services can help you automate an offline dubbing process and ensure a high-quality dubbed version of original input.

*Dubbing* is the process of placing a replacement track over an original audio/video source. It includes speech-to-speech transformation from a source to a target speech stream. There are two types of dubbing. *Real-time dubbing* modifies the original audio/video track for the content. *Offline dubbing* refers to a postproduction process. The offline process enables human assistance to improve the overall outcome as compared to that of real-time implementations. You can correct errors at each stage and add enough metadata to make the output more truthful to the original. You can use the same pipeline to provide subtitles for the video track.

*FFmpeg* is a trademark of Fabrice Bellard.

## Architecture

This architecture shows a pipeline that performs human-assisted speech-to-speech dubbing in offline mode. The successful completion of each module triggers the next one. The speech-to-speech pipeline contains the required elements for generating an offline dubbed audio stream. It uses the transcript of the spoken text to produce subtitles. You can optionally enhance the subtitles to produce closed captioning as well.



[Download a PowerPoint file](#) of this architecture.

## Workflow

1. **Video ingestion:** FFmpeg open-source software divides the input video content into an audio stream and a video stream. The pipeline saves the audio stream as a WAV file in *WAV file container* (blob storage). FFmpeg merges the video stream later in the process.
2. **Speech-to-text module:** The speech-to-text module uses the audio file from *WAV file container* as input. It uses Azure Speech service to determine the language or languages of the source audio and to detect individual speakers. It transcribes the audio and saves it in text format. The speech-to-text module stores the text file, a subtitle caption file, and a word-level time stamp file in *STT transcript container*.
3. **Subtitle file production and filtering:** The speech-to-text module produces the subtitle file along with notes to aid manual review of the content and correct errors introduced by the speech-to-text module. The subtitle file includes text, time stamps, and other metadata, like speaker ID and language ID, that enables the rest of the pipeline to function correctly. The pipeline stores the subtitle file in *STT transcript container* as a WebVTT file. It contains notes on potential points for human verification. These notes don't interfere with the ability to add the file as a subtitle file to the final output video. After the speech-to-text module produces the subtitle file, human editors can correct errors and optionally add emotion tags to reflect the input audio.
4. **Translator module:** The translation of the text is independent of time stamps. The input subtitle text is compiled based on the speakers. The translator service requires only the text with its context to perform the mapping to the target language. The timing and placement of the audio is forwarded to the text-to-speech module in the pipeline for proper representation of the source audio in the output. This input also includes language identification of the various segments. It enables the translation service to skip segments that don't require translation. The translation module can also perform content filtering.
5. **Target subtitle file production:** This module is responsible for reproducing the subtitle file in the target language. It does this by replacing the text in the transcribed input file. It maintains the time stamps and metadata that are associated with the text. The metadata might also include hints for the human reviewer and correct the content to improve the quality of translated text. Like the speech-to-text output, the file is a WebVTT file that's stored in Azure Storage. It includes notes that highlight potential points for human editing.
6. **Text-to-speech module:** The Speech service converts the transcribed text from the source audio to the target speech track. The service uses Speech Synthesis Markup

Language (SSML) to represent the conversion parameters of the target text. It fine-tunes pitch, pronunciation, emotion, speech rate, pauses, and other parameters in the text-to-text process. This service enables multiple customizations, including the ability to use custom neural voices. It generates the output audio files (WAV files), which are saved in blob storage for further use.

7. **Timing adjustment and SSML file production:** The audio segments in source and target languages might be of different lengths, and the specifics for each language might differ. The pipeline adjusts the timing and positioning of the speech within the target output stream to be truthful to the source audio but still sound natural in the target language. It converts the output speech to an SSML file. It also matches the audio segments to the right speaker and provides the right tone and emotions. The pipeline produces one SSML file as output for the input audio. The SSML file is stored in Azure Storage. It highlights segments that might benefit from human revision, for both placement and rate.
8. **Merge:** FFmpeg adds the generated WAV file to the video stream to produce a final output. The subtitles that are generated in this pipeline can be added to the video.
9. **Azure Storage:** The pipeline uses Azure Storage to store and retrieve content as it's produced and processed. Intermediate files are editable if you need to correct errors at any stage. You can restart the pipeline from different modules to improve the final output via human verification.
10. **Platform:** The platform components complete the pipeline, enabling enhanced security, access rights, and logging. Microsoft Entra ID and Azure Key Vault regulate access and store secrets. You can enable Application Insights to perform logging for debugging.

The pipeline is aware of errors. This awareness is significant when the language or speaker changes. The key to getting the right speech-to-text and translation output is understanding the context of the speech. You might need to check and correct the output text after each step. For certain use cases, the pipeline can perform automatic dubbing, but it's not optimized for real-time speech-to-speech dubbing. In offline mode, the full audio is processed before the pipeline continues. This enables each module to run for the full length of the audio and aligns with module design.

## Components

- [Azure Cognitive Services](#) is a suite of cloud-based AI services that helps developers build cognitive intelligence into applications without having direct AI or data science skills or knowledge. The services are available through REST APIs and client library SDKs in popular development languages.

- [Azure Speech service](#) unifies speech-to-text, text-to-speech, speech translation, voice assistant, and speaker recognition functionality into a single service-based subscription offering.
- [Azure translator service](#) is a cloud-based neural machine translation service that's part of the Cognitive Services family of REST APIs. You can use it with any operating system. Translator service enables many Microsoft products and services that are used by thousands of businesses worldwide to perform language translation and other language-related operations.

## Use cases

Audio dubbing is one of the most useful and widely used tools for media houses and OTT platforms. It helps increase global reach by adding relevance to content for local audiences. Offline processing works best for dubbing, but real-time dubbing can be used for some applications.

## Offline dubbing

Following are some use cases in which offline audio dubbing is the best choice:

- Reproduce films and other media in a different language.
- Replace original on-set audio that isn't usable because of poor recording equipment, ambient noise, or inadequate performance by the subject.
- Implement content filtering to remove profanity or slang. Implement pronunciation correction in applicable segments to adjust content for the target audience.
- Generate subtitles. The dubbing pipeline generates a transcript from the audio/video as a byproduct of the process. You can use this transcript to generate subtitles in the original or dubbed language. You can enhance the transcribed text to produce closed captioning for media.

These scenarios have traditionally presented difficult problems that required manual intervention by trained professionals. Given recent improvements in speech and language modeling, you can inject AI modules into these processes to make them more efficient and cost effective.

## Real-time dubbing

You can implement real-time dubbing with this pipeline, but it doesn't allow human intervention for corrections. Also, longer audio files might require additional processing power and time, which increases the cost of transcription. One way to handle this

limitation is to divide the input into smaller segments of audio that are pushed through the pipeline as separate elements. However, this technique might present some problems:

- Dividing the input audio into smaller clips can disrupt context and reduce the quality of transcription. Default speaker ID tagging might yield inconsistent results, especially if you don't use custom models for speaker identification. Default tagging that's based on the order of appearance makes accurate speaker mapping difficult when audio is divided into fragments. You can resolve this challenge if the source audio has separate channels assigned to specific speakers.
- Timing is crucial to obtaining natural and synchronized output audio. You need to minimize latency in the production pipeline and measure it accurately. By identifying the delay caused by the pipeline and using sufficient buffer, you can ensure that the output consistently matches the pace of the input audio. You also need to account for potential pipeline failures and maintain output consistency accordingly.

Taking into account the previous points, you could use this pipeline to process real-time audio in scenarios where:

- Audio contains only one speaker who's speaking clearly in one language.
- There's no overlapping audio from multiple speakers, and there's a clear pause between any speaker or language changes.
- Different audio channels are produced for each speaker, and they can be processed independently.

## Design considerations

This section describes services and features in the audio dubbing pipeline, highlighting the factors to consider to generate high-quality output. Note that this guide's implementation uses base or universal models for all cognitive models. Custom models can enhance the output of each service and the overall solution.

### Speech-to-text

The performance of each module significantly influences the quality of the overall output. When the input quality deteriorates because of speech issues or background noise, errors can occur in subsequent modules, making it increasingly difficult and time-consuming to generate high-quality output that's free of errors. However, you can take advantage of various features in Speech service to minimize this problem and ensure high-quality output.

**Segment the audio properly.** Proper segmentation of the source audio is critical to the process of improving the overall solution. The speech-to-text module performs segmentation based on significant pauses, which is a coarse division of source audio. Additionally, the speech-to-text module divides longer segments of audio into segments that remain within the speech-to-text service. However, this segmentation introduces problems, like incorrect context and inappropriate speaker mapping.

**Extract timing information.** The process of dubbing involves a margin of timing misalignment that originates from the differences in how content is expressed in different languages. The speech-to-text produces timing information from the source audio, which is fundamental to the reproduction of correctly timed speech in the target language. Although it's not required in the translation of the text, the timing information needs to be passed on to the module that generates the SSML file. The granularity of the timing data remains at the segment level. And because the translation doesn't provide alignment information, the word-level timing isn't transferable to the target language. Therefore, modifications of the text input to accommodate context in translation need to be reversible.

**Present audio source as one stream.** The ability to distinguish speakers in the source audio is another important factor in producing correct dubbing targets. The process of partitioning an audio stream into homogeneous segments according to the identity of each speaker is called *diarization*. When different speaker segments aren't available as separate audio streams, the easiest way to perform consistent diarization is to present the audio source as one stream, as opposed to breaking it down into individual segments. If you present the audio in this way, the various speakers are indexed according to their order of appearance, which reduces the need to use custom speaker-identification models to identify speakers accurately.

**Consider combining models.** Custom models enable better speaker identification and appropriate labeling of the text produced by any specific speaker. In some cases, it might be best to mix two approaches, building speaker identification models for key speakers and diarizing the remaining speakers in order of their appearance by using the service. You can implement this method by using Azure speaker recognition to train speaker identification models for the key speakers. You would need to collect enough audio data for each key speaker. Use the text transcripts generated by Azure speech-to-text to identify speaker changes and segment the audio data into different speaker segments. Then post-process and refine the results as needed. This process might involve combining or splitting segments, adjusting segment boundaries, or correcting speaker identification errors.

The implementation details might vary depending on your specific requirements and the characteristics of your audio data. Additionally, diarization systems can be

computationally intensive, so you need to consider the processing speed of your speech-to-text model. You might need to use a model that's optimized for speed or use parallel-processing techniques to speed up the diarization process.

**Use the lexical text to reduce potential errors generated by the ITN.** Speech-to-text provides an array of formatting features to ensure that the transcribed text is legible and appropriate. Speech-to-text produces several representations of the input speech, each with different formatting. The recognized text obtained after disfluency removal, inverse text normalization (ITN), punctuation, capitalization, profanity filter, and so on, is called *display text*. The display text improves the readability of the text and uses pre-built models to define how to display different components. For example, the display format might include the use of capital letters for proper nouns or the insertion of commas to separate phrases. This format should be moved forward in the pipeline because it helps provide context in the translation and minimizes the need to reformat the text.

The words recognized by the service appear in a format known as the *lexical format*. The lexical format includes factors like the spelling of words and the use of homophones. The architecture uses this format to assess the extent of human intervention that's needed for correction on the speech-to-text output. This format represents the actual recognition of the speech-to-text service, and the formatting doesn't mask mismatched recognition. For example, the lexical format might represent the word "there" as "their" if the context suggests that it's the intended meaning.

**Filter profanity later in the pipeline.** Speech-to-text provides profanity filtering as an option. The architecture presented here doesn't use profanity filtering. You can enable profanity filtering via the [display formatting settings](#). It's better to avoid using this filtering early in the pipeline because it might introduce some loss of context and result in mismatches as the pipeline proceeds. However, the filtering affects the display text and MaskedITN, not the lexical format in the speech-to-text output.

**Consider creating a joint-language model when you use a custom model.** Speech service provides a language identification feature. To use this feature, you need to define, beforehand, the locales that are relevant to the input audio. Language identification requires some context, and sometimes there's a lag in flagging a language change. If you use custom models, consider creating a joint-language model, where the model is trained with utterances from multiple languages. This approach is useful for handling speech from regions where it's common to use two different languages together, like Hinglish, which contains both Hindi and English.

**Ensure clear turn-taking and natural pauses between speakers.** Like language switching, user switching also works best when the speakers take turns and there's a natural pause between speakers. If there are overlapping conversations or a short pause

between speakers and the audio is bundled into one channel, the speech-to-text output might contain errors that require human intervention. Background speech that might not be the focus of the content but that affects the output quality can cause similar errors. Correcting these errors requires word-level timing to identify the appropriate offsets for various speakers.

## Translator service

The translator service provides features that you can use to generate higher-quality output.

**Maintain enough context.** The main point to consider when you use the translator service is how much context to present in order to produce an accurate translation. To help with accurate translation, consider bundling the text from the same speaker ID, within a meaningful timespan if the context is observed appropriately. Note that speech-to-text divides the input audio based on meaningful pauses, but, for longer monologues, the audio is divided at approximately 30-second segments. This segmentation might result in some loss of context. Also, overlapping speech from different speakers might disturb the context and produce a poor translation output. Therefore, human intervention is frequently indispensable during this step.

**Consider skipping segments in multilingual sources.** In some cases, input includes multiple languages and some segments don't require translation. For example, the source and target locales might match, or terminology or jargon might be used as is. To skip languages that you don't want to translate, each segment needs to define the source and target locales for the translator service. The service needs a configuration file that defines languages that shouldn't be translated. To reduce costs, consider filtering out segments that don't require translation so that you don't input them into the translator service.

**Filter caption profanity independently.** Like speech-to-text, the translator service enables you to filter profanity. Masking or marking content for profanity can make captions more appropriate to the target audience. However, removing content via profanity filtering might confuse the timing analysis and placement of the speech output by the service. It might also lead to misinterpretations of the context. You should consider independently producing the captions file that's filtered for profanity.

**Ensure accurate timing alignment in captions.** Although the speech-to-text output produces word-level time stamps, the translator service doesn't match the time stamps with alignment data. Therefore, the time stamps in the captions file in WebVTT format remain the same as the time stamps in the source file. This might present marginal

misalignment in the presentation of captions because translating to a different language might reduce or increase the length of the content.

## Text-to-speech

**Select appropriate voices.** Based on the type of content and the intended audience, the selection of a speaker voice can have a significant influence on how well the content is received. The service provides [pre-built voices](#) in a range of languages in male and female voices. It also provides pre-built models that allow injection of emotions.

Alternatively, you can use a custom neural voice that's available in two versions, Lite and Pro, for the target speech language. If you use the Pro version, you can add emotions to the voice to model the source speech input. In all cases, you need to define a voice for each speaker and for each language. Unless you check the speech-to-text output to ensure that there are no unidentified speakers, you need to define a default voice.

**Perform speech placement.** The placement of the target speech and the integration of it into one SSML file are key elements in the text-to-text module. The following sections describe these elements.

*Integrate SSML files accurately.* SSML enables you to customize text-to-text output with identifying details on audio formation, speakers, pauses, and timing placement of the target audio. You can use it to format text-to-text output attributes like pitch, pronunciation, rate, and emotion. For more information, see [Speech synthesis markup](#).

The proper formation of the SSML file requires the timing information and the voice selection for a given segment. Therefore, everything needs to be returned to the correct segment of the source audio, and any change that's made along the pipeline should be snapped to these segments. Because the translation service doesn't produce timing information, the only source of timing details is the speech-to-text output. Additionally, the only elements that are transferable are the segment timing and the offsets that should be used to determine the duration of the target audio, in addition to the breaks and pauses in between. All this information should be carried forward to the remaining components in the pipeline, and the adjustment should be performed at the timing of the target audio as indicated previously.

Splitting the outputs by using the `voice` tag introduces a leading and trailing silence. You can define a value for the silence in the `voice` tag. In the following example, the gap between two speech segments is 450 ms (300 ms trailing and 150 ms leading). You need to account for these gaps when you calculate the final placement of the output. Here's a sample SSML element that defines some of these details:

XML

```

<speak version="1.0" xmlns="http://www.w3.org/2001/10/synthesis" xml:lang="en-US">
    <voice name="en-US-JennyNeural">
        <mstts:silence type="Leading-exact" value="100ms"/> <mstts:silence type="Sentenceboundary-exact" value="200ms"/> <mstts:silence type="Tailing-exact" value="300ms"/>
        If we're home schooling, the best we can do is roll with what each day brings and try to have fun along the way.
    </voice>
    <voice name="en-US-JennyNeural">
        <mstts:silence type="Leading-exact" value="150ms"/> <mstts:silence type="Sentenceboundary-exact" value="250ms"/> <mstts:silence type="Tailing-exact" value="350ms"/>
        A good place to start is by trying out the slew of educational apps that help children stay happy and smash their schooling at the same time.
    </voice>
</speak>

```

**Generate separate SSML files for each speaker.** The formatting of the SSML file allows you to define relative times for different voice segments. The value must be a positive value. For example, this XML defines a break after the spoken text is produced:

XML

```

<speak version="1.0" xmlns="http://www.w3.org/2001/10/synthesis"
xml:lang="en-US">
    <voice name="en-US-JennyNeural">
        Welcome to Microsoft Cognitive Services <break time="100ms" /> Text-to-Speech API.
    </voice>
</speak>

```

When multiple speakers speak concurrently, this technique doesn't work. One option is to generate a separate SSML file for every speaker (or in combinations that don't overlap) and overlay these files after the audio is produced.

**Include background audio effects in a separate channel.** The background noise in the source audio doesn't propagate through this pipeline, so the composition of the SSML file doesn't include it in the final output. You need to present any applicable background sound effects in a separate channel. You then add it to the final text-to-text output by using external tools.

**Add emotions and styles manually to enhance voice synthesis.** Emotions and tones aren't detected in source audio, so they're not propagated through the pipeline and don't appear in the SSML file. You need to add these details manually. It's best to add them in the source WebVTT output and present them as information carried over in the

WebVTT notes. The emotion and style details are added as notes like all the other information in the element and parsed when the SSML is produced. You can improve the synthesis of the voices either by presenting different models for different emotions or by using the Pro Custom Neural Voice capability.

## Timing placement and adjustment

Translating speech into different languages usually introduces a difference in the length of the content. You need to consider the placement and adjustment of the output speech to accommodate for this difference. Some of the ways to perform this adjustment are suggested here.

### Adjust the speech rate dynamically based on source and target language rates.

Matching the target speech to the duration of the source is a straightforward approach. This approach requires running the text-to-text once to determine the duration of the target text when it's converted to speech. The final rate used to produce the actual target audio reflects the ratio of the target text duration relative to the original source time for the same segment.

This approach is simple and produces a variable rate for each speech segment. The output sounds unnatural. The actual rate of the speaker in the source might change to present audible effects. You can improve the output by measuring the rate relative to a standard rate for the language and then adjusting the rate to reflect the source. You don't try to fit it into the same time span. To make this adjustment, you use the average word and character count per minute for normal speech in the source language. These counts are compared to the output of speech-to-text and used to identify the rate of the source speaker in each segment. You then use this rate in the target language. The natural rate of speakers in the target language should be incorporated in the text-to-text module. You can also set limits on the rate to make sure that the produced audio output is within controlled limits. The following code implements this process:

C#

```
internal static double GetRelativeTargetRate(PreProcessTTSInput input,
IOrchestratorLogger<TestingFrameworkOrchestrator> logger)
{
    string sourceLanguage = input.IdentifiedLocale.Split('-')[0];
    string targetLanguage = input.TargetLocale.Split('-')[0];
    double maxTargetRate =
input.PreProcessingStepConfig.MaxSpeechRate;
    double minTargetRate =
input.PreProcessingStepConfig.MinSpeechRate;
    if (SpeechRateLookup.Rate.ContainsKey(sourceLanguage) &&
SpeechRateLookup.Rate.ContainsKey(targetLanguage))
    {
```

```

    try
    {
        // Look up values from static table
        double sourceWordRate =
SpeechRateLookup.Rate[sourceLanguage].WordRate;
        double targetWordRate =
SpeechRateLookup.Rate[targetLanguage].WordRate;

        double sourceCharRate =
SpeechRateLookup.Rate[sourceLanguage].CharRate;
        double targetCharRate =
SpeechRateLookup.Rate[targetLanguage].CharRate;
        int sourceWordCount = input.LexicalText.Split(
").Length;
        int sourceCharCount = input.LexicalText.Length;
        // Calculate the rate of the source segment relative to
the nominal language rate
        double sourceRelativeWordRate = ((sourceWordCount /
input.Duration.TotalMinutes) / sourceWordRate);
        double sourceRelativeCharRate = ((sourceCharCount /
input.Duration.TotalMinutes) / sourceCharRate);
        double sourceAverageRelativeRate =
(sourceRelativeWordRate + sourceRelativeCharRate) / 2.0;
        // Calculate the ratio between the target and source
nominal rates
        double averagedLanguageRateRatio = (((targetWordRate /
sourceWordRate) + (targetCharRate / sourceCharRate)) / 2.0);
        // Scale the target rate based on the source/target
ratio and the relative rate of input with respect to the nominal source rate
        double relativeTargetRate = sourceAverageRelativeRate /
averagedLanguageRateRatio;
        relativeTargetRate = Math.Min(relativeTargetRate,
maxTargetRate);
        relativeTargetRate = Math.Max(relativeTargetRate,
minTargetRate);
        return relativeTargetRate;
    }
    catch (DivideByZeroException exception)
    {
        logger.LogError(exception.Message);
        return 1.0;
    }
}
else
{
    return 1.0;
}
}

```

Adjust the pauses and timing to achieve natural-sounding text-to-speech output. If the target language's speech duration is longer, use the pauses between speech

segments in the original audio to accommodate the extra duration. Doing so might slightly reduce the pause between sentences, but it ensures a more natural speech rate.

Align the produced audio with the original one. You can do this at the beginning, middle, or end of the original audio. For this example, assume that you align the audio segment in the target output with the middle of the original audio span.

You can estimate the speech signal duration from the number of tokens or characters in the target language. Alternatively, make a first pass with the text-to-speech service to determine the exact duration. Because the audio is centered, distribute the difference between the target and source audio duration evenly across the two pauses before and after the original audio.

If the target audio doesn't overlap with any other audio in the target, placing it is straightforward. The new offset in the target audio is the original offset minus half the difference between the target and source audio duration, as shown here:

$$\text{offset}_t^i = \text{offset}_s^i - (t_t^i - t_s^i)/2$$

However, multiple audio segments in the target audio might overlap. The goal of this process is to correct this problem, considering only the previous segment. Following are the possible scenarios in which this overlap can occur:

- $\text{offset}_t^i > \text{offset}_t^{(i-1)} + t_t^{(i-1)}$ . In this scenario, both segments fit within the pause assigned between them in the original audio with no issues.
- $\text{offset}_t^i = \text{offset}_t^{(i-1)} + t_t^{(i-1)}$ . In this scenario, there's no pause between two consecutive audio segments in the target language. Add a small offset (enough for one word in the language) to the  $\text{offset}_t^i$  ( $\Delta t_t$ ). ( $\Delta t_t$ ) is to avoid continuous audio in the target audio that isn't present in the original.
- $\text{offset}_t^i < \text{offset}_t^{(i-1)} + t_t^{(i-1)}$ . In this case, the two audio segments overlap. There are two possible scenarios:
  - $\text{offset}_s^{(i-1)} + t_s^{(i-1)} < \text{median} \{ \text{offset}_s^{(i-1)} + t_s^{(i-1)} \} \forall i$ . In this case, the pause is shorter than the nominal pause in the video. You can shift the translated text to resolve the problem. You can assume that the longer pauses make up for the overlap. Therefore,  $\text{offset}_t^i = \text{offset}_t^{(i-1)} + t_t^{(i-1)} + \Delta t_t$ .
  - $\text{offset}_s^{(i-1)} + t_s^{(i-1)} > \text{median} \{ \text{offset}_s^{(i-1)} + t_s^{(i-1)} \} \forall i$ . In this case, the pause is longer than the nominal pause in the video, but it doesn't fit the translated audio. You can use the rate to adjust the segment to be more appropriately timed. You can use this equation to calculate the new rate:  $\text{rate}_t^i = \text{rate}_t^{(i-1)} * (((\text{offset}_t^{(i-1)} + t_t^{(i-1)}) - \text{offset}_t^i) + t_t^i + \Delta t_t) / (t_t^i)$ .

Implementing this process might cause the last segment to run longer than the time of the original audio, especially if the video ends at the end of the original audio but the

target audio is longer. In this situation, you have two choices:

- Let the audio run longer.
- Add the full offset of the target audio to the pre-offset, to make the last offset<sub>t</sub><sup>n</sup> = offset<sub>s</sub><sup>n</sup> - (t<sub>t</sub><sup>n</sup> - t<sub>s</sub><sup>n</sup>). If this offset overlaps with the audio from the previous segment, use the rate modification described previously.

Aligning audio segments to the middle is appropriate in translation scenarios in which the target text is longer than the original. If the target is always faster than the original, it might be better to align the timing to the beginnings of the original audio segments.

C#

```
private List<PreProcessTTSInput>
CompensatePausesAnchorMiddle(List<PreProcessTTSInput> inputs)
{
    logger.LogInformation($"Performing Text to Speech Preprocessing on
{inputs.Count} segments.");
    foreach (var input in inputs)
    {
        ValidateInput(input);
        double rate = PreprocessTTSHelper.GetRelativeTargetRate(input,
logger);
        input.Rate = rate;
    }

    TimeSpan medianPause = PreprocessTTSHelper.CalculateMedianPause(inputs);

    // Setting nominal pause as average spoken duration of an English word
    TimeSpan nominalPause = new TimeSpan(0, 0, 0, 0, (int)(60.0 / 228.0 *
1000.0));

    string targetLanguage = inputs[0].TargetLocale.Split('-')[0];
    if (SpeechRateLookup.Rate.ContainsKey(targetLanguage))
    {
        nominalPause = new TimeSpan(0, 0, 0, 0, (int)(60.0 /
SpeechRateLookup.Rate[targetLanguage].WordRate * 1000));
    }

    TimeSpan previous_target_offset = new TimeSpan(0);
    TimeSpan previous_target_duration = new TimeSpan(0);

    TimeSpan previous_source_offset = new TimeSpan(0);
    TimeSpan previous_source_duration = new TimeSpan(0);

    foreach (PreProcessTTSInput source_segment in inputs)
    {
        var source_duration = source_segment.Duration;
        var source_offset = source_segment.Offset;

        var target_duration =
```

```

PreprocessTTSShader.GetTargetDuration(source_segment);
    var target_offset = source_offset - (target_duration -
source_duration) / 2;

        if (target_offset == previous_target_offset +
previous_target_duration)
        {
            target_offset += nominalPause;
            source_segment.HumanInterventionRequired = true;
            source_segment.HumanInterventionReason = "After translation this
segment just fit in the space available, a small pause was added before it
to space it out.";
        }
        else if (target_offset < previous_target_offset +
previous_target_duration)
        {
            source_segment.HumanInterventionRequired = true;
            var previous_pause = source_segment.Offset -
previous_source_offset + previous_source_duration;
            if (previous_pause < medianPause)
            {
                target_offset = previous_target_offset +
previous_target_duration + nominalPause;
                source_segment.HumanInterventionReason = "After translation
this segment overlapped with previous segment, but since the pause before
this segment in the source was relatively small, this segment was NOT sped
up and was placed right after the previous segment with a small pause";
            }
            else
            {
                var target_rate = source_segment.Rate *
((previous_target_offset + previous_target_duration - target_offset) +
target_duration + nominalPause) / target_duration;
                source_segment.Rate = target_rate;
                target_duration =
PreprocessTTSShader.GetTargetDuration(source_segment);
                target_offset = previous_target_offset +
previous_target_duration + nominalPause;
                source_segment.HumanInterventionReason = "After translation
this segment overlapped with previous segment, also the pause this segment
in the source was relatively large (and yet the translated segment did not
fit), hence this segment was sped up and placed after the previous segment
with a small pause";
            }
        }
    }

    source_segment.Offset = target_offset;

    previous_target_offset = target_offset;
    previous_target_duration = target_duration;

    previous_source_offset = source_offset;
    previous_source_duration = source_duration;
}

```

```
    return inputs;  
}
```

## Creating WebVTT files for captions and human editing

Web Video Text Tracks Format (WebVTT) is a format for displaying timed text tracks, like subtitles or captions. It's used for adding text overlays to video. The WebVTT file presents the output of the pipeline for human consumption. It's also used for the captions file. This section describes some considerations for using WebVTT in this pipeline.

**Include information in the notes of the WebVTT file.** The WebVTT file should contain all the information that's required to process the various modules. In addition to timing and textual information, it includes the language ID, the user ID, and flags for human intervention. To include these details in the WebVTT file and keep it compatible for use as a captions/subtitles file, use the `NOTE` section of each segment in the WebVTT file.

The following code shows the recommended format:

```
C#  
  
NOTE  
{  
  "Locale": "en-US",  
  "Speaker": null,  
  "HumanIntervention": true,  
  "HumanInterventionReasons": [  
    {  
      "Word": "It's",  
      "WordIndex": 3,  
      "ContentionType": "WrongWord"  
    }  
  ]  
}  
  
1  
00:00:00.0600000-->00:00:04.6000000  
- Hello there!  
- It's a beautiful day
```

**Ensure that subtitles display correctly on various screen sizes.** When you produce captions files, you need to consider the number of words and lines of text to display on the screen. You need to consider this specifically for each individual use case during the production of the WebVTT file. Humans in the loop review these files and use prompts

to resolve problems. The file also contains metadata that other modules later in the pipeline need in order to function.

To ensure that subtitles appear properly, you need to divide longer segments (transcribed or translated results) into multiple VTTCues. You need to preserve the notes and other details so that they still point to the right context and ensure that the pipeline doesn't interrupt the translation or the text-to-text placement of text.

When you divide the original text segment into multiple VTTCues, you also need to consider the timing for each division. You can use the word-level time stamps that are produced during transcription for this purpose. Because the WebVTT file is open for modification, you can update these time stamps manually, but the process creates considerable maintenance overhead.

## Identify potential for human intervention

The dubbing pipeline includes three important modules: speech-to-text, translation, and text-to-speech. Errors can occur in each of these modules, and errors can propagate and become more pronounced later in the pipeline. We strongly recommend that you include human intervention to review and rectify results to ensure an acceptable level of quality. You can save significant time by identifying the specific areas that require intervention. This section describes potential errors in each module that require manual intervention.

### Speech-to-text

The output of speech-to-text can be affected by the audio itself, the background noise, and the language model. You can improve audio content by customizing the language model to better suit the input. Consider the following options, depending on which points of human intervention are available in your scenario:

**Identify transcription errors.** The speech-to-text system can sometimes produce transcription errors. It provides multiple transcription options, each with a confidence score. These scores can help you determine the most accurate transcription. The *n*-best list contains the most probable hypotheses for a given speech input. The top hypothesis in this list has the highest score, indicating its likelihood of being the correct transcription. For example, a score of 0.95 indicates 95% confidence in the accuracy of the top hypothesis. By comparing the scores and differences among the outputs, you can identify discrepancies that require human intervention. You need to consider the threshold and majority voting when you review the *n*-best list:

*Set a higher threshold for improved quality in the dubbing pipeline.* The threshold plays a crucial role in identifying potential errors in the  $n$ -best results. It determines the number of  $n$ -best results that are compared during the process. A higher threshold returns more  $n$ -best results for comparison, resulting in a higher quality of intervention. Setting the threshold to zero limits the selection of candidates but provides higher confidence, as compared to the transcription produced by the speech-to-text system.

*Balance recall and precision.* When you use majority voting, select the lexical form for each candidate that falls within the threshold. Next, determine the insertions, deletions, and replacements in relation to the output phrase. You need to consider these points of contention for highlighting, but highlighting all of them prioritizes recall at the expense of lower precision. To enhance precision, order the insertions, deletions, and replacements based on a normalized average score. The count of these errors, relative to the total number of candidates minus one, serves as a weight. You can set a threshold to calibrate the desired precision level. A weight of 0 prioritizes recall, indicating that the error is present in at least one candidate. Conversely, a weight of 1 prioritizes precision, implying that the error appears in all candidates. In this pipeline, you need to find the appropriate balance between recall and precision.

C#

```
public string EvaluateCorrectness(SpeechCorrectnessInput input)
{
    var speechOutputSegments = input.Input;
    var ConfidenceThreshold = input.Configuration.ConfidenceThreshold;
    var errorOccurrenceThreshold = input.Configuration.OccurrenceThreshold;
    var speechCorrectnessOutputSegments = new
List<SpeechCorrectnessOutputSegment>();

    foreach (var speechOutputSegment in speechOutputSegments)
    {
        // Filter the candidates within the lower confidence candidate threshold
        var filteredCandidates =
GetFilteredCandidates(speechOutputSegment, ConfidenceThreshold);

        ICollection<SpeechPointOfContention> interventions = null;
        if (filteredCandidates.Count > 0)
        {
            // Calculate the possible reasons for needed intervention based on the
            // selected text, the filtered candidates, and the error threshold
            interventions =
CalculateInterventionReasons(speechOutputSegment.LexicalText,
filteredCandidates, errorOccurrenceThreshold);
        }
        // Build the speech correctness segment
        SpeechCorrectnessOutputSegment correctionSegment = new
SpeechCorrectnessOutputSegment()
        {
            SegmentID = speechOutputSegment.SegmentID
        }
    }
}
```

```

    };
    if (interventions != null && interventions.Count > 0)
    {
        correctionSegment.InterventionNeeded = true;
        correctionSegment.InterventionReasons = interventions;
    }
    else
    {
        correctionSegment.InterventionNeeded = false;
        correctionSegment.InterventionReasons = null;
    }
    speechCorrectnessOutputSegments.Add(correctionSegment);
}

return JsonConvert.SerializeObject(speechCorrectnessOutputSegments);
}

private ICollection<SpeechPointOfContention>
CalculateInterventionReasons(string selectedText,
ICollection<SpeechCandidate> candidates, double errorOccurrenceThreshold)
{
    ICollection<ICollection<ICollection<SpeechPointOfContention>>>
allPossiblePointsOfContention = new
List<ICollection<ICollection<SpeechPointOfContention>>>();

foreach (var candidate in candidates)
{
    // Get points of contention for all speech candidates in this
    // segment, compared to the text selected
    var candidateSentence = candidate.LexicalText.Split(' ');
    var segmentSentence = selectedText.Split(' ');
    var editDistanceResults = CalculateEditDistance(candidateSentence,
segmentSentence);
    var allPossibleCandidatePointsOfContention =
GetAllPossiblePointsOfContention(
        candidateSentence,
        segmentSentence,
        new Stack<SpeechPointOfContention>(),
        editDistanceResults);

    allPossiblePointsOfContention.Add(allPossibleCandidatePointsOfContention);
}
    // Calculate which combination of all possible points of contention will
    // lead to the highest count for the selected points of contention across all
    // candidates
    var bestCandidatePointsOfContention =
GetBestContentionCombination(allPossiblePointsOfContention);

    // Keep count of occurrences of all points of contention across the
    // candidates
    var candidatePOCCounts = new Dictionary<SpeechPointOfContention, int>();
    foreach (var pointsOfContention in bestCandidatePointsOfContention)
    {
        CountPointsOfContention(pointsOfContention, candidatePOCCounts);
    }
}

```

```

var filteredPointsOfContention = new List<SpeechPointOfContention>();
foreach (var pocToCount in candidatePOCCCounts)
{
    // For each point of contention, calculate its rate of occurrence
    // relative to the number of candidates
    var occurrenceRate = (double)pocToCount.Value / candidates.Count;
    // If the rate of occurrence is within the threshold, count the
    point of contention as an error that requires intervention
    if (occurrenceRate >= errorOccurrenceThreshold)
    {
        filteredPointsOfContention.Add(pocToCount.Key);
    }
}
return filteredPointsOfContention;
}

```

**Implement well-spaced language transition detection.** In language identification, detecting a transition from one language to another involves identifying multiple words. When the transition occurs within a certain number of tokens from the previous text, it's flagged. This assumption is based on the assumption that well-spaced segments are more likely to be detected accurately. The aim is to avoid false negatives, where a language change occurs but the speech-to-text system fails to detect it before shifting back to the original language.

**Flag unidentified speaker IDs.** When you use speech-to-text with diarization, the system might label some speaker IDs as `Unidentified`. This labeling typically occurs when a speaker speaks for a short time. We recommend that you flag these labels for human editing.

## Translation

The BLEU score is a commonly used metric for evaluating the quality of translated output. However, the BLEU score has limitations. It relies on comparing the machine-translated output with human-generated reference translations to measure similarity. This metric doesn't consider important elements like fluency, grammar, and the preservation of meaning.

You can use an alternative method to assess translation quality. It involves performing a bidirectional translation and comparing the results. First, translate the input segment from the source language to the target language. Next, reverse the process and translate back from the target language to the source language. This process generates two strings: the source segment ( $S_i$ ) and the expected value of the source ( $E(S_i)$ ). By comparing the two strings, you can figure out the number of insertions and deletions.

If the number of insertions and deletions exceeds a predefined threshold, the statement is flagged for further examination. If the number falls below the threshold, the segment Si is considered a well-translated match. In cases where differences are found, it might be beneficial to highlight them in the source language. This approach helps draw attention to specific terms that need correction, rather than showing that the entire segment requires human verification.

## Text-to-speech

The text-to-speech highlighting is mostly associated with the changed rate and with shifts in placement that are beyond the gaps in the source audio. The locations where there are alterations of rate or placement, as pointed out in the text-to-text section, are highlighted.

## Contributors

*This article is maintained by Microsoft. It was originally written by the following contributors.*

Principal author:

- [Nayer Wanasyan](#) | Principal Software Engineer

Other contributors:

- [Mick Alberts](#) | Technical Writer
- [Vivek Chettiar](#) | Software Engineer II
- [Bernardo Chuecos Rincon](#) | Software Engineer II
- [Pratyush Mishra](#) | Principal Engineering Manager

*To see non-public LinkedIn profiles, sign in to LinkedIn.*

## Next steps

- [Sample implementation of offline audio dubbing](#)
- [What is Azure Cognitive Services?](#)
- [Language and voice support for the Speech service](#)

## Related resources

- [Choose an Azure Cognitive Services technology](#)

- Implement custom speech-to-text

# Monitor Media Services

Article • 02/14/2023

When you have applications and business processes relying on Azure resources, you want to monitor those resources for their availability, performance, and operation. This article describes the monitoring data generated by Media Services and how you can use the features of Azure Monitor to analyze and alert on this data.

## Azure Monitor

Media Services creates monitoring data using [Azure Monitor](#), which is a full stack monitoring service in Azure that provides a complete set of features to monitor your Azure resources in addition to resources in other clouds and on-premises.

Start with reading the article [Monitoring Azure resources with Azure Monitor](#), which describes the following concepts:

- What is Azure Monitor?
- Costs associated with monitoring
- Monitoring data collected in Azure
- Configuring data collection
- Standard tools in Azure for analyzing and alerting on monitoring data

## Media Services monitoring data

Media Services collects the same kinds of monitoring data as other Azure resources that are described in [Monitoring data from Azure resources](#).

All data collected by Azure Monitor fits into one of two fundamental types: metrics and logs. With these two types you can:

- Visualize and analyze the metrics data using Metrics Explorer.
- Monitor Media Services diagnostic logs and create alerts and notifications for them.
- You can send or stream logs to:
  - Azure Storage
  - Azure Event Hubs
  - Log Analytics
  - Use third-party services

# Collection and routing

*Platform metrics* and the *Activity log* are collected and stored automatically, but can be routed to other locations by using a diagnostic setting.

*Resource Logs* are **not** collected and stored until you create a diagnostic setting and route them to one or more locations.

See the article [Create diagnostic setting to collect platform logs and metrics in Azure](#) for the detailed process of creating a diagnostic setting.

## Media Services metrics

Media Services metrics are collected at regular intervals whether or not the value changes.

### Metric types

Metrics available for Media Services are:

- [Media Services account metrics, including Key Delivery](#)
- [Live event metrics](#)
- [Streaming endpoint metrics](#)

### Analyzing metrics

You can analyze metrics for Media Services along with metrics from other Azure services using Metrics Explorer. See [Getting started with Azure Metrics Explorer](#) for details on using this tool.

## Media Services logs

### Activity logs

The [Activity log](#) is a platform log that provides insight into subscription-level events. You can view it independently or route it to Azure Monitor Logs, where you can do much more complex queries using Log Analytics.

### Resource logs

Resource logs provide rich and frequent data about the operation of an Azure resource. For more information, see [How to collect and consume log data from your Azure resources](#).

Media Services supports the following resource logs: [Microsoft.Media/mediaservices](#)

## Media Services diagnostic logs

Some things that you can examine with diagnostic logs are:

- The number of licenses delivered by DRM type
- The number of licenses delivered by policy
- The latency on key delivery requests
- The number of unauthorized license requests from clients

## Analyzing logs

Data in Azure Monitor Logs is stored in tables where each table has its own set of unique properties.

All resource logs in Azure Monitor have the same fields followed by service-specific fields. The common schema is outlined in [Azure Monitor resource log schema](#).

## Alerts

Azure Monitor alerts proactively notify you when important conditions are found in your monitoring data. They allow you to identify and address issues in your system. You can set alerts on metrics, logs, and the activity log. For more information, see [Azure Monitor Alerts overview](#).

## Schemas

For detailed description of the top-level diagnostic logs schema, see [Supported services, schemas, and categories for Azure Diagnostic Logs](#).

## Media Account Health

Name	Description
TimeGenerated	The timestamp (UTC) of when the event was generated.

Name	Description
OperationName	The name of the operation that triggered the event.
Level	Message level. Possible values are Informational, Warning, Error, Critical and Verbose.
Location	Location of the service sending the log.
EventCode	The event code.
EventMessage	The event status message.

## Key Delivery

Name	Description
TimeGenerated	The timestamp (UTC) of when the event was generated.
OperationName	The name of the operation that triggered the event.
OperationVersion	Azure Media Services operation version.
ResultType	Azure Media Services operation result type.
ResultSignature	Azure Media Services operation result signature.
DurationMs	Azure Media Services operation duration in milliseconds.
Level	Message level. Possible values are Informational, Warning, Error, Critical and Verbose.
Location	Location of the service sending the log.
RequestId	Id of the request.
KeyType	Could be one of the following values: Clear (no encryption), FairPlay, PlayReady, or Widevine.
KeyId	The ID of the requested key.
TokenType	The token type.
PolicyName	The Azure Resource Manager name of the policy.
StatusMessage	The status message.

## Sample key delivery log

## JSON

```
{  
    "time": "2019-01-11T17:59:10.4908614Z",  
    "resourceId": "/SUBSCRIPTIONS/00000000-0000-0000-0000-  
0000000000/RESOURCEGROUPS/SBKEY/PROVIDERS/MICROSOFT.MEDIA/MEDIASERVICES/SBDN  
STEST",  
    "operationName": "MICROSOFT.MEDIA/MEDIASERVICES/CONTENTKEYS/READ",  
    "operationVersion": "1.0",  
    "category": "KeyDeliveryRequests",  
    "resultType": "Succeeded",  
    "resultSignature": "OK",  
    "durationMs": 315,  
    "identity": {  
        "authorization": {  
            "issuer": "http://testacs",  
            "audience": "urn:test"  
        },  
        "claims": {  
            "urn:microsoft:azure:mediaservices:contentkeyidentifier":  
"3321e646-78d0-4896-84ec-c7b98eddfca5",  
            "iss": "http://testacs",  
            "aud": "urn:test",  
            "exp": "1547233138"  
        }  
    },  
    "level": "Informational",  
    "location": "uswestcentral",  
    "properties": {  
        "requestId": "b0243468-d8e5-4edf-a48b-d408e1661050",  
        "keyType": "Clear",  
        "keyId": "3321e646-78d0-4896-84ec-c7b98eddfca5",  
        "policyName": "56a70229-82d0-4174-82bc-e9d3b14e5dbf",  
        "tokenType": "JWT",  
        "statusMessage": "OK"  
    }  
}
```

## Live Events

Name	Description
TimeGenerated	The timestamp (UTC) when the event was generated.
OperationName	The name of the operation that triggered the event.
Level	Message level. Possible values are Informational, Warning, Error, Critical and Verbose.
Location	Location of the service sending the event.

Name	Description
Properties	Operation details.

## Sample live event log

JSON

```
[
  {
    "TimeGenerated": "2022-10-11T06:02:13.4730825Z",
    "OperationName": "LIVEEVENTS/INGESTBEGIN",
    "Level": "Informational",
    "Location": "westcentralus",
    "Properties": {
      "liveEventName": "CONTOSOLIVE",
      "streamName": "1234",
      "remoteIP": "10.0.0.xxx",
      "remotePort": "35091"
    }
  },
  {
    "TimeGenerated": "2022-10-11T06:02:19.8229491Z",
    "OperationName": "LIVEEVENTS/STREAMINFO",
    "Level": "Informational",
    "Location": "westcentralus",
    "Properties": {
      "liveEventName": "CONTOSOLIVE",
      "remoteIP": "10.0.0.xxx",
      "remotePort": "35091",
      "trackName": "audio_160000",
      "trackType": "audio",
      "bitrate": 160000,
      "timestamp": 66,
      "timescale": 1000,
      "resolution": "n/a"
    }
  },
  {
    "TimeGenerated": "2022-10-11T06:04:41.1375866Z",
    "OperationName": "LIVEEVENTS/INGESTEND",
    "Level": "Informational",
    "Location": "westcentralus",
    "Properties": {
      "liveEventName": "CONTOSOLIVE",
      "streamName": "1234",
      "remoteIP": "10.0.0.xxx",
      "remotePort": "35091",
      "resultCode": "MPE_CLIENT_TERMINATED_SESSION"
    }
  },
  {
    "TimeGenerated": "2022-10-11T06:07:01.0446756Z",
    "OperationName": "LIVEEVENTS/INGESTDISCONTINUITY",
    "Level": "Warning",
    "Location": "westcentralus",
    "Properties": {
      "liveEventName": "CONTOSOLIVE",
      "trackName": "audio",
      "timestamp": 156777,
      "disco
      ntinuityGap": 12605
    }
  }
]
```

## Streaming Endpoints

Name	Description
TimeGenerated	The timestamp (UTC) when the event was generated.
OperationName	The name of the operation that triggered the event.
OperationVersion	Azure Media Services operation version.
Level	Message level. Possible values are Informational, Warning, Error, Critical and Verbose.
Location	Location of the service sending the event.
ClientIP	IP address of the client.
URL	The streaming URL from Azure Media Services.
Status	Status code of the request.

## Sample streaming endpoint log

JSON

```
[
  {
    "time": "2022-09-30T07:40:06.1524833Z",
    "resourceId": "/SUBSCRIPTIONS/00000000-0000-0000-0000-000000000001/RESOURCEGROUPS/CONTOSORG/PROVIDERS/MICROSOFT.MEDIA/MEDIASERVICE S/CONTOSOMEDIA/STREAMINGENDPOINTS/DEFAULT",
    "operationName": "MICROSOFT.MEDIA/MEDIASERVICES/STREAMINGENDPOINTS/GET",
    "category": "StreamingEndpointRequests",
    "level": "Informational",
    "location": "uswc1",
    "properties": {
      "ClientIP": "10.0.0.1",
      "URL": "https://cdn--contosomedia-uswc.streaming.media.azure.net:443/00000000-0000-0000-0000-000000000000/contoso.ism/QualityLevels(127999)/Fragments(aac_eng_2_127999_2_1=20053333,format=mpd-time-csf)",
      "Status": "200"
    },
    "operationVersion": "1.0"
  }
]
```

## How-tos

- [Route Media Services metrics to storage](#)

- Audit Media Services control plane logs
- Autoscaling premium streaming endpoints
- Create an Azure Monitor dashboard and alerts

# Content protection with dynamic encryption and key delivery

Article • 01/10/2023



Media Services API v3

## ⚠️ Warning

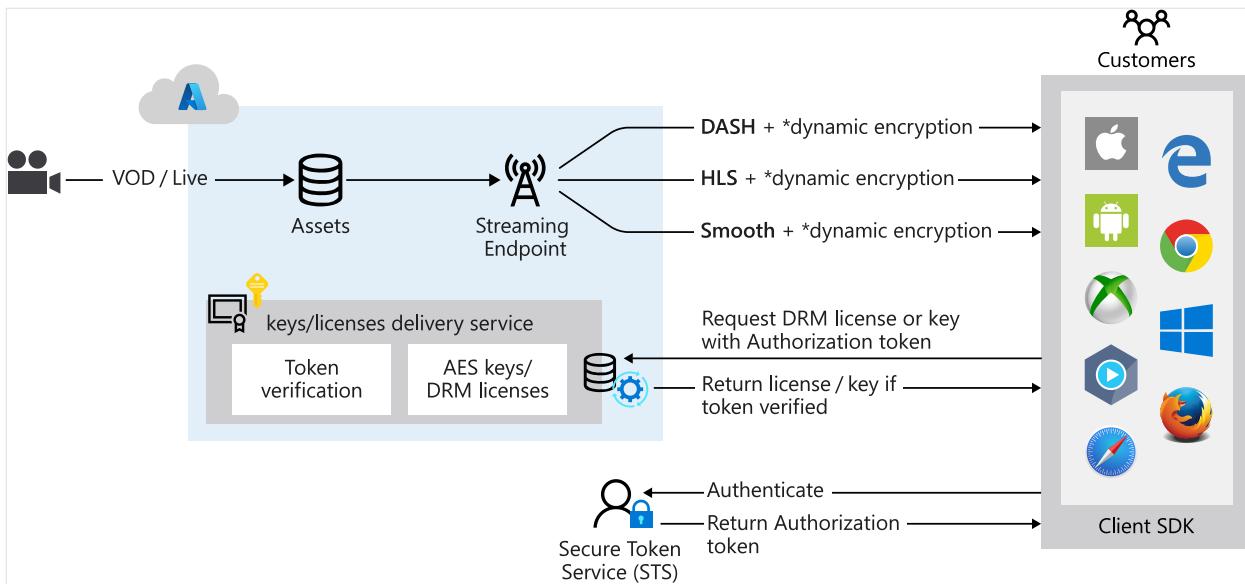
Azure Media Services will be retired June 30th, 2024. For more information, see the [AMS Retirement Guide](#).

Use Azure Media Services to secure your media from the time it leaves your computer all the way through storage, processing, and delivery. With Media Services, you can deliver your live and on-demand content encrypted dynamically with Advanced Encryption Standard (AES-128) or any of the three major digital rights management (DRM) systems: Microsoft PlayReady, Google Widevine, and Apple FairPlay.

FairPlay Streaming is an Apple technology that is only available for video transferred over HTTP Live Streaming (HLS) on iOS devices, in Apple TV, and in Safari on macOS. Media Services also provides a service for delivering AES keys and DRM (PlayReady, Widevine, and FairPlay) licenses to authorized clients. If content is encrypted with an AES clear key and is sent over HTTPS, it is not in clear until it reaches the client.

In Media Services v3, a content key is associated with Streaming Locator (see [this example](#)). If using the Media Services key delivery service, you can let Azure Media Services generate the content key for you. The content key should be generated yourself if you're using your own key delivery service, or if you need to handle a high availability scenario where you need to have the same content key in two data centers.

When a stream is requested by a player, Media Services uses the specified key to dynamically encrypt your content by using AES clear key or DRM encryption. To decrypt the stream, the player requests the key from Media Services key delivery service or the key delivery service you specified. To decide if the user is authorized to get the key, the service evaluates the content key policy that you specified for the key.



You can use the REST API, or a Media Services client library to configure authorization and authentication policies for your licenses and keys.

Widevine is not available in the GovCloud region.

**ⓘ Note**

Media services will be enforcing TLS 1.2 for all requests to KeyDelivery, RESTv2, Streaming Endpoint and Live Event streaming origins. Accounts with existing TLS 1.0 or 1.1 usage will be exempt from this enforcement. If you wish to enforce TLS 1.2 for all your requests to these media services endpoints, please contact AMS support.

## Browsers that support DRM clients

Common browsers support the following DRM clients:

Browser	Encryption
Chrome	Widevine
Microsoft Edge, Internet Explorer 11	PlayReady
Firefox	Widevine
Opera	Widevine
Safari	FairPlay

# Controlling content access

You can control who has access to your content by configuring the content key policy. Media Services supports multiple ways of authorizing users who make key requests. The client (player) must meet the policy before the key can be delivered to the client. The content key policy can have *open* or *token* restriction.

An open-restricted content key policy may be used when you want to issue a license to anyone without authorization. For example, if your revenue is ad-based and not subscription-based.

With a token-restricted content key policy, the content key is sent only to a client that presents a valid JWT token or a simple web token (SWT) in the license/key request. This token must be issued by an STS.

## Using Azure AD as an STS

You can use Azure AD as an STS. It must be configured to create a token signed with the specified key and issue claims that you specified in the token restriction configuration. The Media Services license/key delivery service returns the requested license or key to the client if both of these conditions exist:

- The token is valid.
- The claims in the token match those configured for the license or key.

When you configure the token-restricted policy, you must specify the primary verification key, issuer, and audience parameters. The primary verification key contains the key that the token was signed with. The issuer is the STS that issues the token. The audience, sometimes called *scope*, describes the intent of the token or the resource that the token authorizes access to. The Media Services license/key delivery service validates that the values in the token match the values in the template.

## Token replay prevention

The *Token Replay Prevention* feature allows you to set a limit on how many times the same token can be used to request a key or a license. You can add a claim of type `urn:microsoft:azure:mediaservices:maxuses` in the token, where the value is the number of times the token can be used to acquire a license or key. All subsequent requests with the same token to Key Delivery will return an unauthorized response.

## Considerations

- You must have control over token generation. The claim needs to be placed in the token itself.
- When using this feature, requests with tokens whose expiry time is more than one hour away from the time the request is received are rejected with an unauthorized response.
- Tokens are uniquely identified by their signature. Any change to the payload (for example, update to the expiry time or the claim) changes the signature of the token and it will count as a new token that Key Delivery hasn't come across before.
- Playback fails if the token has exceeded the `maxuses` value.
- It can be used for all existing protected content (only the token issued needs to be changed).
- It works with both JWT and SWT.

## Using a custom STS

You might choose to use a custom STS to provide tokens. Reasons include:

- Your identity provider (IDP) doesn't support STS.
- You might need more flexible or tighter control to integrate the STS with your subscriber billing system.

For example, an [OTT](#) service operator might offer multiple subscriber packages, such as premium, basic, and sports. The operator might want to match the claims in a token with a subscriber's package so that only the contents in a specific package are made available. In this case, a custom STS provides the needed flexibility and control.

- To include custom claims in the token to select between different `ContentKeyPolicyOptions` with different DRM license parameters, for example, a subscription license versus a rental license.
- To include a claim representing the content key identifier of the key that the token grants access to.

When you use a custom STS, two changes must be made:

- When you configure a license delivery service for an asset, you need to specify the security key used for verification by the custom STS instead of the current key from Azure AD.
- When a JWT token is generated, a security key is specified instead of the private key of the current X509 certificate in Azure AD.

There are two types of security keys:

- Symmetric key: The same key is used to generate and to verify a JWT.
- Asymmetric key: A public-private key pair in an X509 certificate is used with a private key to encrypt/generate a JWT and with the public key to verify the token.

#### ⓘ Note

If you use .NET Framework/C# as your development platform, the X509 certificate used for an asymmetric security key must have a key length of at least 2048. This key length is a requirement of the class `System.IdentityModel.Tokens.X509AsymmetricSecurityKey` in .NET Framework. Otherwise, the following exception is thrown: IDX10630: The '`System.IdentityModel.Tokens.X509AsymmetricSecurityKey`' for signing can't be smaller than '2048' bits.

## Using a license/key delivery service other than Media Services

You can edit key policy templates if you want to use a different license/key delivery service.

## How-tos, tutorials and samples

[.Net Digital Rights Management sample](#) shows you how to implement a multi-DRM system with Media Services v3 by using .NET.

There are additional content protection samples available for Node.JS and Python:

Node.JS	Python	Description
<a href="#">Node.JS Upload and stream HLS and DASH with PlayReady and Widevine DRM</a>	<a href="#">Python Upload and stream HLS and DASH with PlayReady and Widevine DRM</a>	Demonstrates how to encode and stream using Widevine and PlayReady DRM
<a href="#">Node.JS Basic Playready DRM content protection and streaming</a>	<a href="#">Python Basic Playready DRM content protection and streaming</a>	Demonstrates how to encode and stream using PlayReady DRM
<a href="#">Node.JS Basic Widevine DRM content protection and streaming</a>	<a href="#">Python Basic Widevine DRM content protection and streaming</a>	Demonstrates how to encode and stream using Widevine DRM

# Get help and support

You can contact Media Services with questions or follow our updates by one of the following methods:

- [Q & A](#)
- [Stack Overflow](#). Tag questions with `azure-media-services`.
- [@MSFTAzureMedia](#) or use [@AzureSupport](#) to request support.
- Open a support ticket through the Azure portal.

# Azure security baseline for Media Services

Article • 09/20/2023

This security baseline applies guidance from the [Microsoft cloud security benchmark version 1.0](#) to Media Services. The Microsoft cloud security benchmark provides recommendations on how you can secure your cloud solutions on Azure. The content is grouped by the security controls defined by the Microsoft cloud security benchmark and the related guidance applicable to Media Services.

You can monitor this security baseline and its recommendations using Microsoft Defender for Cloud. Azure Policy definitions will be listed in the Regulatory Compliance section of the Microsoft Defender for Cloud portal page.

When a feature has relevant Azure Policy Definitions, they are listed in this baseline to help you measure compliance with the Microsoft cloud security benchmark controls and recommendations. Some recommendations may require a paid Microsoft Defender plan to enable certain security scenarios.

## ⓘ Note

Features not applicable to Media Services have been excluded. To see how Media Services completely maps to the Microsoft cloud security benchmark, see the [full Media Services security baseline mapping file](#).

## Security profile

The security profile summarizes high-impact behaviors of Media Services, which may result in increased security considerations.

Service Behavior Attribute	Value
Product Category	Media
Customer can access HOST / OS	No Access
Service can be deployed into customer's virtual network	True
Stores customer content at rest	True

# Network security

For more information, see the [Microsoft cloud security benchmark: Network security](#).

## NS-1: Establish network segmentation boundaries

### Features

#### Virtual Network Integration

**Description:** Service supports deployment into customer's private Virtual Network (VNet). [Learn more](#).

Supported	Enabled By Default	Configuration Responsibility
False	Not Applicable	Not Applicable

**Configuration Guidance:** This feature is not supported to secure this service.

## NS-2: Secure cloud services with network controls

### Features

#### Azure Private Link

**Description:** Service native IP filtering capability for filtering network traffic (not to be confused with NSG or Azure Firewall). [Learn more](#).

Supported	Enabled By Default	Configuration Responsibility
True	False	Customer

**Configuration Guidance:** Deploy private endpoints to establish a private access point for the resources.

**Reference:** [Overview of using Azure Private Link with Azure Media Services](#)

#### Disable Public Network Access

**Description:** Service supports disabling public network access either through using service-level IP ACL filtering rule (not NSG or Azure Firewall) or using a 'Disable Public

Network Access' toggle switch. [Learn more](#).

Supported	Enabled By Default	Configuration Responsibility
True	False	Customer

**Configuration Guidance:** Disable public network access either using the service-level IP ACL filtering rule or a toggling switch for public network access.

Reference: [Public network access flag](#)

## Identity management

For more information, see the [Microsoft cloud security benchmark: Identity management](#).

### IM-1: Use centralized identity and authentication system

#### Features

##### Azure AD Authentication Required for Data Plane Access

**Description:** Service supports using Azure AD authentication for data plane access.

[Learn more](#).

Supported	Enabled By Default	Configuration Responsibility
True	False	Customer

**Configuration Guidance:** Use Azure Active Directory (Azure AD) as the default authentication method to control your data plane access. Azure AD can be combined with Azure Media Services Key delivery and content protection services to secure playback of media content. Samples and guidance can be found in the documentation.

Reference: [End-to-End content protection using Azure AD](#)

##### Local Authentication Methods for Data Plane Access

**Description:** Local authentications methods supported for data plane access, such as a local username and password. [Learn more](#).

Supported	Enabled By Default	Configuration Responsibility
False	Not Applicable	Not Applicable

**Configuration Guidance:** This feature is not supported to secure this service.

## IM-3: Manage application identities securely and automatically

### Features

#### Managed Identities

**Description:** Data plane actions support authentication using managed identities. [Learn more.](#)

Supported	Enabled By Default	Configuration Responsibility
True	True	Microsoft

**Configuration Guidance:** No additional configurations are required as this is enabled on a default deployment.

**Reference:** [Managed identities](#)

### Service Principals

**Description:** Data plane supports authentication using service principals. [Learn more.](#)

Supported	Enabled By Default	Configuration Responsibility
True	False	Customer

**Configuration Guidance:** There is no current Microsoft guidance for this feature configuration. Please review and determine if your organization wants to configure this security feature.

**Reference:** [Access the Azure Media Services with Azure AD authentication](#)

## IM-7: Restrict resource access based on conditions

## Features

### Conditional Access for Data Plane

**Description:** Data plane access can be controlled using Azure AD Conditional Access Policies. [Learn more.](#)

Supported	Enabled By Default	Configuration Responsibility
False	Not Applicable	Not Applicable

**Configuration Guidance:** This feature is not supported to secure this service.

### IM-8: Restrict the exposure of credential and secrets

## Features

### Service Credential and Secrets Support Integration and Storage in Azure Key Vault

**Description:** Data plane supports native use of Azure Key Vault for credential and secrets store. [Learn more.](#)

Supported	Enabled By Default	Configuration Responsibility
False	Not Applicable	Not Applicable

**Configuration Guidance:** This feature is not supported to secure this service.

## Privileged access

*For more information, see the [Microsoft cloud security benchmark: Privileged access](#).*

### PA-1: Separate and limit highly privileged/administrative users

## Features

### Local Admin Accounts

**Description:** Service has the concept of a local administrative account. [Learn more](#).

Supported	Enabled By Default	Configuration Responsibility
False	Not Applicable	Not Applicable

**Configuration Guidance:** This feature is not supported to secure this service.

## PA-7: Follow just enough administration (least privilege) principle

### Features

#### Azure RBAC for Data Plane

**Description:** Azure Role-Based Access Control (Azure RBAC) can be used to manage access to service's data plane actions. [Learn more](#).

Supported	Enabled By Default	Configuration Responsibility
True	False	Customer

**Configuration Guidance:** Currently, Azure Media Services does not define any custom roles specific to the service. However, you can define custom roles based on your needs for Azure RBAC based access control.

**Reference:** [Azure role-based access control \(Azure RBAC\) for Media Services accounts](#)

## PA-8: Determine access process for cloud provider support

### Features

#### Customer Lockbox

**Description:** Customer Lockbox can be used for Microsoft support access. [Learn more](#).

Supported	Enabled By Default	Configuration Responsibility
False	Not Applicable	Not Applicable

**Configuration Guidance:** This feature is not supported to secure this service.

## Data protection

For more information, see the [Microsoft cloud security benchmark: Data protection](#).

### DP-1: Discover, classify, and label sensitive data

#### Features

##### Sensitive Data Discovery and Classification

**Description:** Tools (such as Azure Purview or Azure Information Protection) can be used for data discovery and classification in the service. [Learn more](#).

Supported	Enabled By Default	Configuration Responsibility
False	Not Applicable	Not Applicable

**Configuration Guidance:** This feature is not supported to secure this service.

### DP-3: Encrypt sensitive data in transit

#### Features

##### Data in Transit Encryption

**Description:** Service supports data in-transit encryption for data plane. [Learn more](#).

Supported	Enabled By Default	Configuration Responsibility
True	True	Microsoft

**Configuration Guidance:** No additional configurations are required as this is enabled on a default deployment.

### DP-4: Enable data at rest encryption by default

#### Features

## Data at Rest Encryption Using Platform Keys

**Description:** Data at-rest encryption using platform keys is supported, any customer content at rest is encrypted with these Microsoft managed keys. [Learn more](#).

Supported	Enabled By Default	Configuration Responsibility
True	True	Microsoft

**Configuration Guidance:** No additional configurations are required as this is enabled on a default deployment.

**Reference:** [Bring your own key \(customer-managed keys\) with Media Services](#)

## DP-5: Use customer-managed key option in data at rest encryption when required

### Features

#### Data at Rest Encryption Using CMK

**Description:** Data at-rest encryption using customer-managed keys is supported for customer content stored by the service. [Learn more](#).

Supported	Enabled By Default	Configuration Responsibility
True	False	Customer

**Configuration Guidance:** If required for regulatory compliance, define the use case and service scope where encryption using customer-managed keys are needed. Enable and implement data at rest encryption using customer-managed key for those services.

**Reference:** [Bring your own key \(customer-managed keys\) with Media Services](#)

## DP-6: Use a secure key management process

### Features

#### Key Management in Azure Key Vault

**Description:** The service supports Azure Key Vault integration for any customer keys, secrets, or certificates. [Learn more.](#)

Supported	Enabled By Default	Configuration Responsibility
False	Not Applicable	Not Applicable

**Configuration Guidance:** This feature is not supported to secure this service.

## DP-7: Use a secure certificate management process

### Features

#### Certificate Management in Azure Key Vault

**Description:** The service supports Azure Key Vault integration for any customer certificates. [Learn more.](#)

Supported	Enabled By Default	Configuration Responsibility
False	Not Applicable	Not Applicable

**Configuration Guidance:** This feature is not supported to secure this service.

## Asset management

*For more information, see the [Microsoft cloud security benchmark: Asset management](#).*

## AM-2: Use only approved services

### Features

#### Azure Policy Support

**Description:** Service configurations can be monitored and enforced via Azure Policy. [Learn more.](#)

Supported	Enabled By Default	Configuration Responsibility
True	False	Customer

**Configuration Guidance:** Use Microsoft Defender for Cloud to configure Azure Policy to audit and enforce configurations of your Azure resources. Use Azure Monitor to create alerts when there is a configuration deviation detected on the resources. Use Azure Policy [deny] and [deploy if not exists] effects to enforce secure configuration across Azure resources.

Reference: [Azure Policy for Media Services](#)

## Logging and threat detection

For more information, see the [Microsoft cloud security benchmark: Logging and threat detection](#).

### LT-4: Enable logging for security investigation

#### Features

##### Azure Resource Logs

**Description:** Service produces resource logs that can provide enhanced service-specific metrics and logging. The customer can configure these resource logs and send them to their own data sink like a storage account or log analytics workspace. [Learn more](#).

Supported	Enabled By Default	Configuration Responsibility
True	False	Customer

**Configuration Guidance:** Enable the resource log. Azure Monitor logging provides additional information about Media Services resources. See the Azure Monitor logging reference.

Reference: [Monitor Media Services](#)

## Backup and recovery

For more information, see the [Microsoft cloud security benchmark: Backup and recovery](#).

### BR-1: Ensure regular automated backups

#### Features

## Azure Backup

**Description:** The service can be backed up by the Azure Backup service. [Learn more.](#)

Supported	Enabled By Default	Configuration Responsibility
False	Not Applicable	Not Applicable

**Configuration Guidance:** This feature is not supported to secure this service.

## Service Native Backup Capability

**Description:** Service supports its own native backup capability (if not using Azure Backup). [Learn more.](#)

Supported	Enabled By Default	Configuration Responsibility
False	Not Applicable	Not Applicable

**Configuration Guidance:** This feature is not supported to secure this service.

## Next steps

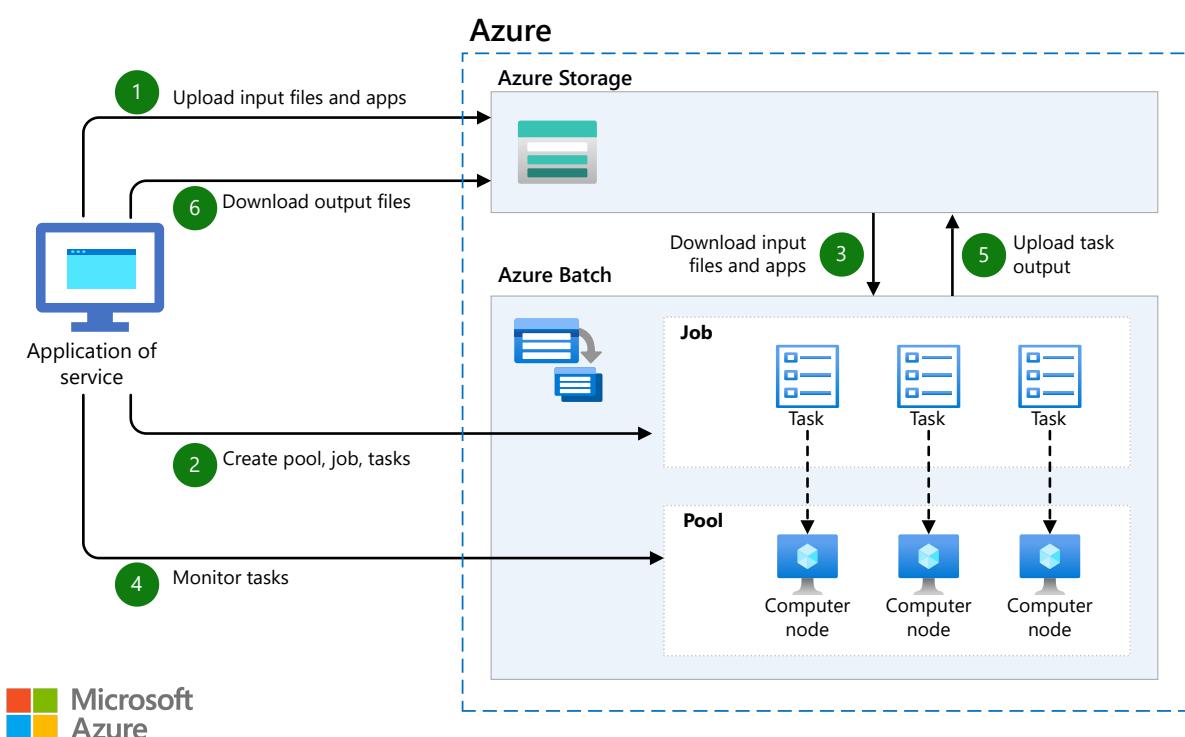
- See the [Microsoft cloud security benchmark overview](#)
- Learn more about [Azure security baselines](#)

# 3D video rendering

Azure Batch   Azure Storage   Azure Virtual Network   Azure Virtual Machine Scale Sets

3D video rendering is a time consuming process that requires a significant amount of CPU time to complete. On a single machine, the process of generating a video file from static assets can take hours or even days depending on the length and complexity of the video you are producing. Many companies will purchase either expensive high end desktop computers to perform these tasks, or invest in large render farms that they can submit jobs to. However, by taking advantage of Azure Batch, that power is available to you when you need it and shuts itself down when you don't, all without any capital investment.

## Architecture



Download a [Visio file](#) of this architecture.

## Dataflow

This scenario shows a workflow that uses [Azure Batch](#). The data flows as follows:

1. Upload input files and the applications to process those files to your Azure Storage account.

2. Create a Batch pool of compute nodes in your Batch account, a job to run the workload on the pool, and tasks in the job.
3. Download input files and the applications to Batch.
4. Monitor task execution.
5. Upload task output.
6. Download output files.

To simplify this process, you could also use the [Batch Plugins for Maya and 3ds Max](#)

## Components

[Azure Batch](#) builds on the following Azure technologies:

- [Azure Virtual Networks](#) are used for both the head node and the compute resources.
- [Azure Storage accounts](#) are used for synchronization and data retention.
- [Azure Virtual Machine Scale Sets](#) are used by CycleCloud for compute resources.

## Alternatives

If you require more control over your rendering environment in Azure or need a hybrid implementation, then CycleCloud computing can help orchestrate an IaaS grid in the cloud. Using the same underlying Azure technologies as Azure Batch, it makes building and maintaining an IaaS grid an efficient process. To find out more, see [What is Azure CycleCloud?](#).

For a complete overview of all the HPC solutions that are available to you in Azure, see the article [HPC, Batch, and Big Compute solutions using Azure VMs](#).

## Scenario details

Batch gives you a consistent management experience and job scheduling, whether you select Windows Server or Linux compute nodes. With Batch, you can use your existing Windows or Linux applications, including AutoDesl Maya and Blender, to run large-scale render jobs in Azure.

## Potential use cases

This solution is ideal for the media and entertainment industries. Other relevant use cases include:

- 3D modeling
- Visual FX (VFX) rendering
- Video transcoding
- Image processing, color correction, and resizing

## Considerations

These considerations implement the pillars of the Azure Well-Architected Framework, which is a set of guiding tenets that can be used to improve the quality of a workload. For more information, see [Microsoft Azure Well-Architected Framework](#).

## Machine Sizes available for Azure Batch

While most rendering customers will choose resources with high CPU power, other workloads using virtual machine scale sets may choose VMs differently and will depend on a number of factors:

- Is the application being run memory bound?
- Does the application need to use GPUs?
- Are the job types embarrassingly parallel or require infiniband connectivity for tightly coupled jobs?
- Require fast I/O to access storage on the compute Nodes.

Azure has a wide range of VM sizes that can address each and every one of the above application requirements, some are specific to HPC, but even the smallest sizes can be used to provide an effective grid implementation:

- [HPC VM sizes](#) Due to the CPU bound nature of rendering, Microsoft typically suggests the Azure H-Series VMs. This type of VM is built specifically for high end computational needs, they have 8 and 16 core vCPU sizes available, and features DDR4 memory, SSD temporary storage, and Haswell E5 Intel technology.
- [GPU VM sizes](#) GPU optimized VM sizes are specialized virtual machines available with single or multiple NVIDIA GPUs. These sizes are designed for compute-intensive, graphics-intensive, and visualization workloads.
- NC, NCv2, NCv3, and ND sizes are optimized for compute-intensive and network-intensive applications and algorithms, including CUDA and OpenCL-based applications and simulations, AI, and Deep Learning. NV sizes are optimized and designed for remote visualization, streaming, gaming, encoding, and VDI scenarios using frameworks such as OpenGL and DirectX.
- [Memory optimized VM sizes](#) When more memory is required, the memory optimized VM sizes offer a higher memory-to-CPU ratio.

- [General purposes VM sizes](#) General-purpose VM sizes are also available and provide balanced CPU-to-memory ratio.

## Availability

Monitoring of the Azure Batch components is available through a range of services, tools, and APIs. Monitoring is discussed further in the [Monitor Batch solutions](#) article.

## Scalability

Pools within an Azure Batch account can either scale through manual intervention or, by using a formula based on Azure Batch metrics, be scaled automatically. For more information on scalability, see the article [Create an automatic scaling formula for scaling nodes in a Batch pool](#).

## Security

Security provides assurances against deliberate attacks and the abuse of your valuable data and systems. For more information, see [Overview of the security pillar](#).

For general guidance on designing secure solutions, see the [Azure Security Documentation](#).

## Resiliency

While there is currently no failover capability in Azure Batch, we recommend using the following steps to ensure availability if there is an unplanned outage:

- Create an Azure Batch account in an alternate Azure location with an alternate Storage Account
- Create the same node pools with the same name, with zero nodes allocated
- Ensure Applications are created and updated to the alternate Storage Account
- Upload input files and submit jobs to the alternate Azure Batch account

## Cost optimization

Cost optimization is about looking at ways to reduce unnecessary expenses and improve operational efficiencies. For more information, see [Overview of the cost optimization pillar](#).

The cost of using Azure Batch will depend on the VM sizes that are used for the pools and how long these VMs are allocated and running, there is no cost associated with an Azure Batch account creation. Storage and data egress should be taken into account as these will apply additional costs.

The following are examples of costs that could be incurred for a job that completes in 8 hours using a different number of servers:

- 100 High-Performance CPU VMs: [Cost Estimate ↗](#)  
100 x H16m (16 cores, 225 GB RAM, Premium Storage 512 GB), 2 TB Blob Storage, 1-TB egress
- 50 High-Performance CPU VMs: [Cost Estimate ↗](#)  
50 x H16m (16 cores, 225 GB RAM, Premium Storage 512 GB), 2 TB Blob Storage, 1-TB egress
- 10 High-Performance CPU VMs: [Cost Estimate ↗](#)  
10 x H16m (16 cores, 225 GB RAM, Premium Storage 512 GB), 2 TB Blob Storage, 1-TB egress

## Pricing for low-priority VMs

Azure Batch also supports the use of low-priority VMs in the node pools, which can potentially provide a substantial cost saving. For more information, including a price comparison between standard VMs and low-priority VMs, see [Azure Batch Pricing ↗](#).

### Note

Low-priority VMs are only suitable for certain applications and workloads.

## Deploy this scenario

### Create an Azure Batch account and pools manually

This scenario demonstrates how Azure Batch works while showcasing Azure Batch Labs as an example SaaS solution that can be developed for your own customers:

[Azure Batch Labs ↗](#)

# Deploy the components

The template will deploy:

- A new Azure Batch account
- A storage account
- A node pool associated with the batch account
- The node pool will be configured to use A2 v2 VMs with Canonical Ubuntu images
- The node pool will contain zero VMs initially and will require you to manually scale to add VMs

Click the link below to deploy the solution.



[Learn more about Resource Manager templates](#)

## Next steps

Product documentation:

- [What is Azure Batch?](#)
- [What is Azure Virtual Network?](#)
- [Azure Storage accounts](#)
- [What are Virtual Machine Scale Sets?](#)

Learn modules:

- [Introduction to Azure Remote Rendering](#)
- [Render a model with Azure Remote Rendering](#)

## Related resources

- [HPC media rendering](#)
- [HPC system and big-compute solutions](#)
- [Run CFD simulations](#)

# Batch scoring for deep learning models using Azure Machine Learning pipelines

Azure Logic Apps

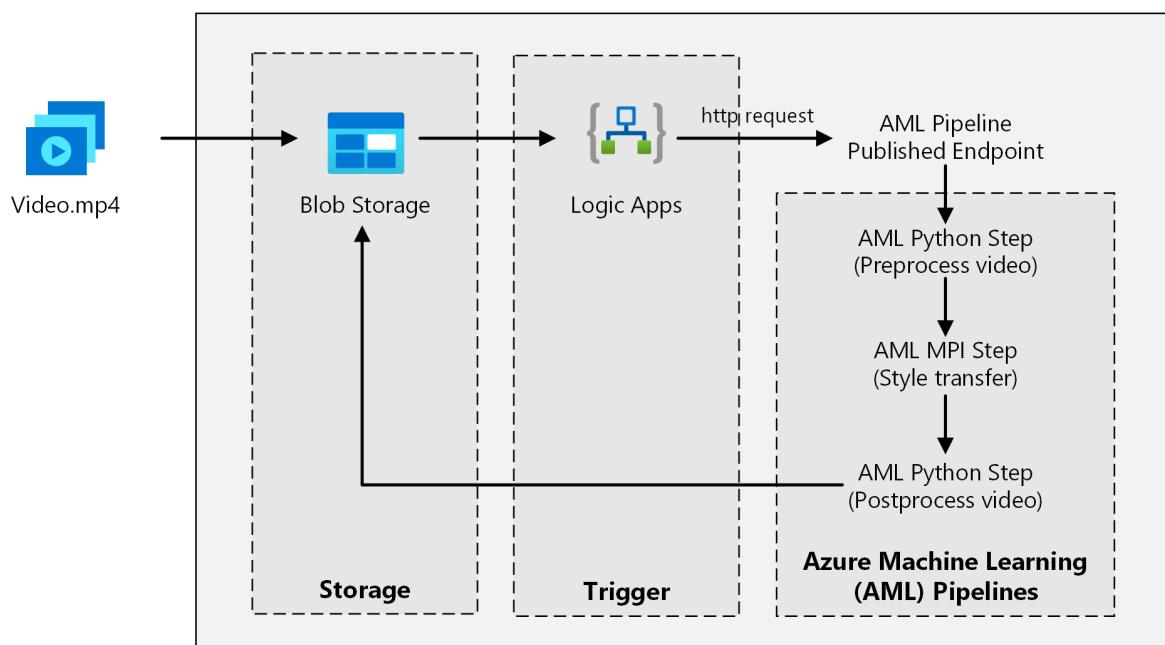
Azure Machine Learning

Azure Role-based access control

Azure Storage

This reference architecture shows how to apply neural-style transfer to a video, using Azure Machine Learning. *Style transfer* is a deep learning technique that composes an existing image in the style of another image. You can generalize this architecture for any scenario that uses batch scoring with deep learning.

## Architecture



Download a [Visio file](#) of this architecture.

## Workflow

This architecture consists of the following components.

## Compute

[Azure Machine Learning](#) uses pipelines to create reproducible and easy-to-manage sequences of computation. It also offers a managed compute target (on which a pipeline computation can run) called [Azure Machine Learning Compute](#) for training, deploying, and scoring machine learning models.

## Storage

[Azure Blob Storage](#) stores all the images (input images, style images, and output images). Azure Machine Learning integrates with Blob Storage so that users don't have to manually move data across compute platforms and blob storages. Blob Storage is also cost-effective for the performance that this workload requires.

## Trigger

[Azure Logic Apps](#) triggers the workflow. When the Logic App detects that a blob has been added to the container, it triggers the Azure Machine Learning pipeline. Logic Apps is a good fit for this reference architecture because it's an easy way to detect changes to blob storage, with an easy process for changing the trigger.

## Preprocess and postprocess the data

This reference architecture uses video footage of an orangutan in a tree.

1. Use [FFmpeg](#) to extract the audio file from the video footage, so that the audio file can be stitched back into the output video later.
2. Use FFmpeg to break the video into individual frames. The frames are processed independently, in parallel.
3. At this point, you can apply neural style transfer to each individual frame in parallel.
4. After each frame has been processed, use FFmpeg to restitch the frames back together.
5. Finally, reattach the audio file to the restitched footage.

## Components

- [Azure Machine Learning](#)
- [Azure Blob Storage](#)
- [Azure Logic Apps](#)

## Solution details

This reference architecture is designed for workloads that are triggered by the presence of new media in Azure storage.

Processing involves the following steps:

1. Upload a video file to Azure Blob Storage.
2. The video file triggers Azure Logic Apps to send a request to the Azure Machine Learning pipeline published endpoint.
3. The pipeline processes the video, applies style transfer with MPI, and postprocesses the video.
4. The output is saved back to Blob Storage once the pipeline is completed.

## Potential use cases

A media organization has a video whose style they want to change to look like a specific painting. The organization wants to apply this style to all frames of the video in a timely manner and in an automated fashion. For more background about neural style transfer algorithms, see [Image Style Transfer Using Convolutional Neural Networks](#) (PDF).

## Considerations

These considerations implement the pillars of the Azure Well-Architected Framework, which is a set of guiding tenets that can be used to improve the quality of a workload. For more information, see [Microsoft Azure Well-Architected Framework](#).

## Performance efficiency

Performance efficiency is the ability of your workload to scale to meet the demands placed on it by users in an efficient manner. For more information, see [Performance efficiency pillar overview](#).

## GPU versus CPU

For deep learning workloads, GPUs generally out-perform CPUs by a considerable amount, to the extent that a sizeable cluster of CPUs is typically needed to get comparable performance. Although you can use only CPUs in this architecture, GPUs provide a much better cost/performance profile. We recommend using the latest [NCv3 series](#) of GPU optimized VMs.

GPUs aren't enabled by default in all regions. Make sure to select a region with GPUs enabled. In addition, subscriptions have a default quota of zero cores for GPU-optimized

VMs. You can raise this quota by opening a support request. Make sure that your subscription has enough quota to run your workload.

## Parallelize across VMs versus cores

When you run a style transfer process as a batch job, the jobs that run primarily on GPUs need to be parallelized across VMs. Two approaches are possible: You can create a larger cluster using VMs that have a single GPU, or create a smaller cluster using VMs with many GPUs.

For this workload, these two options have comparable performance. Using fewer VMs with more GPUs per VM can help to reduce data movement. However, the data volume per job for this workload isn't large, so you won't observe much throttling by Blob Storage.

## MPI step

When creating the [Azure Machine Learning pipeline](#), one of the steps used to perform parallel computation is the (message processing interface) MPI step. The MPI step helps split the data evenly across the available nodes. The MPI step doesn't execute until all the requested nodes are ready. Should one node fail or get preempted (if it's a low-priority virtual machine), the MPI step will have to be rerun.

## Security

Security provides assurances against deliberate attacks and the abuse of your valuable data and systems. For more information, see [Overview of the security pillar](#). This section contains considerations for building secure solutions.

## Restrict access to Azure Blob Storage

In this reference architecture, Azure Blob Storage is the main storage component that needs to be protected. The baseline deployment shown in the GitHub repo uses storage account keys to access the blob storage. For further control and protection, consider using a [shared access signature \(SAS\)](#) instead. This grants limited access to objects in storage, without needing to hard code the account keys or save them in plaintext. This approach is especially useful because account keys are visible in plaintext inside of Logic App's designer interface. Using an SAS also helps to ensure that the storage account has proper governance, and that access is granted only to the people intended to have it.

For scenarios with more sensitive data, make sure that all of your storage keys are protected, because these keys grant full access to all input and output data from the workload.

## Data encryption and data movement

This reference architecture uses style transfer as an example of a batch scoring process. For more data-sensitive scenarios, the data in storage should be encrypted at rest. Each time data is moved from one location to the next, use Transport Layer Security (TSL) to secure the data transfer. For more information, see [Azure Storage security guide](#).

## Secure your computation in a virtual network

When deploying your Machine Learning compute cluster, you can configure your cluster to be provisioned inside a subnet of a [virtual network](#). This subnet allows the compute nodes in the cluster to communicate securely with other virtual machines.

## Protect against malicious activity

In scenarios where there are multiple users, make sure that sensitive data is protected against malicious activity. If other users are given access to this deployment to customize the input data, note the following precautions and considerations:

- Use [Azure role-based access control \(Azure RBAC\)](#) to limit users' access to only the resources they need.
- Provision two separate storage accounts. Store input and output data in the first account. External users can be given access to this account. Store executable scripts and output log files in the other account. External users should not have access to this account. This separation ensures that external users can't modify any executable files (to inject malicious code), and don't have access to log files, which could hold sensitive information.
- Malicious users can perform a [DDoS attack](#) on the job queue or inject malformed poison messages in the job queue, causing the system to lock up or causing dequeuing errors.

## Cost optimization

Cost optimization is about looking at ways to reduce unnecessary expenses and improve operational efficiencies. For more information, see [Overview of the cost optimization pillar](#).

Compared to the storage and scheduling components, the compute resources used in this reference architecture by far dominate in terms of costs. One of the main challenges is effectively parallelizing the work across a cluster of GPU-enabled machines.

The Azure Machine Learning Compute cluster size can automatically scale up and down depending on the jobs in the queue. You can enable autoscale programmatically by setting the minimum and maximum nodes.

For work that doesn't require immediate processing, configure autoscale so the default state (minimum) is a cluster of zero nodes. With this configuration, the cluster starts with zero nodes and only scales up when it detects jobs in the queue. If the batch scoring process happens only a few times a day or less, this setting results in significant cost savings.

Autoscaling may not be appropriate for batch jobs that happen too close to each other. The time that it takes for a cluster to spin up and spin down also incur a cost, so if a batch workload begins only a few minutes after the previous job ends, it might be more cost effective to keep the cluster running between jobs.

Azure Machine Learning Compute also supports low-priority virtual machines, which allows you to run your computation on discounted virtual machines, with the caveat that they may be preempted at any time. Low-priority virtual machines are ideal for non-critical batch scoring workloads.

## Monitor batch jobs

While running your job, it's important to monitor the progress and make sure that the job is working as expected. However, it can be a challenge to monitor across a cluster of active nodes.

To check the overall state of the cluster, go to the Machine Learning service in the Azure portal to check the state of the nodes in the cluster. If a node is inactive or a job has failed, the error logs are saved to Blob Storage, and are also accessible in the Azure portal.

Monitoring can be further enriched by connecting logs to Application Insights or by running separate processes to poll for the state of the cluster and its jobs.

## Log with Azure Machine Learning

Azure Machine Learning will automatically log all stdout/stderr to the associated Blob Storage account. Unless otherwise specified, your Azure Machine Learning workspace will automatically provision a storage account and dump your logs into it. You can also

use a storage navigation tool such as [Azure Storage Explorer](#), which is an easier way to navigate log files.

## Deploy this scenario

To deploy this reference architecture, follow the steps described in the [GitHub repo](#).

You can also deploy a batch scoring architecture for deep learning models by using the Azure Kubernetes Service. Follow the steps described in this [GitHub repo](#).

## Contributors

*This article is maintained by Microsoft. It was originally written by the following contributors.*

Principal author:

- [Jian Tang](#) | Program Manager II

*To see non-public LinkedIn profiles, sign in to LinkedIn.*

## Next steps

- [Batch scoring of Spark models on Azure Databricks](#)
- [Batch scoring of Python models on Azure](#)
- [Batch scoring with R models to forecast sales](#)

## Related resources

- [Artificial intelligence architecture](#)
- [What is Azure Machine Learning?](#)
- [Azure Machine Learning pipelines](#)

# Gridwich cloud media system

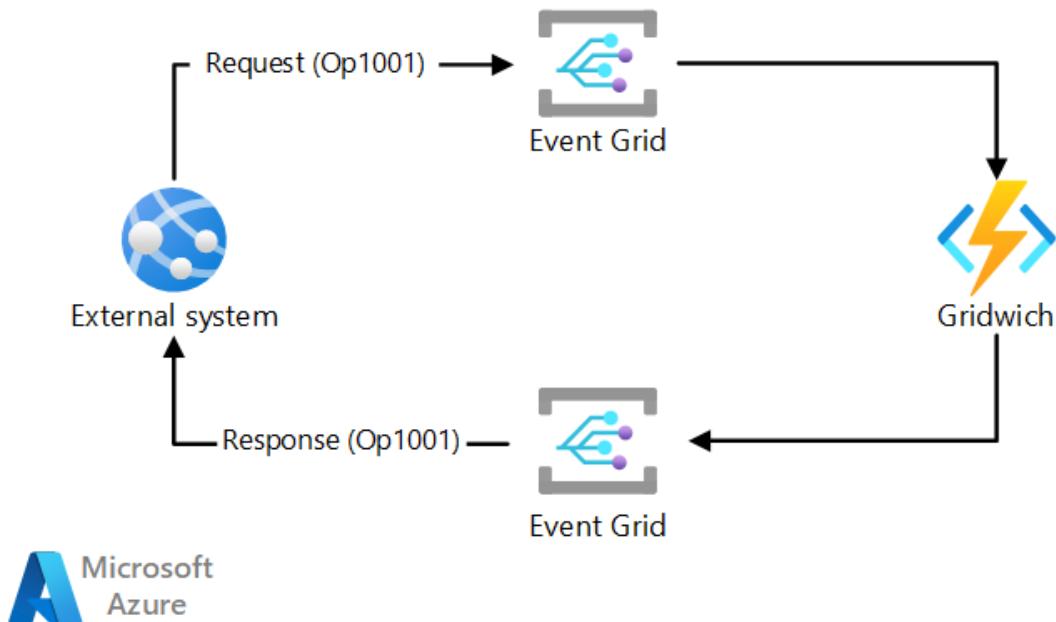
Azure Blob Storage   Azure Event Grid   Azure Functions   Azure Logic Apps   Azure Media Services

The Gridwich pipelines ingest, process, store, and deliver media assets with the help of two new methods, the *Azure Event Grid Sandwich* and the *Terraform Sandwich*.

## Architecture

The Gridwich architecture features two *sandwiches* that address the requirements of asynchronous event processing and infrastructure as code:

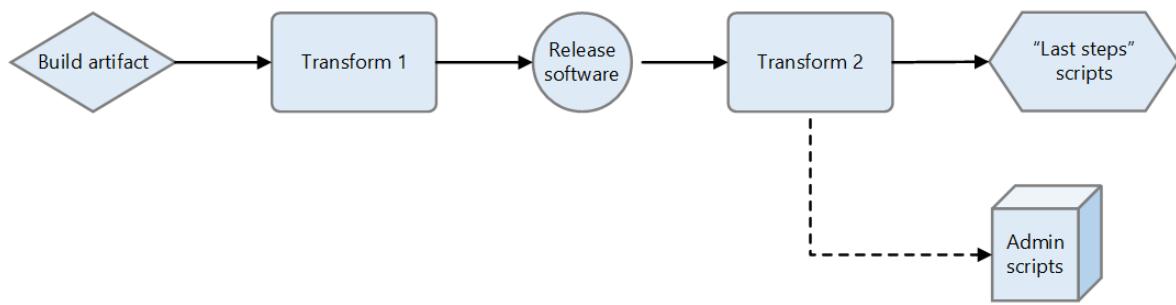
- The *Event Grid sandwich* abstracts away remote and long-running processes like media encoding from the external saga workflow system by sandwiching them between two Event Grid handlers. This sandwich lets the external system send a request event, monitor scheduled events, and wait for an eventual success or failure response that might arrive minutes or hours later.



Download a [Visio file](#) of this architecture.

- The *Terraform Sandwich* is a multi-stage [Terraform](#) pattern updated to support [infrastructure as code](#). Separating infrastructure and software releases means the Azure Functions app must be released and running before Terraform can deploy

the Event Grid subscription. To address this requirement, there are two Terraform jobs in the CI/CD pipeline:



- Terraform 1 creates all the resources except for the Azure Event Grid subscriptions.
- Terraform 2 creates the Event Grid subscriptions after the software is up and running.

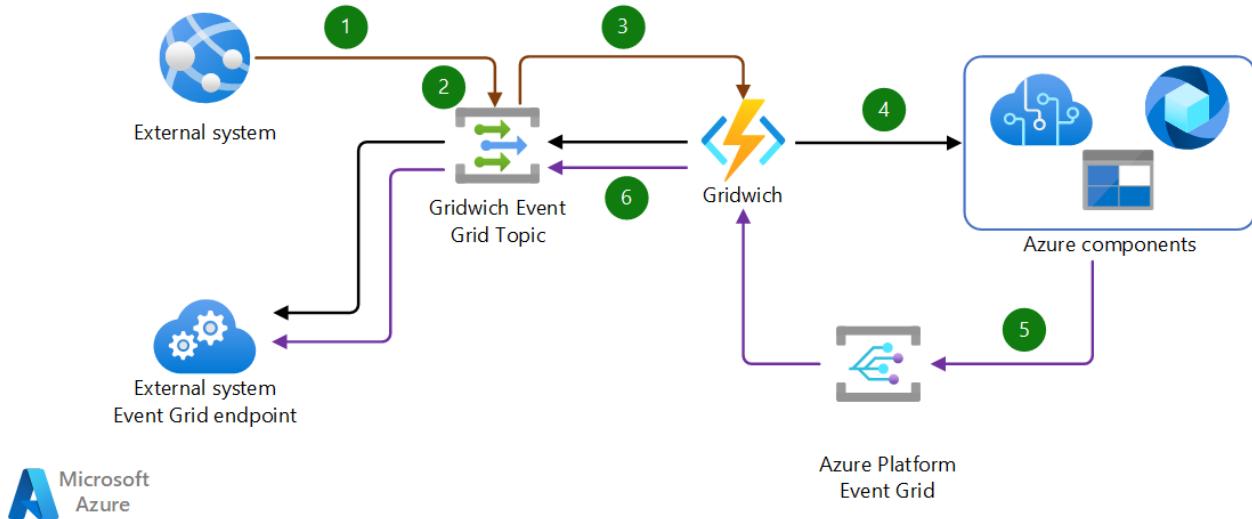
This way, Terraform can wholly manage and deploy the solution infrastructure, even when not all the [Azure resources](#) can be created before the software artifacts are deployed.

## Workflow

The Gridwich request and response process covers the following requests:

- Creation
- Transport
- Reception
- Dispatch to Gridwich components
- Acknowledgment and actions
- Responses

The following steps describe the request and response process between an external system and Gridwich. In Gridwich, the external system is a MAM and saga workflow orchestration system. For the exact formats of Gridwich operations message events, see [Gridwich message formats](#).



1. The external system creates a request and sends it to the request broker.
2. The request broker is responsible for dispatching requests to Gridwich request listeners in a traditional publication-subscription model. In this solution, the request broker is Azure Event Grid. All requests are encapsulated using the [Event Grid event schema](#).
3. The Gridwich Azure Functions app consumes events from Event Grid. For better throughput, Gridwich defines an [HTTP endpoint](#) as a push model that Event Grid initiates, rather than the [Event Grid binding](#) polling model that Azure Functions provides.
4. The Azure Functions App reads the event properties and dispatches events to parts of the Gridwich code that handle that event type and version.
5. Any handler that will work with the current request uses the common [EventGridHandlerBase](#) class to immediately send an Acknowledgment message when it receives the request. The handler then dispatches the work to the derived class.

Gridwich request workflows can be synchronous or asynchronous in nature. For requests that are easy to perform and fast to complete, a [synchronous handler](#) does the work and returns the Success or Failure event almost immediately after the Acknowledgment is sent.

For requests that are long-running, an [asynchronous handler](#) evaluates the request, validates arguments, and initiates the long-running operation. The handler then returns a Scheduled response to confirm that it requested the work activity. On completing the work activity, the request handler is responsible for providing a Success or Failure completed event for the work.

The event publication service communicates the Acknowledgment, Failure, Scheduled, or Success messages to the Event Grid request broker.

6. The event publisher in the Azure Function sends the response event to an Event Grid topic, which acts as a reliable message broker. The external system subscribes to the topic and consumes the messages. The Event Grid platform provides its normal retry logic for publication to the external system.

## Message order

While an Acknowledgment precedes both the Success and Scheduled responses, Gridwich doesn't guarantee that a Scheduled response will always precede the corresponding Success response. A valid response sequence could be either **Acknowledged > Scheduled > Success** or **Acknowledged > Success > Scheduled**.

## Request failures

Request failures can be caused by bad requests, missing pre-conditions, processing failures, security exceptions, or unhandled exceptions. Almost all failures have the same message form, and include the original [operation context](#) value. The common [EventGridHandlerBase](#) class typically sends Failure responses to Event Grid via the Azure Function event publisher interface. [Application Insights](#) also logs failures via [structured logging](#).

## Operation context

The external system might generate thousands of requests per day, per hour, or per second. Each [request event](#) to Gridwich must include a JSON object property named `operationContext`.

If a request contains an operation context, like `{"id": "Op1001"}`, each Gridwich [response](#) must include a corresponding opaque *operation context*, whether the request is short-running or long-running. This operation context persists through the lifetime of even very long-running requests.

The response requirement is for a "corresponding" rather than the "same" JSON object. Gridwich features like [context muting](#) take advantage of the fact that the external system processes the returned JSON object in a top-down fashion.

Specifically, the external system has:

- No dependency on property ordering, so Gridwich can send back an object with the same properties, possibly in a different order. For example, `{"a":1,"b":2}` vs. `{"b":2,"a":1}`.
- No issue with extra properties being present, so Gridwich, having received `{"b":2,"a":1}`, could validly return `{"a":1,"b":2,"~somethingExtra":"yes"}`. To minimize the possibility of collisions, Gridwich prefixes the names of added properties with a tilde (~), for example `~muted`.
- No JSON-formatting dependencies. For example, there are no assumptions about where whitespace padding may fall within the string representation of the JSON. Gridwich capitalizes on this lack of formatting dependency by compressing out unneeded whitespace in string representations of the JSON objects. See [JSONHelpers.SerializeOperationContext](#).

## Saga participants and operation context

In the Gridwich saga orchestration system, each [saga participant](#) contributes one or more work activities to the system. Each saga participant works independently of the other participants, and more than one saga participant might act on a single request.

Each of the saga participants must retain the operation context, but may implement it differently. For example:

- Short-running synchronous operations retain the operation context.
- Azure Storage provides an opaque string property called `ClientRequestId` for most operations.
- Azure Media Services v3 has a `Job.CorrelationData` property.
- Other cloud APIs offer similar concepts to an opaque operation context that they can return when signaling progress, completion, or failure.

For more information about sagas and saga participants, see [Saga orchestration](#).

## Synchronous and asynchronous handlers

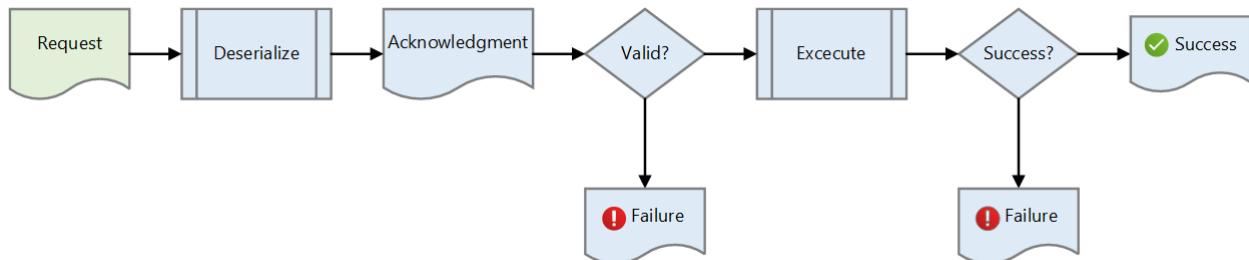
Every event handler in the system uses a common [EventGridHandlerBase](#) class to provide generic services such as request acknowledgment, failure handling, and publication of response events. The event publication service communicates the Acknowledgment, Failure, Scheduled, or Success messages to the Event Grid request broker.

Any handler that plans to work with the current request must provide an acknowledgment when it receives the request. The base class immediately sends an Acknowledgment message, and then dispatches the work to the derived class.

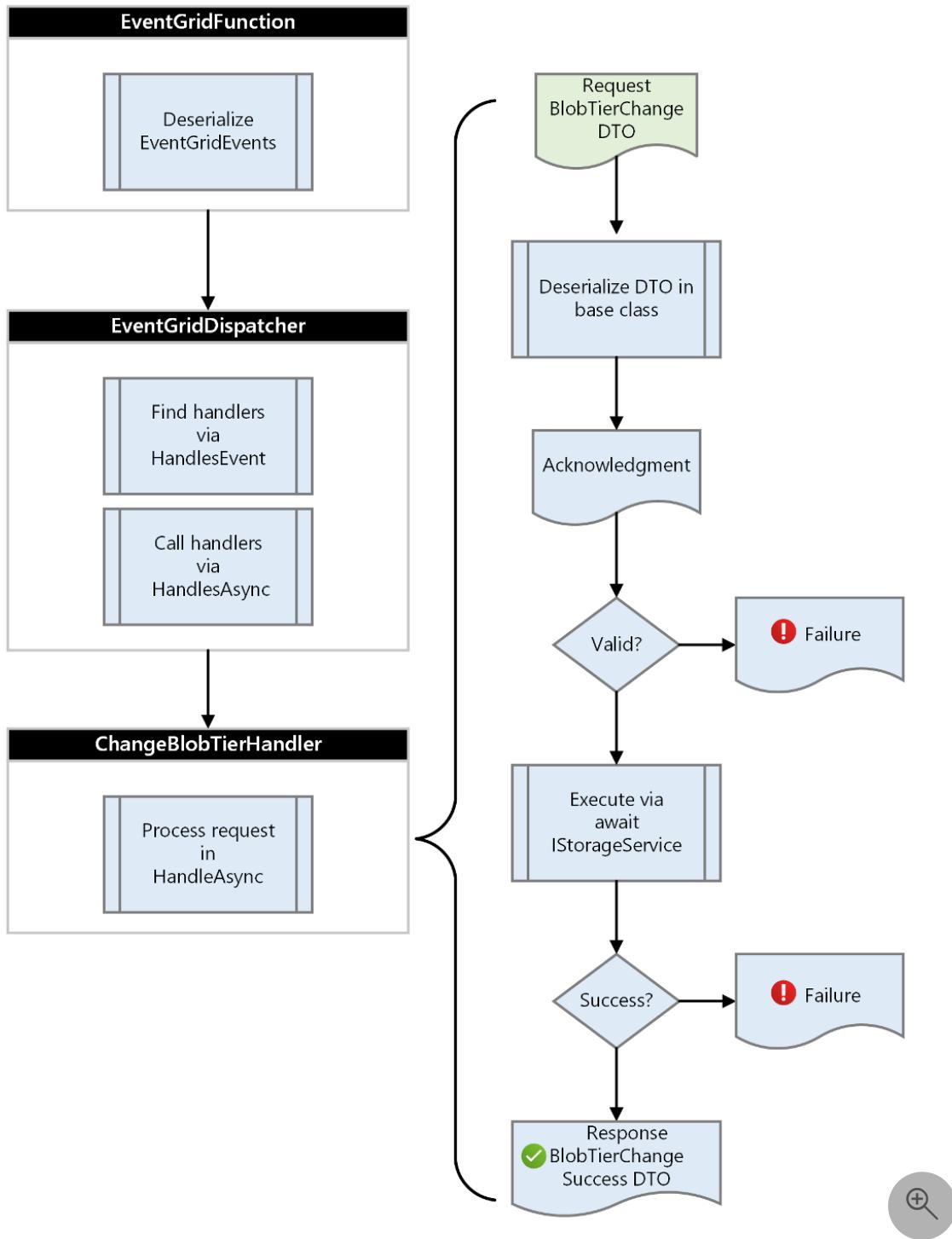


## Synchronous event processing

For requests that are easy to perform and fast to complete, the handler does the work synchronously and returns the Success event, with its operation context, almost immediately after the Acknowledgment is sent.

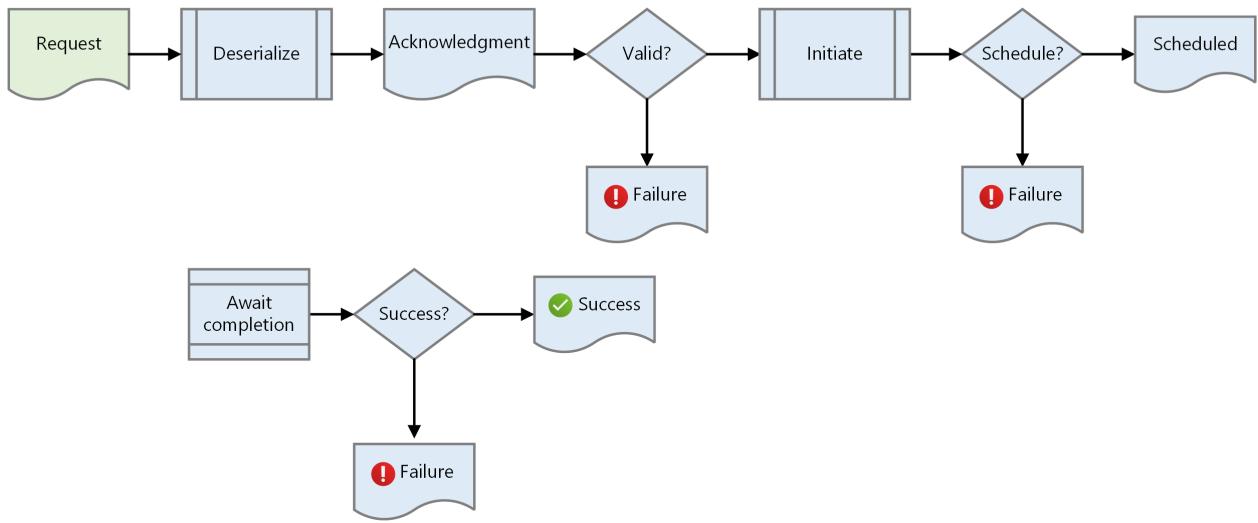


For example, the [ChangeBlobTierHandler](#) is a simple synchronous flow. The handler gets a Request data transfer object (DTO), calls and awaits a single service to do the work, and returns a Success or Failure response.



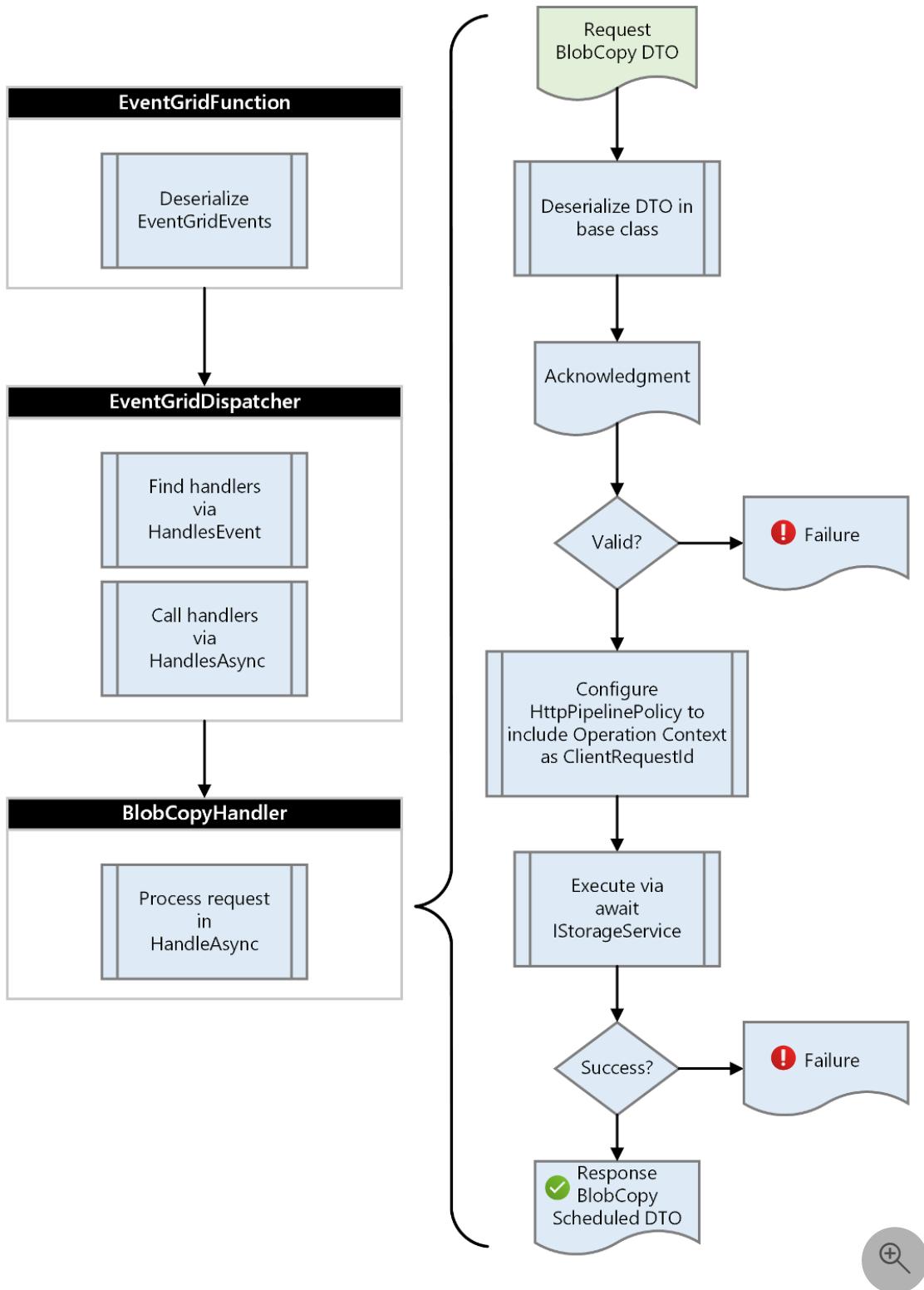
## Asynchronous event processing

Some requests are long-running. For example, encoding media files can take hours. In these cases, an *asynchronous request handler* evaluates the request, validates arguments, and initiates the long-running operation. The handler then returns a Scheduled response to confirm that it requested the work activity.

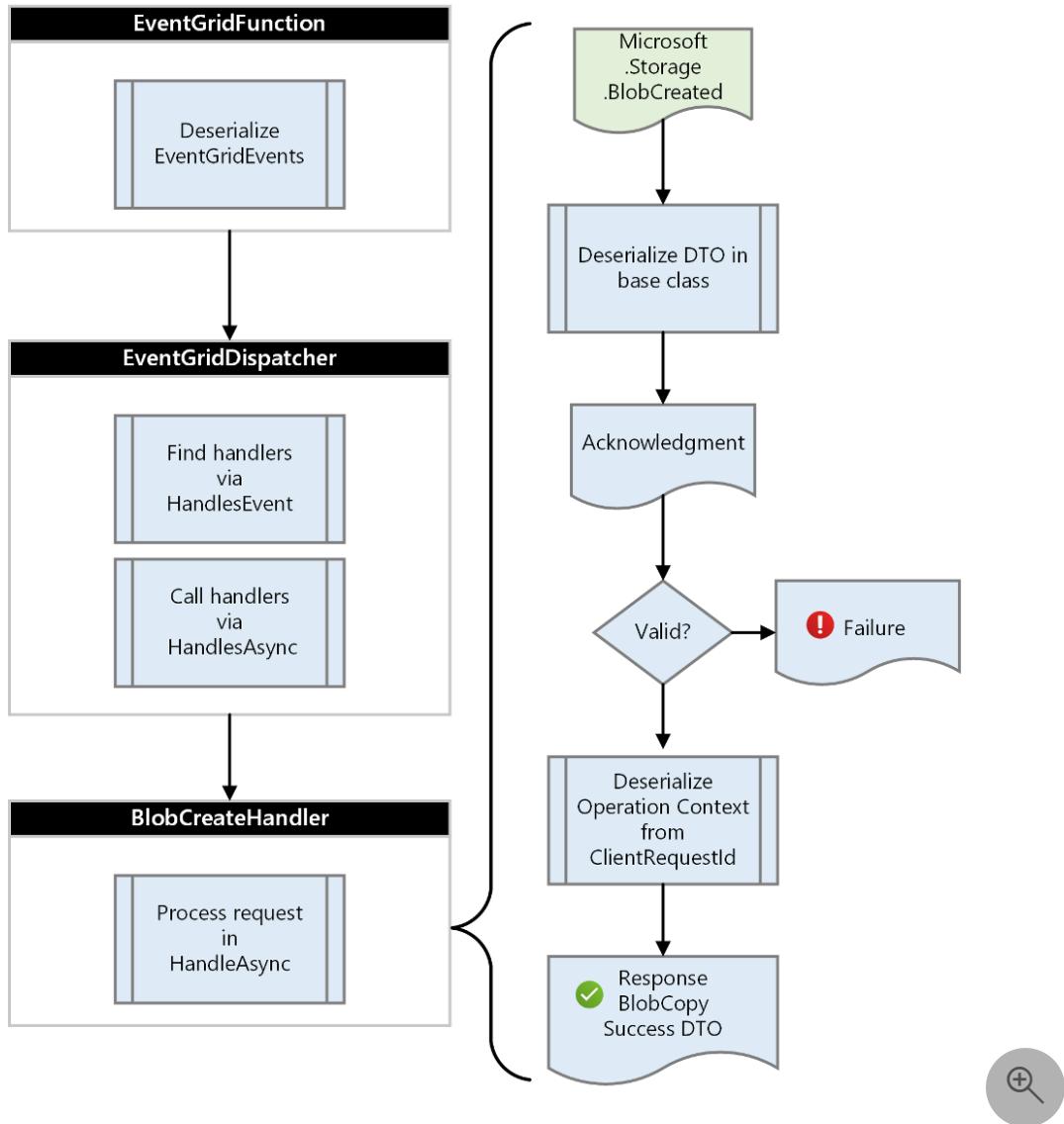


On completing the work activity, the request handler is responsible for providing a Success or Failure completed event for the work. While remaining stateless, the handler must retrieve the original [operation context](#) and place it in the Completed event message payload.

For example, the [BlobCopyHandler](#) shows a simple asynchronous flow. The handler gets a Request DTO, calls and awaits a single service to initiate the work, and publishes a Scheduled or Failure response.



To complete the long-running request flow, the [BlobCreatedHandler](#) consumes the platform event `Microsoft.Storage.BlobCreated`, extracts the original operation context, and publishes a Success or Failure completion response.



## Long-running functions

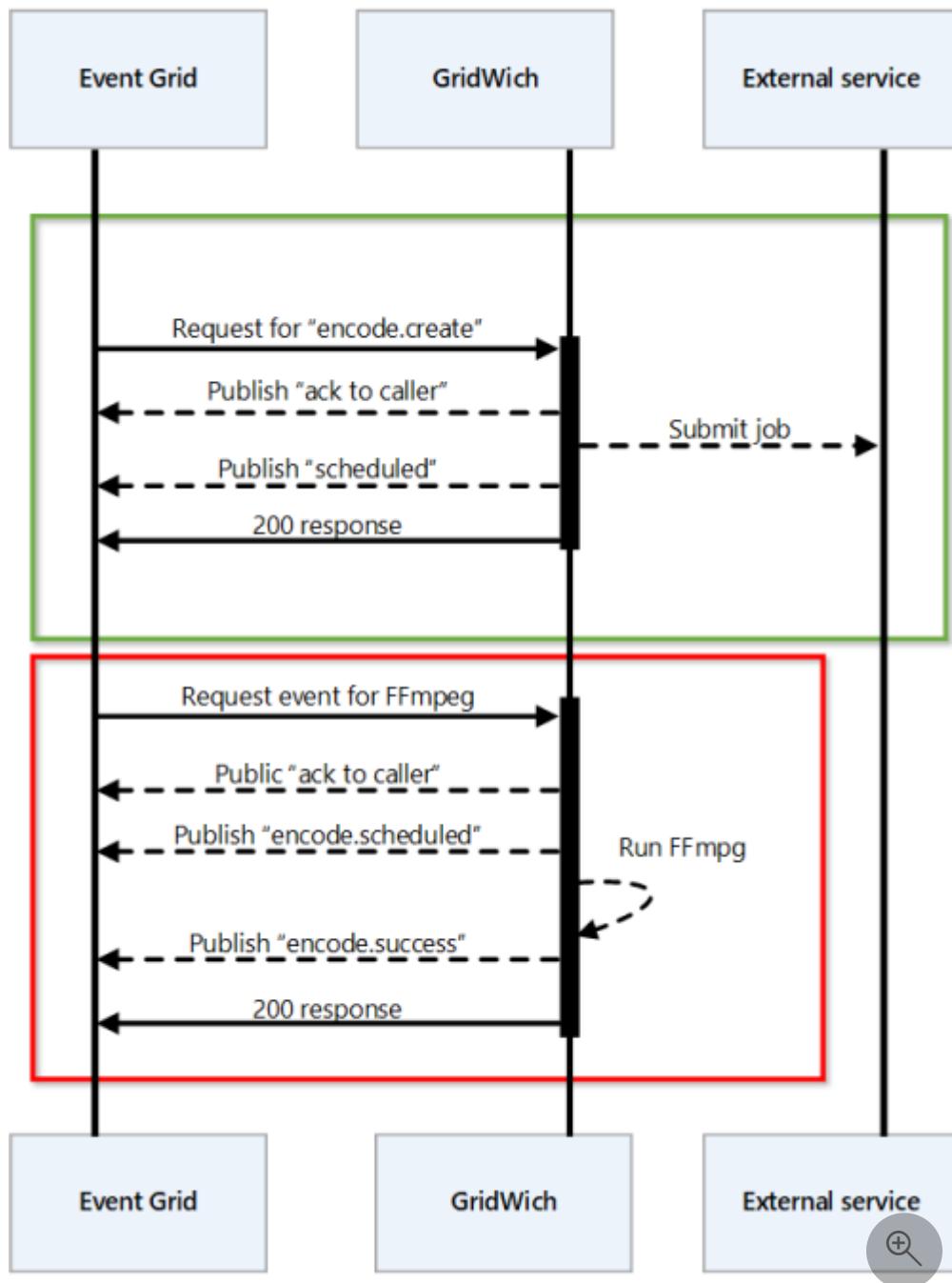
When Gridwich deploys a new Functions App, it may drop current long-running processes. If these processes end abruptly, there's no status and no report back to the caller. Gridwich must deploy new Functions Apps while gracefully handling the transition for long-running functions and not missing any messages.

The solution must:

- Not keep the state of running instances of the Gridwich app.
- Not kill processes just because something new is deploying or a new message is requesting the same activity.
- Not invoke any additional Azure workloads, like Durable Functions, Functions Apps, Logic Apps, or Azure Container Instances.

Gridwich uses Azure Functions *slot deployment* and *cancellation tokens* to meet these requirements for reliable, long-running functions.

The following diagram shows how most Gridwich jobs work. The green box represents a job that Gridwich passes to an external service. Gridwich then reacts in an event-driven way to the status. The red box shows a function that is long-running on Gridwich itself.



The Functions runtime adds the cancellation token when the application is shutting down. Gridwich detects the token and returns error codes for all requests and currently running processes.

Slot deployment deploys new software versions. The production slot has the running application, and the staging slot has the new version. Azure does a series of deployment

steps and then swaps the slot instances. The old instance restarts as the last step of the process.

Gridwich waits 30 seconds after remapping the host names, so for HTTP-triggered functions, Gridwich guarantees at least 30 seconds before the restart for the old production slot. Other triggers might be controlled by app settings that don't have a mechanism to wait on app setting updates. Those functions risk interruption if execution starts right before the old production slot restarts.

For more information, see [What happens during a slot swap for Azure Functions](#) and [Azure Functions deployment slots](#).

## Components

The Gridwich media processing solution uses Azure Event Grid, Azure Functions, Azure Media Services, Azure Blob Storage, Azure Logic Apps, and Azure Key Vault. The CI/CD and saga orchestration processes use Azure Repos, Azure Pipelines, and Terraform.

- [Azure Event Grid](#) ↗ manages the routing of Gridwich events, with two sandwiched Event Grid jobs that allow for asynchronous media event processing. Event Grid is a highly reliable request delivery endpoint. The Azure platform provides necessary request delivery endpoint uptime and stability.

Gridwich encapsulates events within the [Event Grid schema](#) `Event.Data` property object, which is opaque to the Event Grid broker and transport layer. Gridwich also uses the `eventType` and `dataVersion` object fields to route events. So that the Event Grid request broker can be substituted with other publication-subscription event brokers, Gridwich depends on the fewest event fields possible, and doesn't use the `topic` or `subject` fields.

- [Azure Functions](#) ↗ lets you run event-triggered code without having to explicitly provision or manage infrastructure. Gridwich is an Azure Functions App that hosts execution of various functions.
- [Azure Media Services](#) ↗ is a cloud-based workflow platform for indexing, packaging, protecting, and streaming media. Media Services uses [Digital Rights Management \(DRM\)](#) ↗ to protect content, and supports [Microsoft PlayReady](#) ↗, [Google Widevine](#) ↗, and [Apple FairPlay](#) ↗. Gridwich lets you [scale Media Services resources](#) to your expected workload.
- [Azure Blob Storage](#) ↗ provides scalable, cost-efficient cloud storage and access for unstructured data like media assets. Gridwich uses both Azure Storage block blobs and containers.

- [Azure Logic Apps](#) lets you create automated cloud workflow solutions. Gridwich uses Logic Apps to [manage Key Vault keys and secrets](#).
- [Azure Key Vault](#) safeguards cryptographic keys, passwords, and other secrets that Azure and third-party apps and services use.
- [Azure DevOps](#) is a set of developer and operations services, including Git-based code repositories and automated build and release pipelines, that integrate with Azure. Gridwich uses [Azure Repos](#) to store and update the code projects, and [Azure Pipelines](#) for CI/CD and other workflows.
- [Terraform](#) is an open-source tool that uses Infrastructure as Code to provision and manage infrastructures and services.

## Alternatives

- [Durable Functions](#), which have a built-in state store for long-running operations, could also provide an opaque operation context. Durable Functions could create a series of tasks within an operation, and save the operation context as an input or output for the operation. In fact, Gridwich could use Durable Functions for all work activities, but this approach would increase code complexity.
- You could achieve better decoupling from the Event Grid infrastructure by refactoring the [EventGridHandlerBase](#) into a `RequestHandlerBase`, and removing any linkage to Event Grid objects or types. This refactored class would deal only in base DTOs, and not in transport-specific object types. Similarly, the [IEventGridDispatcher](#) could become an `IResponseDispatcher` with a specific `EventGridDispatcher` implementation.
- The [Gridwich.SagaParticipants.Storage.AzureStorage](#) library also contains storage services that other saga participants use. Having the interfaces in a core project avoids Inversion of Control (IoC) issues, but you could extract the interfaces into a separate core storage infrastructure gateway library.
- The Gridwich Functions App uses [dependency injection](#) to register one or more request handlers for specific event types and data versions. The app injects the [EventGridDispatcher](#) with the collection of Event Grid event handlers, and the dispatcher queries the handlers to determine which ones will process the event.

Alternatively, you could use the event subscription and filtering mechanism that the Event Grid platform provides. This mechanism imposes a 1:1 deployment model, where one Azure Function hosts only one event handler. Although Gridwich

uses a 1:many model, its [clean architecture](#) means that refactoring the solution for 1:1 wouldn't be difficult.

- For an alternative microservices rather than monolithic Gridwich architecture, see [Microservices alternative](#).

## Scenario details

A well-known mass media and entertainment conglomerate replaced their on-premises video streaming service with a cloud-based solution for ingesting, processing, and publishing video assets. The company's main goals were to take advantage of Azure cloud capacity, cost, and flexibility to:

- Ingest raw video files, process and publish them, and fulfill media requests.
- Improve both encoding and new intake and distribution capabilities at scale, and with a cleanly architected approach.
- Implement continuous integration and delivery (CI/CD) for the media asset management (MAM) pipeline.

To meet these goals, the Microsoft engineering team developed Gridwich, a stateless event-processing framework driven by an external [saga workflow orchestration system](#).

## Potential use cases

The engineering team developed Gridwich to align with principles and industry standards for:

- [Clean monolith architecture](#)
- [Project structure and naming](#)
- [CI/CD](#)
- [Content protection and digital rights management \(DRM\)](#)
- [Azure Storage usage and scaling](#)
- [Logging](#)

The Gridwich system embodies best practices for processing and delivering media assets on Azure. Although the Gridwich system is media-specific, the message processing and eventing framework can apply to any stateless event processing workflow.

## Deploy this scenario

- [Run the admin scripts for Azure permissions.](#)

- Set up a local development environment.

## Next steps

- [Terraform starter project for Azure Pipelines](#)
- [Azure Function with Event Grid and Terraform Sandwich sample](#). Subscribe an Azure Function to Event Grid Events via Terraform, using a Terraform Sandwich.
- [MediaInfoLib with Azure Storage](#). Azure Functions and console samples that use cross-platform .NET Core to retrieve a report on a media file stored in Azure Storage.
- [Event Grid Viewer Blazor](#). An EventGrid Viewer application, using Blazor and SignalR, with Microsoft Entra authorization support.
- [Azure Function with Managed Service Identity for Azure Storage](#). Use Managed Service Identity between Azure Functions and Azure Storage.
- [Updates to existing media-services-v3-dotnet-core-functions-integration sample](#)

## Related resources

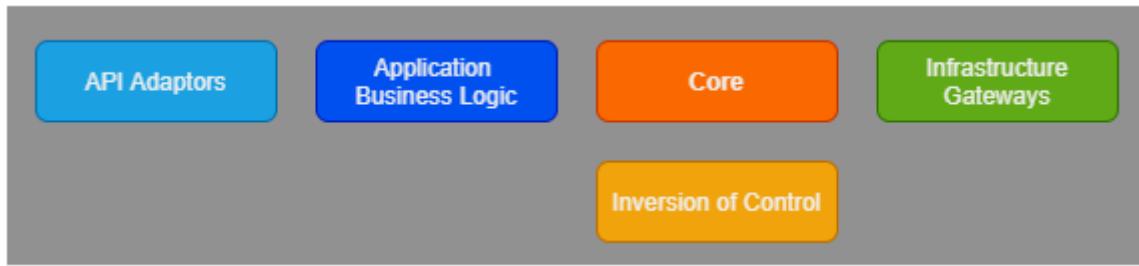
- [Understand Azure Pipelines to Terraform variable flow](#)
- [Set up content protection and DRM](#)
- [Create a new sandbox or test cloud environment](#)
- [Maintain and manage Key Vault keys](#)
- [Scale Media Services resources](#)

# Gridwich clean monolith architecture

Azure Event Grid   Azure Functions

The code in this project is organized as a clean-architecture [monolith](#), with the following typical conceptual components:

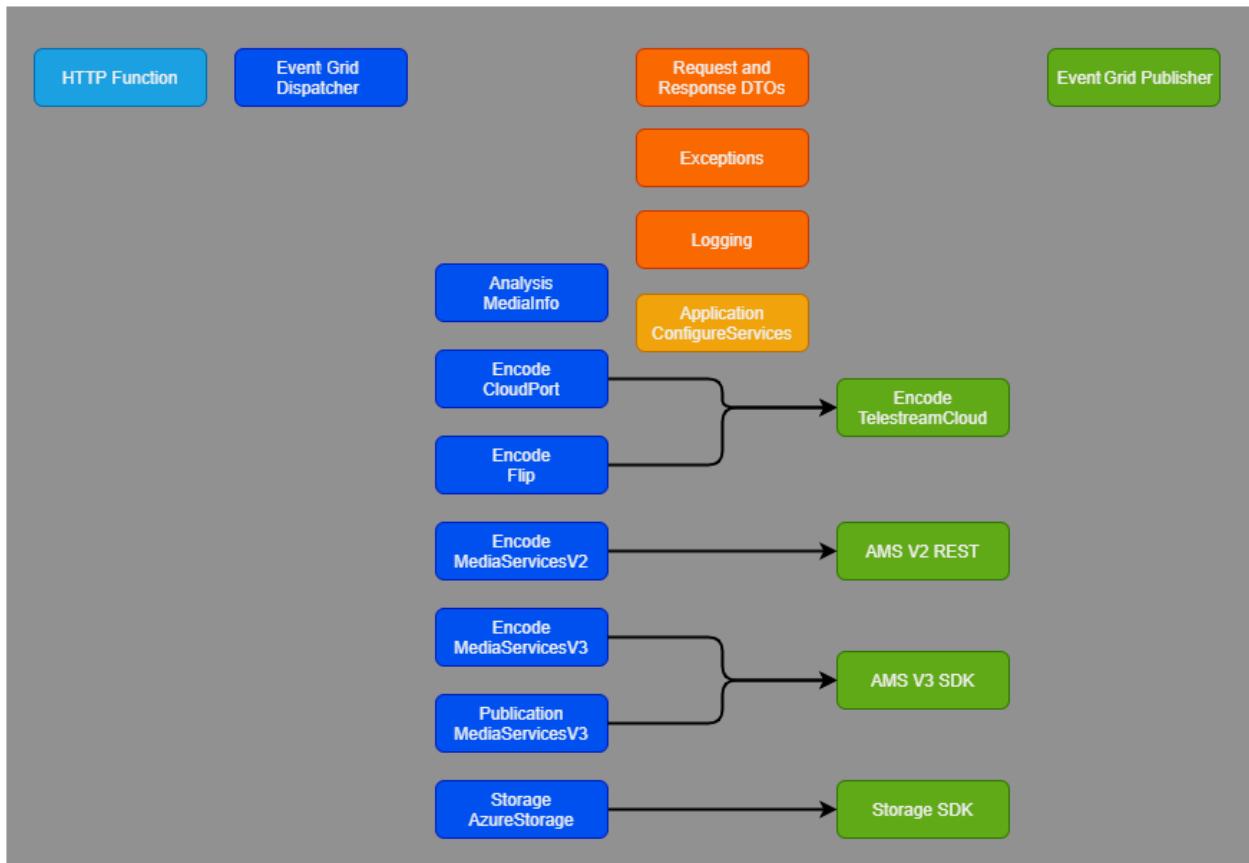
- API adapters
- Decoupled application business logic
- Core domain objects
- Infrastructure gateways
- Inversion of Control (IoC)



The solution is stateless, so it doesn't contain any gateways to persistence layers. The solution has no user interface, so it has no controllers or presenters.

The software component composition uses the [GridwichConfigureServices](#) class to define which concrete classes are available in the IoC container for the Azure Functions App.

## Architecture



The Gridwich solution has a [Core.EventGrid](#) library, which contains:

- The domain request and response data transfer objects (DTOs).
- Interfaces for all application business logic or service objects.
- The base classes that help achieve common domain-driven logic or activities.
- Logging, observability, and exception definitions for use throughout the application.

To encapsulate Azure Event Grid as a request and response broker, the library has:

- An event dispatcher that uses the IoC to identify and dispatch events to listeners.
- An event publisher to place responses on the correct Event Grid topic.

The Event Grid request adapter is an HTTP endpoint in the form of an [Azure Function HTTP Endpoint](#). An adapter to convert web requests to Event Grid arrays is also in the same [EventGridFunction](#).

The Event Grid response gateway consists of:

- The [EventGridHandlerBase](#), which converts a response DTO into an `EventGridEvent` object.
- The [EventGridDispatcher](#), which places the Event Grid event on the correct response Event Grid topic endpoint URI by using the topic key.

The solution decouples the [saga participants](#) into the following libraries, which have responsibilities over domain-specific application business logic. The libraries contain required infrastructure gateways and their SDKs, which accomplish the actions that the business logic requires.

- [Gridwich.SagaParticipants.Analysis.MediaInfo](#) ↗
- [Gridwich.SagaParticipants.Encode.CloudPort](#) ↗
- [Gridwich.SagaParticipants.Encode.Flip](#) ↗
- [Gridwich.SagaParticipants.Encode.MediaServicesV3](#) ↗
- [Gridwich.SagaParticipants.Publication.MediaServicesV3](#) ↗
- [Gridwich.SagaParticipants.Storage.AzureStorage](#) ↗

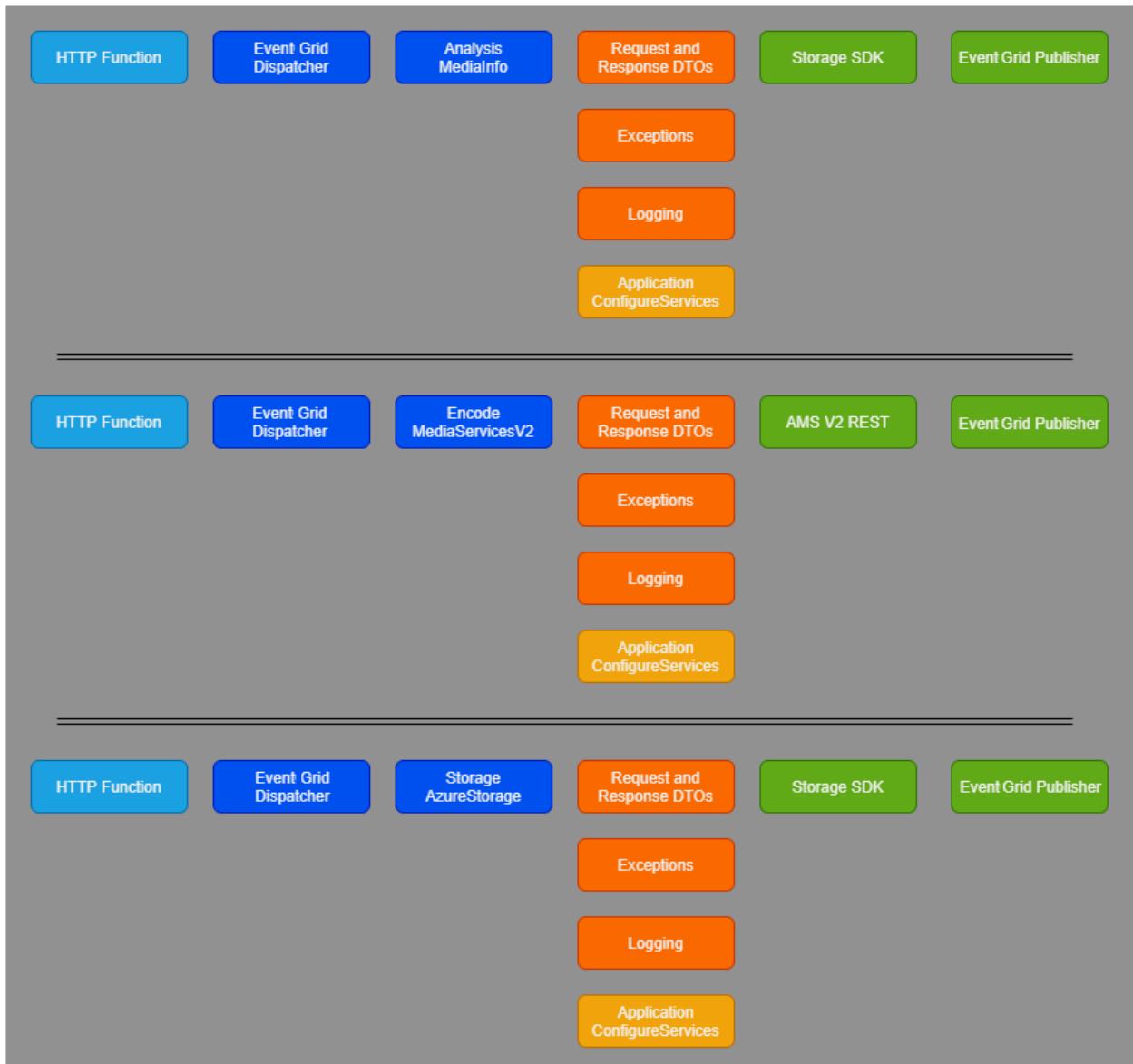
For code reuse and centralization, Gridwich consolidates business logic or infrastructure gateways that several participants use into the following shared libraries:

- [Gridwich.Core.MediaServicesV3](#) ↗
- [Gridwich.SagaParticipants.Encode](#) ↗
- [Gridwich.SagaParticipants.Encode.TelestreamCloud](#) ↗

## Microservices alternative

Nothing in the Gridwich problem space or architecture explicitly pushes the solution into either a monolithic app or several microservices.

You could easily refactor the app into microservices, each a Function App hosting a single saga participant. Each Function app would link the core and core EventGrid libraries. The apps would each have a linkage or use a common library for infrastructure gateways.



The advantage of such a microservices approach is the ability to scale differently for each type of request. If there were thousands of one request type per second, but only hundreds of another request type per day, the overall solution would benefit from having smaller, easy-to-instantiate, and quick-to-execute functions for the high-volume requests.

The drawback of microservices is that any shared models require synchronized rollout of the microservices, or request pool draining and switchover if there's a data schema change. This requirement would complicate future development, continuous deployment, and operations. Since the business problem didn't demonstrate a need for microservices, Gridwich architecture uses a clean monolith approach.

## Next steps

- [What are microservices?](#): Explore microservice architecture.
- [Introduction to Azure Functions](#): Learn more about Azure Functions.

- [Azure Media Services as an Event Grid source](#): Familiarize yourself with the schemas and properties for Media Services events.

## Related resources

- [Understand Gridwich cloud media system](#)
- [Explore Gridwich project naming](#)
- [Set up Gridwich CI/CD pipeline](#)

# Gridwich saga orchestration

Azure Media Services

Azure Storage

In the example implementation, the external system is a large media company's media asset management (MAM) and workflow orchestration system. The external system operates as a [saga orchestrator](#) that chains a series of activities to build Gridwich workflows.

Saga activities might or might not include user interactions or approvals. Gridwich assumes that the external system tracks the failure or success of each operation it initiates.

## Saga participants

Each saga participant contributes one or more work activities to the ecosystem. Each participant works independently, and more than one saga participant might act on a single request.

For Gridwich, the available saga participants are:

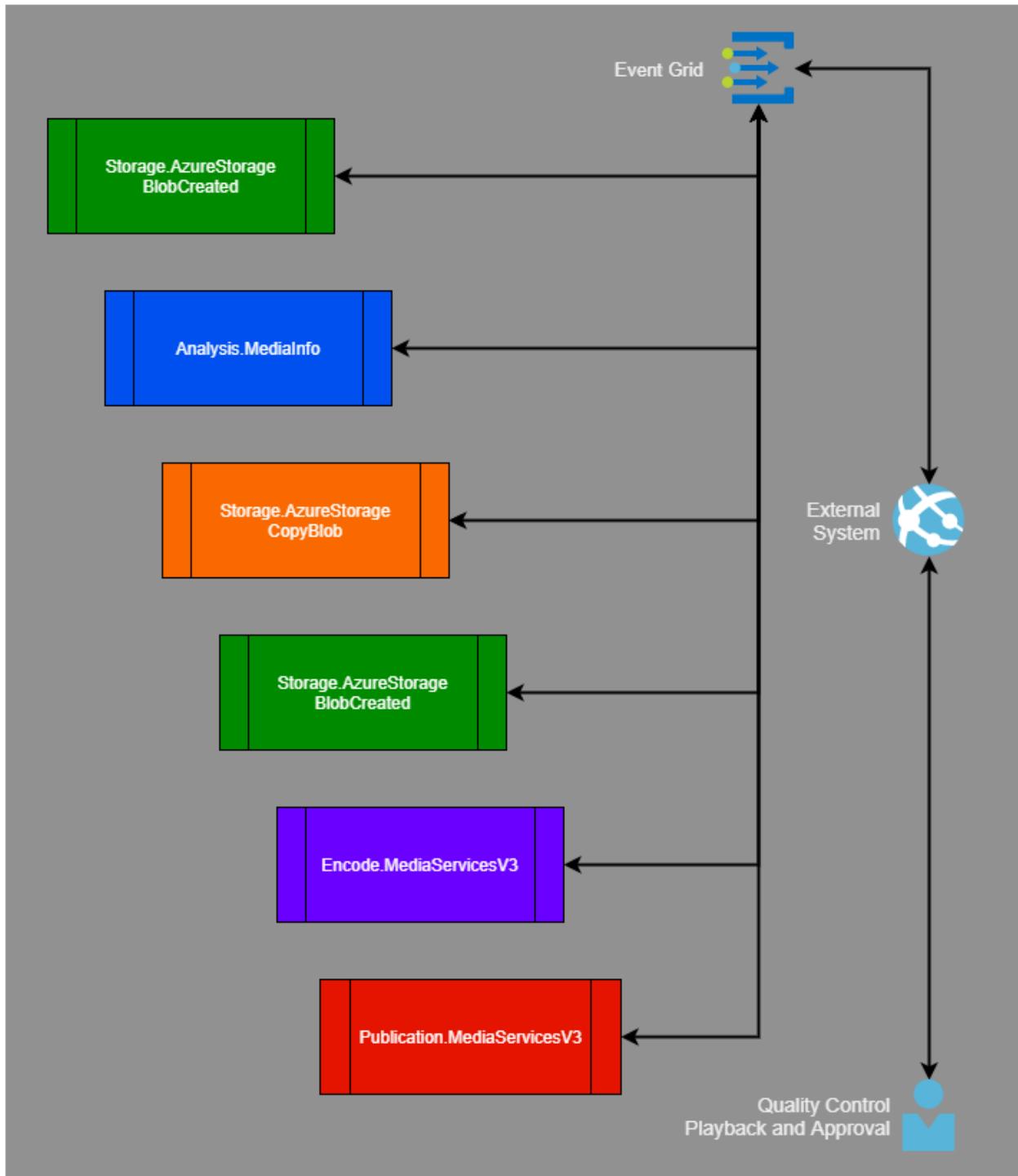
- [Analysis.MediaInfo](#)
- [Encode.CloudPort](#)
- [Encode.Flip](#)
- [Encode.MediaServicesV3](#)
- [Publication.MediaServicesV3](#)
- [Storage.AzureStorage](#)

## Example saga workflow

The external system might run a quality control check saga that does the following steps:

1. Gets a notification of a new blob in the inbox storage account.
2. Requests an analysis using MediaInfo.
3. Reviews the MediaInfo response, auto-approves the file, and starts a copy into an intermediate account.
4. Gets notified that the copy is complete.
5. Starts a multi-bitrate encoding by using Azure Media Services v3 API encoder, requests AAC audio for all tracks, and copies the video codec.

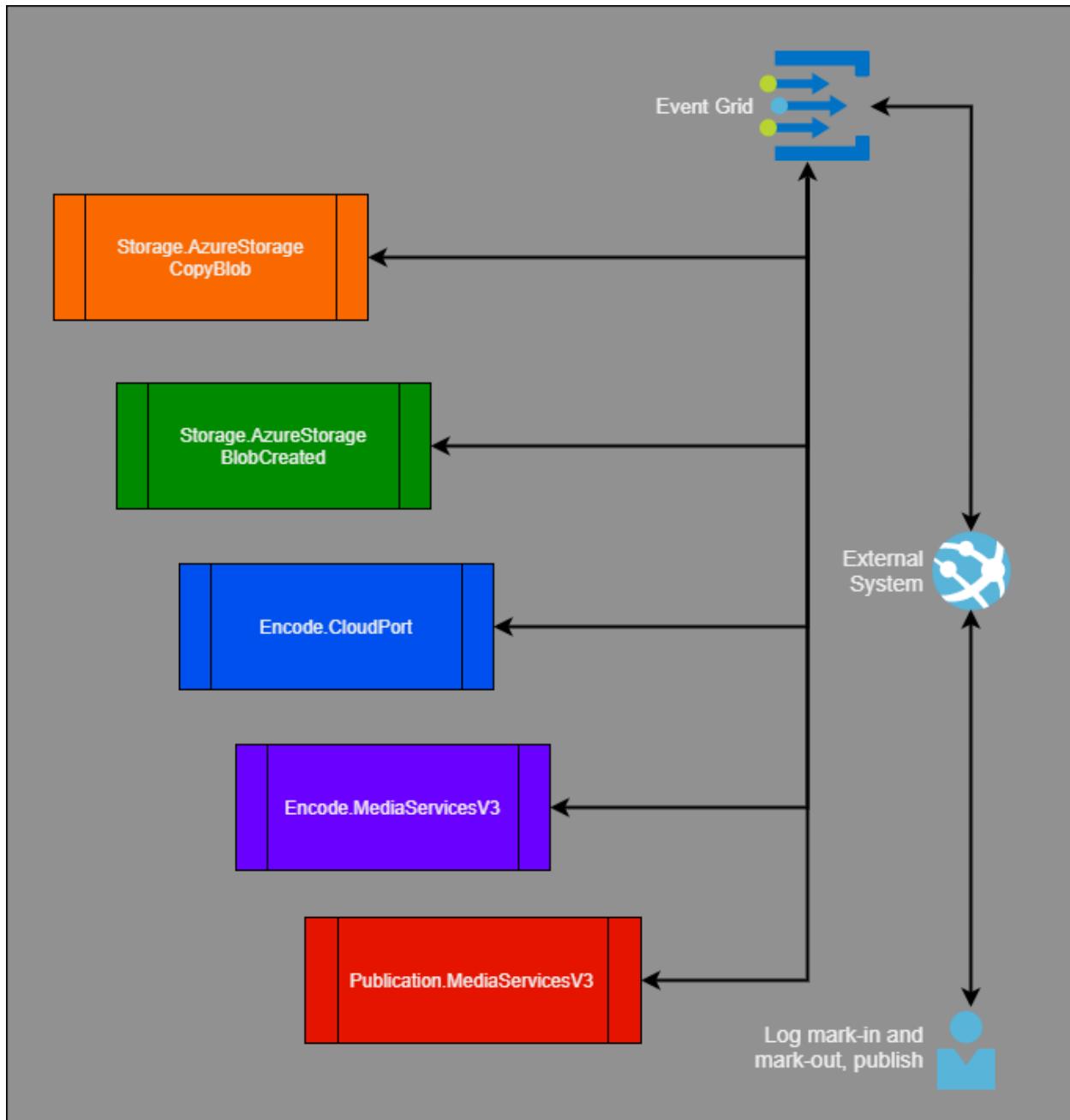
6. Publishes the completed asset using DRM, and notifies an operator that an asset is ready for review.



The operator reviews the asset, identifies the various audio track layouts, and then starts a saga that:

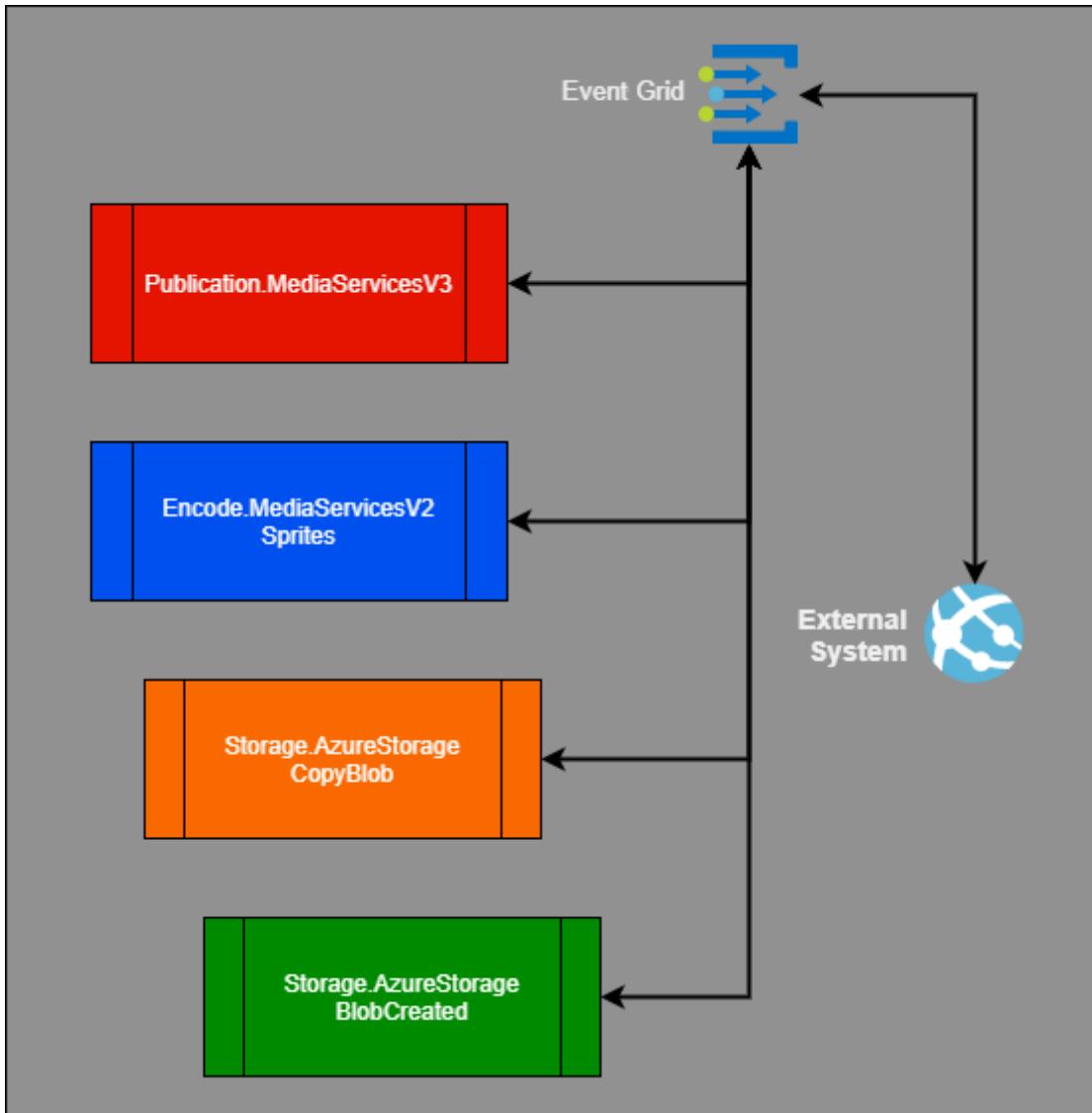
1. Starts a copy into the long-term storage account.
2. Gets notified that the copy is complete.
3. Begins encoding with TeleStream CloudPort to Mux the left and right stereo tracks, along with the video, into a new asset.
4. Creates a multi-bitrate asset by using Azure Media Services v3 API encoder.

5. Publishes the asset with DRM, and notifies the operator that an asset is ready for logging.



The operator reviews the asset contents, extracts metadata for the MAM system, and sets mark-in and mark-out points for one or more features, text-less sequences, or featurelettes. The operator then begins the publication saga, which:

1. Uses the Azure Media Services Publishing v3 API to create a time-based filter for each subasset, and create a locator with that filter and DRM.
2. Simultaneously begins to create sprites for each subasset.
3. After receiving successful responses from both processes, begins a copy of the sprite files into the published asset.
4. Receives the blob created for the copy, and completes the publication flow by updating the MAM system.



## Components

- [Azure Event Grid](#) allows a developer to easily build applications with event-based architectures.
- [Azure Blob storage](#) is a service for storing any type of text or binary data, such as a document, media file, or application installer.
- [Azure Media Services v3 API](#) is a cloud-based platform that enables you to build solutions that achieve broadcast-quality video streaming, enhance accessibility and distribution, analyze content, and much more.

## Next steps

- [Azure Blob storage](#)
- [Azure Event Grid](#)
- [Azure Media Services v3 API](#)
- [Saga](#): Learn more about the Saga distributed transactions pattern.
- [Cloud-native data patterns](#): Explore cloud-native data patterns.

- [Azure Media Services as an Event Grid source](#): Familiarize yourself with the schemas and properties for Media Services events.

## Related resources

- [Understand Gridwich cloud media system](#)
- [Explore Gridwich project naming](#)
- [Set up Gridwich CI/CD pipeline](#)

# Gridwich project naming and namespaces

Azure Functions

Gridwich is a .NET 6 solution composed of multiple projects. It's important for code projects to have a naming convention to help understand application structure, find relevant code quickly, and reduce [bike-shedding](#) in project naming.

The Gridwich system has three major components, `Core`, `Host.FunctionApp`, and `SagaParticipants`.

- The `core` project has system-wide interfaces, models, data transfer objects (DTOs), and base classes.

`Core.{Technology}` projects have the client classes and base functionalities that various capability implementations use.

- The `Host.FunctionApp` project is the public interface to the overall system.
- `SagaParticipants` projects provide external function capabilities like analysis, encoding, publishing, and storage.

`SagaParticipants.{Capability}` projects describe the interfaces, exceptions, and events that a capability produces.

`SagaParticipants.{Capability}.{Technology}` projects provide actual capability implementation, event listeners, and capability-specific functionality.

A Gridwich `Technology` is an actual implementation of a capability or core function. A `{Technology}` project can be under either a `Core` or a `SagaParticipants.{Capability}` namespace and project name, depending on usage.

## Project creation

You can use the following decision tree when naming a new Gridwich project:

Is the code a contract, like base classes, interfaces, models, or DTOs, or a service extension?

- Yes: Does the code relate to a specific capacity or service?

- Yes: `Gridwich.SagaParticipants.{Capability}`
- No: `Gridwich.Core`
- No: Does the code relate to an event listener or an implementation of a specific technology?
  - Yes: Will more than one service use the code?
    - Yes, for example an SDK wrapper: `Gridwich.Core.{Technology}`
    - No: `Gridwich.SagaParticipants.{Capability}.{Technology}`
  - No: Does the code relate to a specific capability?
    - Yes: `Gridwich.SagaParticipants.{Capability}`
    - No: Is the code an Azure Function App endpoint?
      - Yes: `Gridwich.Host.FunctionApp`
      - No: `Gridwich.Core`

## Project structure

Each package has two child subdirectories:

- `src` contains the non-test production code.
- `tests` contains unit tests.

Every project has a `tests` subdirectory, but if there are no unit tests for a package, the directory may be empty.

Each of the two subdirectories contains the C# or other files to build the code, plus a `.csproj` file. The `.csproj` filename follows the package name, for example:

- `Gridwich.Host.FunctionApp/src/Gridwich.Host.FunctionApp.csproj`
- `Gridwich.Host.FunctionApp/tests/Gridwich.Host.FunctionAppTests.csproj`

The code namespaces that the packages use also follow this convention, for example:

- `Gridwich.Host.FunctionApp`
- `Gridwich.Host.FunctionAppTests`

During build and test cycles, transient directories like `bin`, `obj`, and `TestResults` appear, which contain no git-eligible artifacts. The `dotnet clean` processing cleans up these transient directories.

# Project names and namespaces

Gridwich project names and namespaces have the following characteristics.

## Core and SagaParticipants Technology namespaces

`Gridwich.Core.{Technology}` namespaces don't include the purpose of the technology, mainly to avoid *bike-shedding*. `Core` namespaces are internal projects that `SagaParticipants` or `Host.FunctionApp` projects use, and don't need well-defined names.

For example, the `Gridwich.Core.MediaServicesV3` project could have been named `Gridwich.Core.Media.MediaServicesV3` or `Gridwich.Core.Processing.MediaServicesV3`. The `Gridwich.Core.EventGrid` project could be `Gridwich.Core.Events.EventGrid` or `Gridwich.Core.Messaging.EventGrid`. However, the `Core` project names already suggest that the technologies contribute to the core system.

A technology could also contribute to the system in more than one way. For example, you could call Redis a data store or a messaging transport, depending on usage, but it always uses the same SDK wrapper.

The `Gridwich.SagaParticipants.Encode.CloudPort` and `Gridwich.SagaParticipants.Encode.Flip` technology namespaces use components from the `Gridwich.SagaParticipants.Encode` namespace. This code isn't under `Gridwich.Core.Encode` namespace because it's specific to encoding tasks, and doesn't cross into other capabilities like publication.

On the other hand, both `Gridwich.SagaParticipants.Encode.MediaServicesV3` and `Gridwich.SagaParticipants.Publish.MediaServicesV3` use components from `Gridwich.Core.MediaServicesV3`, so those namespaces include the purpose of the technology.

## SagaParticipants packages

Not every `Gridwich.SagaParticipants` package processes external events. Some packages under `Gridwich.SagaParticipants` provide functionality for other saga participants that process external requests.

Besides the `Gridwich.SagaParticipants.Encode` packaging that shares code across multiple encoding technology packages, there are also specialized packages like `Gridwich.SagaParticipants.Encode.TelemetreamCloud`. The Telemetream package provides

Gridwich access to an external Vantage Telestream system. The Flip and CloudPort saga participants use the Telestream package to provide their own request processing.

## Package names and other namespaces

To keep `using` statements to a minimum, Gridwich doesn't restrict package contents to the namespace that the package name indicates. Some packages contribute entities to other namespaces. For example, the package `Gridwich.Core.Tests` contributes the `Gridwich.Core.Helpers.TestHelpers` class.

However, each package builds a DLL that matches the package name for the production code in `src`, and a DLL of unit tests, if any, in `tests`. The test DLL name is the same as the package name, but with a `Tests` suffix.

## Next steps

Product documentation:

- [Gridwich cloud media system](#)
- [Introduction to Azure Functions](#)

Microsoft Learn modules:

- [Create a long-running serverless workflow with Durable Functions](#)
- [Explore Azure Functions](#)

## Related resources

- [Gridwich clean monolith architecture](#)
- [Gridwich content protection and DRM](#)
- [Gridwich saga orchestration](#)

# Gridwich CI/CD pipeline

Microsoft Entra ID   Azure Event Grid   Azure Functions   Azure Key Vault   Azure Pipelines

Gridwich requires multiple resources within and outside Azure to talk to one another securely. This requirement poses continuous integration and continuous delivery (CI/CD) challenges with Microsoft Entra permissions, gates, resource creation, order of operation, and long-running functions deployment. The following guiding principles address these challenges:

- A single build artifact affects all environments in the same pipeline.
- Non-gated environments are disposable.
- Terraform declaratively creates idempotent environments.
- Terraform doesn't release software.
- Infrastructure creation and software release are distinct stages in the pipeline.
- The CI/CD pipeline doesn't assign Microsoft Entra permissions.
- The pipeline considers everything as code.
- The pipeline uses reusable components focused on composability.

For more information about how the Azure Pipelines CI/CD pipelines convert and inject pipeline variables into Terraform modules, and then to Azure Key Vault and Azure Functions app settings, see [Pipelines to Terraform variable flow](#).

The following considerations relate to the preceding principles.

## Single artifact, multiple environments

The Gridwich pipeline scales to multiple environments, but there is only one artifact, which the pipeline promotes from one environment to the next.

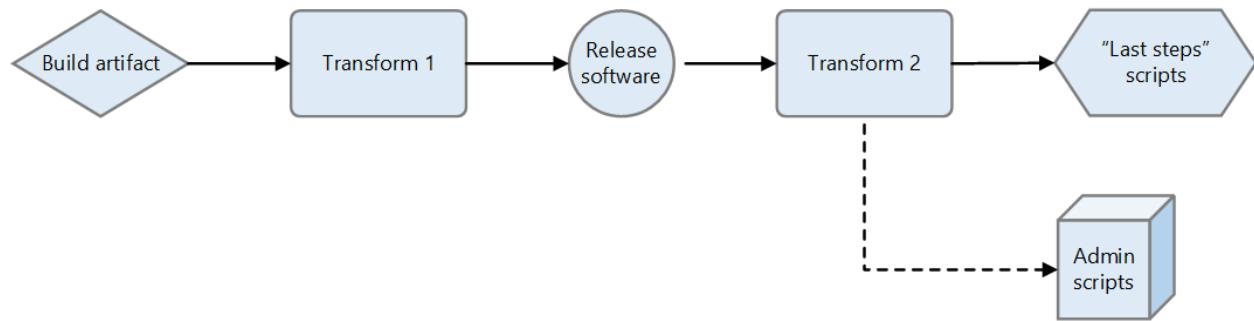
## Software release vs. infrastructure creation

In Gridwich, software release and infrastructure deployment are two separate responsibilities. A single pipeline handles both responsibilities at various stages, using the following general pattern:

**Software builds > Infrastructure deployment > Software release > Software configuration > Custom script deployment**

The guiding principle that infrastructure and software release are two distinct responsibilities makes deploying Event Grid subscriptions more difficult. When Azure creates an Event Grid webhook subscription, it sends a validation event to check whether the registering endpoint accepts Event Grid events. To pass this validation check, the Azure Function must be released and running before Terraform can build the Event Grid subscription resources.

To address this issue, there are two Terraform jobs in the CI/CD pipeline:



- Terraform 1 creates all the resources except for the Azure Event Grid subscriptions.
- Terraform 2 creates the Event Grid subscriptions after the software is up and running.

Because Terraform currently lacks the ability to exclude a specific module, the Terraform 1 job must explicitly target all the modules except the Event Grid subscriptions. This requirement is potentially error prone, and a current [GitHub issue on Terraform](#) tracks this problem.

## Post-deployment scripts

The CI/CD pipeline doesn't do operations that need elevated privileges, but uses [admin script templates](#) to generate a set of admin scripts as pipeline artifacts. An admin with elevated privileges must run these admin scripts whenever a new Gridwich environment is created. For more information, see [Run Azure admin scripts](#).

Terraform and software releases can't complete certain Gridwich operations, including:

- Copying certificates to Azure Key Vault
- Enabling storage analytics in Azure Storage
- Scaling Azure Media Services

The Azure CLI script [azcli-last-steps-template.yml](#) provides these last steps.

# Everything as code and code reuse

One advantage of the "everything as code" practice is component reuse.

- For Terraform, Gridwich relies heavily on [Terraform modules](#) ↗ to enhance composable and reusability.
- For Azure Pipelines YAML, Gridwich uses [Pipeline templates](#).

## Next steps

- [Run the admin scripts](#)
- [Pipeline variables to Terraform flow](#)

# Gridwich operations for Azure Storage

Azure Storage

The Gridwich Azure Storage Service, [Gridwich.SagaParticipants.Storage.AzureStorage](#), provides blob and container operations for Azure Storage Accounts that are configured for Gridwich. Example storage operations are **Create blob**, **Delete container**, **Copy blob**, or **Change storage tier**.

Gridwich requires its storage mechanisms to work for both Azure Storage block blobs and containers. With distinct classes and Storage Service operations for blobs and containers, there's no ambiguity about whether a given storage operation relates to a blob or to a container. This article applies to both blobs and containers, except where noted.

Gridwich exposes most storage operations to external systems within the [Storage.AzureStorage](#) [saga participant](#). Other saga participants use the storage service for tasks like copying blobs between different containers or accounts when they set up encoding workflows.

This article describes how the Gridwich Azure Storage Service meets solution requirements and integrates with mechanisms like event handlers. Links point to the corresponding source code, which contains more extensive commentary on the containers, classes, and mechanisms.

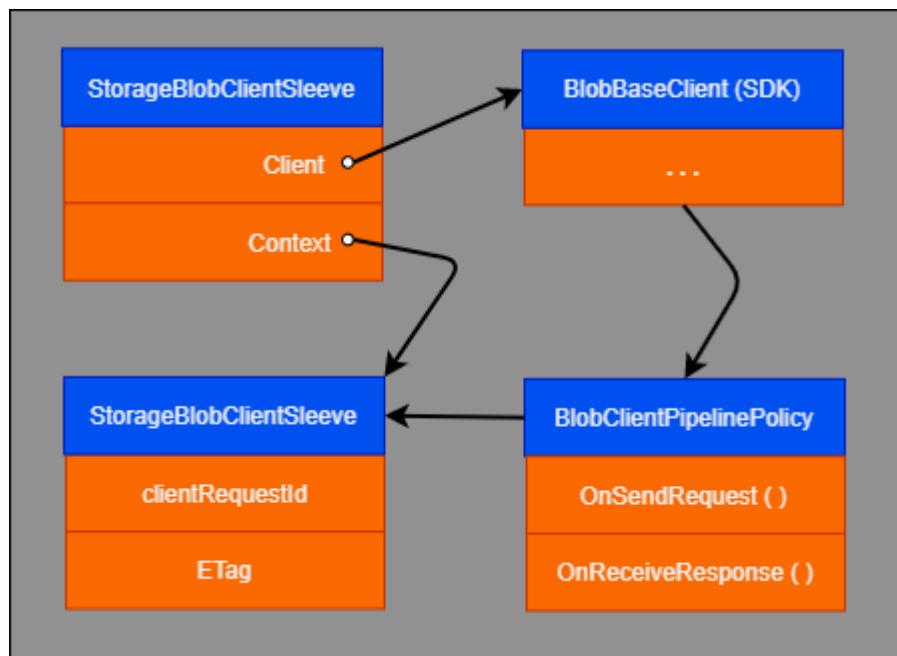
## Azure Storage SDK

Gridwich uses classes from the Azure Storage SDK to interact with Azure Storage, rather than handcrafting REST requests. Within the storage provider, the SDK [BlobBaseClient](#) and [BlobContainerClient](#) classes manage storage requests.

These SDK client classes currently allow only indirect access to the two HTTP headers Gridwich needs to manipulate, `x-ms-client-request-id` for operation context and `ETag` for object version.

In Gridwich, a pair of provider classes dispense [BlobBaseClientProvider](#) and [BlobContainerClientProvider](#) functionality in units called *sleeves*. For details about sleeves, see [Storage sleeves](#).

The following diagram illustrates the structure of the SDK and Gridwich classes, and how instances relate to each other. The arrows indicate "has a reference to."



## Pipeline policy

You set the hook to manipulate the HTTP headers as a pipeline policy instance when you create the client instance. You can set this policy only at client instance creation time, and you can't change the policy. The storage provider code using the client must be able to manipulate the header values during execution. The challenge is to make the storage provider and pipeline interact cleanly.

For the Gridwich pipeline policy, see the [BlobClientPipelinePolicy](#) class.

## Storage Service caching

TCP connection establishment and authentication create overhead when an SDK client object instance sends its first request to Azure Storage. Multiple calls to the same blob in an external system request, for example **Get Metadata**, then **Delete blob**, compound the overhead.

To mitigate overhead, Gridwich maintains a cache of one client instance for each storage blob or container, depending on the SDK classes the operation context uses. Gridwich retains this client instance and can use the instance for multiple Azure Storage operations against the same blob or container for the duration of an external system request.

The Azure SDK-provided client classes require SDK client object instances to be specific to a single blob or container at creation time. The instances also aren't guaranteed safe for simultaneous use on different threads. Since an [operation context](#) represents a single request, Gridwich bases caching on the combination of blob or container name with operation context.

This instance reuse, combined with the Azure Storage SDK client structure, requires additional support code to balance efficiency and code clarity.

## Context argument

Almost all the Gridwich Storage Service operations require a special context argument of type [StorageClientProviderContext](#). This context argument fulfills the following requirements:

- Provides the external system with responses, which include the per-request unique JSON-based *operation context* value that the external system specified on the Gridwich request. For more information, see [Operation context](#).
- Allows Storage Service callers like Gridwich event handlers to control which responses are visible to the external system. This control prevents the service from flooding the external system with irrelevant notification events. For more information, see [Context muting](#).
- Complies with Azure Storage conventions to ensure coherent requests and responses in an environment that allows a mix of parallel readers and writers. For example, supports [ETag tracking](#). For more information, see [ETags](#).

## Storage context

The context for both the blob and container [storage types](#) is the [StorageClientProviderContext](#), which looks like:

C#

```
string ClientRequestID { get; }
JObject ClientRequestIdAs JObject { get; }
bool IsMuted { get; set; }
string ETag { get; set; }
bool TrackingETag { get; set; }
```

The first two properties are different representations of the operation context that was used to initialize the [StorageClientProviderContext](#) instance. The class has a variety of

constructors, including a copy constructor. Additional methods include `ResetTo`, to allow in-place state duplication, and a static `CreateSafe` method to ensure that problematic initializations don't throw exceptions.

The class also contains special handling for creating contexts based on GUIDs and empty strings. The Azure Storage Notification handlers for blob [Created](#) and [Deleted](#), which also process notifications arising from external agents, require the GUID form.

## Context muting

The `IsMuted` property controls whether the application expects the service to publish resulting notifications back to the caller, for example to the external system. In a muted operation, the service doesn't publish resulting events.

An example is blob copies that an encoder executes to arrange blobs in Azure Storage as input to an encoding task. The external system isn't concerned about these details, but only about the status of the encoding job and where it can retrieve the encoded outputs. To reflect these concerns, the encoder:

1. Creates a non-muted storage context based on the request operation context, for example `ctxNotMuted`.
2. Creates a muted storage context, for example `ctxMuted`, by either using the [context class](#) copy constructor or making a new instance. Either option will have the same operation context value.
3. Specifies `ctxMuted` for storage operations involved in the setup for encoding. The external system doesn't see any indication of these operations occurring.
4. Specifies the `ctxNotMuted` context for storage operations that reflect encoding completion, for example copying an output file to a target container. Gridwich handlers publish the resulting Azure Storage notification events to the external system.

The caller controls the ultimate visibility of operations. Both muted and non-muted operations are based on an equivalent `operationContext` value. The intent of context muting is to make it easier to perform issue diagnosis from event tracing logs, because it's possible to see the storage operations related to a request, regardless of operation muting status.

The [ResponseBaseDTO](#) has a boolean property `DoNotPublish`, which event dispatching uses to dictate the final decision about whether to publish. Event dispatching, in turn,

sets the `DoNotPublish` property based on the `IsMuted` property of the context.

The service transmits the muting setting to Azure Storage, which then sets the `clientRequestId` in the storage notification events it presents to the two Gridwich handlers, [Created](#) and [Deleted](#). Those two handlers set `DoNotPublish` to reflect the caller-requested muting.

## ETags for target consistency

Azure Storage uses the HTTP `ETag` header for request sequences that should have target consistency. An example is to ensure a blob hasn't changed between [Retrieve Metadata](#) and [Update Metadata](#) storage operations.

To align with standard HTTP usage, this header has an opaque value whose interpretation is that if the header value changes, then the underlying object has also changed. If a request sends its current `ETag` value for the object, and it doesn't match the current Storage Service `ETag` value, the request immediately fails. If the request doesn't include an `ETag` value, Azure Storage skips that check and doesn't block the request.

## ETags in the Storage Service

For Gridwich, the `ETag` is an internal detail between the Gridwich Storage Service and Azure Storage. No other code needs to be aware of the `ETag`. The Storage Service uses the `ETag` for sequences like the [Get Blob Metadata](#), [Delete Blob](#) operations for processing a [BlobDelete Event](#) request. Using the `ETag` ensures that the [Delete Blob](#) operation targets exactly the same version of the blob as the [Get Metadata](#) operation.

To use the `ETag` for the preceding example:

1. Send the [Get Metadata](#) request with a blank `ETag`.
2. Save the `ETag` value from the response.
3. Add the saved `ETag` value to the [Delete Blob](#) request.

If the two `ETag` values are different, the delete operation fails. The failure implies that some other operation changed the blob between steps 2 and 3. Repeat the process from step 1.

`ETag` is a parameter of constructors and a string property of the [StorageClientProviderContext class](#). Only the Gridwich-specific [BlobClientPipelinePolicy](#) manipulates the `ETag` value.

## Control ETag use

The `TrackingETag` property controls whether to send the `ETag` value on the next request. The value `true` means that the service sends an `ETag` if one is available.

An Azure Storage request with an `ETag` value that doesn't match the subject blob or container results in the operation failing. This failure is by design, because `ETag` is the standard HTTP way of expressing "the exact version that the request is targeting." Requests can include the `TrackingETag` property to state that the `ETags` must match, or not include the `TrackingETag` property to indicate that the `ETag` values don't matter.

The pipeline always retrieves an `ETag` value from an Azure Storage operation if one is present in that REST response. The pipeline always updates the context `ETag` property, if possible, as of the last operation. The `TrackingETag` flag controls only whether the next request from the same client instance sends the value of the `ETag` property. If the `ETag` value is null or empty, the current request sets no HTTP `ETag` value, regardless of the value of `TrackingETag`.

## Storage sleeves

Gridwich requires that its storage mechanisms work for both Azure Storage block blobs and containers. There are distinct classes and Storage Service operations for blobs and containers, so there's no ambiguity about whether a given storage operation relates to a blob or to a container.

A pair of provider classes, one for [blobs](#) and one for [containers](#), dispense the two sets of functionality in units called *sleeves*. Sleeves contain instances of storage helper classes that are part of the Azure SDK. Initializing the Storage Service creates the providers and makes them directly available to Storage Service methods.

## Sleeve structure

The *sleeve* is a container for the SDK Client object instance and a storage context. Storage provider functions reference the sleeve via the two properties `Client` and `Context`. There is a sleeve type for [blobs](#) and another for [containers](#), which have `Client` properties of type [BlobBaseClient](#) and [BlobContainerClient](#), respectively.

The general sleeve structure for blobs looks like:

C#

```
BlobBaseClient Client { get; }
BlobServiceClient Service { get; }
StorageClientProviderContext Context { get; }
```

The `Service` property on the sleeve is a convenience. Some of the final encoder-related operations that use the [SDK BlobServiceClient class](#) require Storage Account keys. This requirement led to adding a Service client instance to the two existing sleeve types, rather than producing a separate provider.

## Sleeve usage

The client storage providers dispense sleeve instances. The storage service code looks similar to the following annotated code sequence, with types spelled out for clarity:

C#

```
public bool DeleteBlob(Uri sourceUri, StorageClientProviderContext
context)
{
    . . .
    StorageBlobClientSleeve sleeve =
    _blobBaseClientProvider.GetBlobBaseClientForUri(sourceUri, context); // Line
A
    BlobProperties propsIncludingMetadata =
    sleeve.Client.GetProperties(); // Line B
    sleeve.Context.TrackingETag = true; // Send ETag from
    GetProperties()
    var wasDeleted = sleeve.Client.DeleteBlob(); // Line C
    sleeve.Context.TrackingETag = false;
    var someResult = sleeve.Client.AnotherOperation(); // Line D
    . . .
}
```

1. Gridwich auto-populates the operation context into the sleeve context at line A. `TrackingETag` defaults to false.
2. After Line B, `sleeve.Context` contains the `ETag` from line A and retains the same `ClientRequestID` value.
3. Line C sends both the `ETag` value from Line B and the `ClientRequestId`.
4. After Line C, the context has a new `ETag` value, as returned in the `Delete()` response.
5. Line D doesn't send an `ETag` value on the request for `AnotherOperation()`.
6. After Line D, the context has a new `ETag` value, as returned in the `AnotherOperation()` response.

The Storage Service is currently set as `Transient` in the [dependency injection configuration](#), which implies that the sleeve-based caching is on a per-request basis. See [Storage Service and dependency injection](#) for more information.

## Storage Service alternatives

The following sections describe alternative approaches that aren't part of the current Gridwich storage solution.

### Gridwich AzureStorageManagement class

In conjunction with the sleeve `Service` member, which is an instance of the [Azure SDK BlobServiceClient class](#), Gridwich also has the [AzureStorageManagement](#) class. The Storage Service `GetConnectionStringForAccount` method and the Telerek encoding's `GetStoreByNameAsync` method use that class to get storage account keys. The class is currently based on the Fluent framework. Additions to the SDK `BlobServiceClient` class should eventually supersede this class, allowing for a more focused information retrieval than the wide variety in the [Fluent IAzure interface](#).

### Hide the pipeline policy via subclassing

Subclassing the SDK client types adds two simple properties to the client, one for each HTTP header value, to completely hide the interaction with the pipeline policy. But because of a deep [Moq](#) bug, it's not possible to create unit tests via `mock` for these derived types. Gridwich uses Moq, so didn't use this subclassing approach.

The Moq bug relates to its mishandling of cross-assembly subclassing in the presence of internal-scope virtual functions. The SDK client classes make use of internal-scope virtual functions involving internal-scope types that are invisible to normal outside users. When Moq tries to create a `mock` of the subclass, which is in one of the Gridwich assemblies, it fails at test execution time as it can't find the internal-scope virtuals in the SDK client classes from which the Gridwich classes are derived. There is no workaround without changes in the Moq Castle proxy generation.

### Storage Service and dependency injection

Gridwich currently registers the Storage Service as a `Transient` dependency injection service. That is, each time dependency injection is asked for the service, it creates a new instance. The current code should also work correctly if the registration changes to `Scoped`, implying one instance per request, for example the external system's request.

However, there will be issues if the registration changes to `Singleton`, one instance across the Gridwich Function app. The Gridwich caching mechanism for sleeves and data byte ranges then won't distinguish between different requests. Also, the cache model isn't a check-out one, so Gridwich doesn't remove the instance from the cache while it's in use. Since the SDK client classes aren't guaranteed to be thread-safe, coordination would require a number of changes.

For these reasons, don't change the Gridwich Storage Service, as is, to `Singleton` dependency injection registration. Gridwich follows this rule in [dependency injection registration](#) and includes a unit test, [CheckThatStorageServiceIsNotASingleton](#), to enforce it.

## Next steps

Product documentation:

- [Gridwich cloud media system](#)
- [What is Azure Blob storage?](#)
- [What is Azure Pipelines?](#)

Microsoft Learn modules:

- [Configure blob storage](#)
- [Explore Azure Storage services](#)

## Related resources

- [Gridwich content protection and DRM](#)
- [Gridwich project naming and namespaces](#)
- [Logging in Gridwich](#)

# Logging in Gridwich

Azure

Best practices for logging include:

- Don't use string formatting or interpolation. Logging a string with  `$"This broke {brokenThing}"` isn't useful for debugging.
- Pass objects so that they become searchable fields.

Instead of `LogInformation(LogEventIds.StartProcessing, $"Processing has started on item {Item.itemNumber}");`, use:

- An object as such, like `LogInformationObject(LogEventIds.StartProcessing, Item);`, or
- Anonymous objects, like `LogInformationObject(LogEventIds.StartProcessing, new {Item.itemNumber, Item.itemDescription, someData});`.
- Follow the EventId numbering conventions outlined in [LogEventIds](#).
- For any significant body of work, use the following logging pattern:
  - Use `LogInformationObject` on entry, for example when about to start an encode job.
  - Use `LogInformationObject` on success, for example when the encode job is successful.
  - Use `LogWarningObject`, `LogErrorObject`, or `LogCriticalObject` on failure, for example if the encoding job fails. Use Exception method variants if applicable.
- Although you can log any information at any stage, don't pollute logs with extraneous noise.

## ObjectLogger

[ObjectLogger](#) with [IObjectLogger](#) is a small wrapper utility for the standard Logger/ILogger. This one-liner utility logs any C# object by converting C# objects to dictionary objects that the logger can consume.

ObjectLogger/IObjectLogger restricts use of logger methods that don't have `EventIds` by using an adapter pattern, rather than inheritance. This restriction forces developers to use `EventIds`, which are useful for debugging.

Other recommendations for using ObjectLogger include:

- Don't bypass IObjectLogger by using ILogger.
- Use IObjectLogger with the proper type for your class: `IObjectLogger<myClass>`.
- In any exception handling catch block, use the IObjectLogger methods that include exceptions. Logging providers like Application Insights can use the exception information.

## Logging schema and data

The underlying Event Grid runtime infrastructure provides a base schema. The [Event Grid event schema](#) includes the event time, device of origin, severity level, and string message. Logger/ILogger default custom properties include `EventId`, `Category`, and `RequestPath`.

## Context objects

To work with complex APIs and workflows that require several inputs and outputs, you can create a *context object*, which is a property bag of important variables that your code can pass or generate. Context objects can deal with many parameters, and method signatures don't need to change when you add or remove parameters. You can also pass context objects to the logger and other interfaces as a unit.

For example, instead of:

C#

```
var store = new StorageBlob();
var tier = req.Query["tier"];
var result = await store.SetBlobStorageTier(blobName, tier);
logger.LogInformationObject(LogEventIds.setBlobProperties, result);
```

You can code:

C#

```
var storageContext = new StorageContext();
storageContext.Store = new StorageBlob();
storageContext.Tier = req.Query["tier"];
storageContext.Result = await store.SetBlobStorageTier(blobName,
storageContext.Tier);
logger.LogInformationObject(LogEventIds.setBlobProperties, storageContext);
```

# Log levels

Assigning the appropriate logging level may not be straightforward. The following general descriptions of log levels is from [LogLevel Enum](#).

[Expand table](#)

LogLevel	Enum	Description
LogTrace	0	Contains the most detailed messages and may contain sensitive application data. These messages are turned off by default and shouldn't be turned on in a production environment.
LogDebug	1	Used for interactive investigation during development. These logs primarily contain information that is useful for debugging and have no long-term value.
LogInformation	2	Tracks the general flow of the application. These logs should have long-term value.
.LogWarning	3	Highlights an abnormal or unexpected event in the application flow, but doesn't stop application execution.
.LogError	4	Logs when the current flow of execution is stopped due to a failure. These logs should indicate failures in the current activity, not an application-wide failure.
LogCritical	5	Describes an unrecoverable application or system crash, or a catastrophic failure that requires immediate attention.
LogNone	6	Not used for writing log messages. Specifies that a logging category should not write any messages.

## Next steps

Product documentation:

- [Gridwich cloud media system](#)
- [Azure Storage analytics logging](#)

Microsoft Learn modules:

- [Configure blob storage](#)
- [Explore Azure Storage services](#)

## Related resources

- [Gridwich operations for Azure Storage](#)
- [Gridwich project naming and namespaces](#)
- [Gridwich request-response messages](#)

# Gridwich request-response messages

Azure Logic Apps   Azure Media Services   Azure Storage

This article details the specific Event Grid events that form the request-response sequence for different Gridwich operations.

## Gridwich events

Gridwich Acknowledgment and Gridwich Failure are different from other Gridwich events. Specifically:

- [Gridwich Acknowledgment \(ACK\)](#) indicates that Gridwich has received, but not necessarily processed, the request in a Request-ACK-Response sequence.
- Each operation has one or more unique Success response events, but almost all operations use the same [Gridwich Failure](#) event to communicate failure.

### Publishing events

- [Publish via Azure Media Services](#)
- [Create asset locator](#)
- [Delete asset locator](#)

### Encoding events

- **Initiate new Encode job**
  - [Encode with Media Services](#)
  - [Encode with CloudPort workflow](#)
  - [Encode with Flip](#)

The immediate response event from each encoder, aside from an ACK, is either a [Failure](#) or an [Encoding dispatched](#) event that indicates successful queuing of the job. The [Encoding progress notification events](#) handle further progress.

- **Encoding progress notifications**

All encoders use the same set of [progress notification status events](#).

- [Encoding scheduled](#)
- [Encoding in process](#)
- [Encoding completed successfully](#)

- [Encoding canceled](#)

## Blob and container storage events

- **Containers**
  - [Create container](#)
  - [Delete container](#)
  - [Change access or visibility level](#)
- **Blobs**
  - [Set blob metadata](#)
  - [Copy blob](#)
  - [Delete blob](#)
  - [Change blob access tier](#)
  - [Get blob SAS URL](#)
  - [Analyze blob](#), for example via [MediaInfo](#)
- **Blob notifications**
  - [Blob created](#)
  - [Blob deleted](#)

## Storage keys

- [Rotate storage keys](#)

# Operation context

Gridwich accepts a JSON `operationContext` object as part of request messages. In general, Gridwich echoes a corresponding object in response messages and isn't concerned with the specific internal structure or content of the context object.

The exception is that the response context object may have extra JSON properties compared to the request equivalent. These extra properties are internal to Gridwich, and their names always start with the tilde `~` character. The request properties are always present on the response context object.

As in normal JSON, the response object properties may appear in a different order than in the request object.

For more information about operation context, see [Operation context](#) in the Gridwich Architecture article.

# Event Grid messages

For more information about request-response message flow, see [the architecture request flow](#).

In the following event descriptions, the JSON property values are the usual string, number, or boolean types. The descriptions use the following specific string content types. If the description includes "opaque," the content and format of the value are arbitrary.

- **GUID-string**, like `"b621f33d-d01e-0002-7ae5-4008f006664e"` is a 16-byte ID value spelled out to 36 characters (32 hex digits, plus 4 dashes). Note the lack of curly braces. The value is case-insensitive. This format corresponds to the result of [System.Guid.ToString\("D"\)](#).
- **Topic-string**, like `"/subscriptions/5edeadbe-ef64-4022-a3aa-133bfef1d7a2/resourceGroups/gws-shared-rg-sb/providers/Microsoft.EventGrid/topics/gws-gws-egt-sb"`, is a string of opaque content.
- **Subject-string**, like `"/blobServices/default/containers/telestreamoutput/blobs/db08122195b66be71db9e54ae04c58df/503220533TAGHD23976fps16x990266772067587.mxf"`, is a string of opaque content.
- **EventType-string**, like `"request.operation.requested"` is generally a string of the form: `{"request" | "response"}.operation[.qualifier]`.
- **DataVersion-string**, like `"1.0"`, is a versioning indicator that message processors use to distinguish different evolutions of the same operation. Gridwich requires this field. The `HandlesEvent` method determines which versions an individual Event Grid Handler can process.
- **URL-string** is an absolute URL that often points to Application Insights logs. These strings are usually a SAS URL, due to target authorization requirements.
- **StorageURL-string** is an absolute URL that often points to an Azure Storage Blob or container. This string isn't usually a SAS URL.
- **StorageURL-SAS-string** is an absolute SAS URL that often points to an Azure Storage Blob or container.
- **OperationContextObject**, like `{ "prodID": 10, "dc": "abc" }`, is an arbitrary JSON object that is accepted on incoming requests and echoed back as part of Gridwich response events.
- **Metadata-Dictionary** is a string-to-string JSON object dictionary with the name-value pairs representing Azure Storage Blob metadata.
- **Encoder-Context** is an opaque JSON object of properties specific to a particular encoder.

## Gridwich generic ACK response

Gridwich > Requester, uses [ResponseAcknowledgeDTO](#).

JSON

```
{  
  "id": "GUID-string",  
  "topic": "Topic-string",  
  "subject": "Subject-string",  
  "dataVersion": "DataVersion-string",  
  "data": {  
    "operationContext": <OperationContextObject>,  
    "eventType": "request.blob.metadata.create"  
  },  
  "eventType": "response.acknowledge"  
}
```

The `data.eventType` string value is the top level `eventType` property from the Request event. For example, for a blob analysis request, the `data.eventType` string value is `request.blob.analysis.create`.

## Gridwich generic Failure response

Gridwich > Requester, uses [ResponseFailureDTO](#).

JSON

```
{  
  "id": "GUID-string",  
  "topic": "Topic-string",  
  "subject": "Subject-string",  
  "dataVersion": "DataVersion-string",  
  "data": {  
    "operationContext": <OperationContextObject>,  
    "logEventId": 30001,  
    "logEventMessage": "the message text for eventId 30001",  
    "logRecordId": "GUID-string",  
    "logRecordUrl": "URL-string", // AppInsight URL to the logRecordId  
    "eventHandlerClassName": "string",  
    "handlerId": "GUID-string"  
  },  
  "eventType": "response.failure"  
}
```

The Failure event doesn't include the original request `eventType` value, but does include the operation context and the handler name that was processing the request. The `log*`

properties relate to the problem information that the configured Application Insights instance recorded.

For a limited set of operations, the Failure event object differs significantly from the preceding message. For more information, see [Roll storage keys](#).

## Requester asks Gridwich to place some metadata onto a blob

Requester > Gridwich uses [RequestBlobMetadataCreateDTO](#).

JSON

```
{  
  "id": "GUID-string",  
  "topic": "Topic-string",  
  "subject": "Subject-string",  
  "dataVersion": "DataVersion-string",  
  "data": {  
    "operationContext": <OperationContextObject>,  
    "blobUri": "StorageURL-string",  
    "blobMetadata": <Metadata-Dictionary>  
  },  
  "eventType": "request.blob.metadata.create"  
}
```

The `blobMetadata` is an object of string-valued properties representing all of the name-value pairs of the desired blob metadata.

Gridwich > Requester uses [ResponseBlobMetadataSuccessDTO](#).

JSON

```
{  
  "id": "GUID-string",  
  "topic": "Topic-string",  
  "subject": "Subject-string",  
  "dataVersion": "DataVersion-string",  
  "data": {  
    "operationContext": <OperationContextObject>,  
    "blobUri": "StorageURL-string",  
    "blobMetadata": <Metadata-Dictionary>  
  },  
  "eventType": "response.blob.metadata.success"  
}
```

To later retrieve the current metadata for a blob, see the [Analyze blob](#) request.

# Requester asks Gridwich to perform an analysis of a blob via MediaInfo

Requester > Gridwich uses [RequestBlobAnalysisCreateDTO](#).

JSON

```
{  
  "id": "GUID-string",  
  "topic": "Topic-string",  
  "subject": "Subject-string",  
  "dataVersion": "DataVersion-string",  
  "data": {  
    "operationContext": <OperationContextObject>,  
    "blobUri": "StorageURL-string",  
    "analyzerSpecificData": {  
      "mediaInfo": {  
        "commandLineOptions": {  
          "Complete": "1",  
          "Output": "JSON"  
        }  
      }  
    }  
  },  
  "eventType": "request.blob.analysis.create"  
}
```

Gridwich > Requester uses [ResponseBlobAnalysisSuccessDTO](#).

JSON

```
{  
  "id": "GUID-string",  
  "topic": "Topic-string",  
  "subject": "Subject-string",  
  "dataVersion": "DataVersion-string",  
  "data": {  
    "operationContext": <OperationContextObject>,  
    "blobUri": "StorageURL-string",  
    "blobMetadata": <Metadata-Dictionary>,  
    "analysisResults": <Analysis-Result-Object>  
  },  
  "eventType": "response.blob.analysis.success"  
}
```

The `analysisResults` object's content isn't specified. In the current project, it's the MediaInfo output.

The `blobMetadata` value is a string > string dictionary object of string-valued properties representing all of the name-value pairs of the specified blob's metadata.

As usual with Azure Storage, metadata item names must conform to C# identifier naming rules. For more information, see the Azure [SetBlobMetadata REST API](#) and the [C# naming rules](#).

## Requester asks Gridwich to copy a blob to a new destination

Requester > Gridwich uses [RequestBlobCopyDTO](#).

JSON

```
{  
  "id": "GUID-string",  
  "topic": "Topic-string",  
  "subject": "Subject-string",  
  "dataVersion": "DataVersion-string",  
  "data": {  
    "operationContext": <OperationContextObject>,  
    "sourceUri": "StorageURL-string",  
    "destinationUri": "StorageURL-string"  
  },  
  "eventType": "request.blob.copy"  
}
```

Gridwich > Requester uses [ResponseBlobCopyScheduledDTO](#).

JSON

```
{  
  "id": "GUID-string",  
  "topic": "Topic-string",  
  "subject": "Subject-string",  
  "dataVersion": "DataVersion-string",  
  "data": {  
    "operationContext": <OperationContextObject>,  
    "sourceUri": "URL-string",  
    "blobMetadata": <Metadata-Dictionary>,  
    "destinationUri": "StorageURL-string"  
  },  
  "eventType": "response.blob.copy.scheduled"  
}
```

## Gridwich tells requester that it created a blob

Gridwich could have created the blob from any source, like a copy result, inbox arrival, or encode result.

Gridwich > Requester, uses [ResponseBlobCreatedSuccessDTO](#).

JSON

```
{  
  "id": "GUID-string",  
  "topic": "Topic-string",  
  "subject": "Subject-string",  
  "dataVersion": "DataVersion-string",  
  "data": {  
    "operationContext": <OperationContextObject>,  
    "blobUri": "StorageURL-string",  
    "blobMetadata": <Metadata-Dictionary>  
  },  
  "eventType": "response.blob.created.success"  
}
```

## Requester asks Gridwich to delete a blob

Requester > Gridwich uses [RequestBlobDeleteDTO](#).

JSON

```
{  
  "id": "GUID-string",  
  "topic": "Topic-string",  
  "subject": "Subject-string",  
  "dataVersion": "DataVersion-string",  
  "data": {  
    "operationContext": <OperationContextObject>,  
    "blobUri": "StorageURL-string",  
  },  
  "eventType": "request.blob.delete"  
}
```

Gridwich > Requester uses [ResponseBlobDeleteScheduledDTO](#).

JSON

```
{  
  "id": "GUID-string",  
  "topic": "Topic-string",  
  "subject": "Subject-string",  
  "dataVersion": "DataVersion-string",  
  "data": {  
    "operationContext": <OperationContextObject>,  
  }
```

```
        "blobUri": "StorageURL-string",
        "blobMetadata": <Metadata-Dictionary>
    },
    "eventType": "response.blob.delete.scheduled"
}
```

## Gridwich informs requester that it deleted a blob

The blob deletion can come from any source, like an explicit request from a requester or a result of internal operations.

Gridwich > Requester, uses [ResponseBlobDeleteSuccessDTO](#).

JSON

```
{
    "id": "GUID-string",
    "topic": "Topic-string",
    "subject": "Subject-string",
    "dataVersion": "DataVersion-string",
    "data": {
        "operationContext": <OperationContextObject>,
        "blobUri": "StorageURL-string"
    },
    "eventType": "response.blob.delete.success"
}
```

## Requester asks Gridwich to return a time-expiration content SAS URL

Requester > Gridwich uses [RequestBlobSasUrlCreateDTO](#).

JSON

```
{
    "id": "GUID-string",
    "topic": "Topic-string",
    "subject": "Subject-string",
    "dataVersion": "DataVersion-string",
    "data": {
        "operationContext": <OperationContextObject>,
        "blobUri": "StorageURL-string",
        "secToLive": 1200
    },
    "eventType": "request.blob.sas-url.create"
}
```

Gridwich > Requester uses ResponseBlobSasUrlSuccessDTO ↴ .

JSON

```
{  
    "id": "GUID-string",  
    "topic": "Topic-string",  
    "subject": "Subject-string",  
    "dataVersion": "DataVersion-string",  
    "data": {  
        "operationContext": <OperationContextObject>,  
        "sasUrl": "StorageURL-SAS-string"  
    },  
    "eventType": "response.blob.sas-url.success"  
}
```

## Requester asks Gridwich to encode via CloudPort Workflow

Requester > Gridwich, uses RequestCloudPortEncodeCreateDTO ↴ .

JSON

```
{  
    "id": "GUID-string",  
    "topic": "Topic-string",  
    "subject": "Subject-string",  
    "dataVersion": "DataVersion-string",  
    "data": {  
        "operationContext": <OperationContextObject>,  
        "inputs": [  
            {"blobUri": "StorageURL-string" }  
        ],  
        "outputContainer":  
        "https://<storageaccountname>.blob.core.windows.net/<containername>/",  
        "workflowName": "TestWorkflow2",  
        "parameters": [ { "prop1": "value1" } ],  
        "secToLive": 18000  
    },  
    "eventType": "request.encode.cloudport.create",  
}
```

## Requester asks Gridwich to encode via Flip

Requester > Gridwich, uses RequestFlipEncodeCreateDTO ↴ .

JSON

```
{
  "id": "GUID-string",
  "topic": "Topic-string",
  "subject": "Subject-string",
  "dataVersion": "DataVersion-string",
  "data": {
    "operationContext": <OperationContextObject>,
    "inputs": [
      {"blobUri": "StorageURL-string" }
    ],
    "outputContainer": "StorageURL-string", // of the Storage
    container
    "factoryName": "gws-dev-flip",
    "profiles": "h264",
    "parameters": [ { "prop1": "value1" } ],
    "secToLive": 18000
  },
  "eventType": "request.encode.flip.create"
}
```

## Example of RequestFlipEncodeCreateDTO

JSON

```
{
  "id": "GUID-string",
  "topic": "Topic-string",
  "subject": "Subject-string",
  "dataVersion": "DataVersion-string",
  "data": {
    "operationContext": { "progId": 1234 },
    "inputs": [
      {
        "blobUri": "https://gws-
        sa1.blob.core.windows.net/Vid0001Container/input.mp4"
      }
    ],
    "outputContainer": "https://gws-
    sa22out.blob.core.windows.net/Out0004/",
    "factoryName": "gws-dev-flip",
    "profiles": "h264",
    "parameters": [ { "someProperty1": "someValue1" } ],
    "secToLive": 18000
  },
  "eventType": "request.encode.flip.create"
}
```

# Requester asks Gridwich to encode a Media Services v3 transform

Requester > Gridwich, uses [RequestMediaServicesV3EncodeCreateDTO](#)

```
JSON

{
  "id": "GUID-string",
  "topic": "Topic-string",
  "subject": "Subject-string",
  "dataVersion": "DataVersion-string",
  "data": {
    "operationContext": <OperationContextObject>,
    "inputs": [
      { "blobUri": "https://<storageaccountname>.blob.core.windows.net/<containername>/<filename>" }
    ],
    "outputContainer": "https://<storageaccountname>.blob.core.windows.net/<containername>/",
    "transformName": "audio-mono-aac-video-mbr-no-bframes",
    "timeBasedEncode": {
      "startSeconds": 01.234,
      "endSeconds": 12.345
    }
  },
  "eventType": "request.encode.mediaservicesv3.create"
}
```

The `transformName` property is one of the [CustomTransforms](#):

- `audio-mono-aac-video-mbr-no-bframes`
- `audio-copy-video-mbr-no-bframes`
- `audio-copy-video-mbr`

The start and end times are always relative to the start of the media file, regardless of the presentation start time.

## Gridwich encoders common request successful dispatch response

Gridwich > Requester, uses [ResponseEncodeDispatchedDTO](#).

```
JSON
```

```
{
  "id": "GUID-string",
  "topic": "Topic-string",
  "subject": "Subject-string",
  "dataVersion": "DataVersion-string",
  "data": {
    "operationContext": <OperationContextObject>,
    "encoderContext": <EncoderContext>,
    "workflowJobName": "CloudPort or Flip assigned job name for workflow
instance, or Media Services Job Id."
  },
  "eventType": "response.encode.<encodername>.dispatched"
}
```

The `<encodername>` is one of `cloudport`, `flip`, or `mediaservicesv3`.

## Gridwich encoder asynchronous status messages

The Gridwich encoders generate four kinds of events during or at the end of encoding:

- Scheduled
- Processing
- Success
- Canceled

An encode request failure generates a Gridwich Failure event.

### Encoding status scheduled

Gridwich > Requester, uses [ResponseEncodeScheduledDTO](#).

JSON

```
{
  "id": "GUID-string",
  "topic": "Topic-string",
  "subject": "Subject-string",
  "dataVersion": "DataVersion-string",
  "data": {
    "operationContext": <OperationContextObject>,
    "encoderContext": <EncoderContext>,
    "workflowJobName": "CloudPort or Flip assigned job name for workflow
instance, or Media Services Job Id."
  },
  "eventType": "response.encode.<encodername>.scheduled"
}
```

## Encoding status processing

Gridwich > Requester, uses [ResponseEncodeProcessingDTO](#).

JSON

```
{  
    "id": "GUID-string",  
    "topic": "Topic-string",  
    "subject": "Subject-string",  
    "dataVersion": "DataVersion-string",  
    "data": {  
        "operationContext": <OperationContextObject>,  
        "encoderContext": <EncoderContext>,  
        "workflowJobName": "CloudPort or Flip assigned job name for workflow  
instance, or Media Services Job Id.",  
        "currentStatus": "string",  
        "percentComplete": 50,  
    },  
    "eventType": "response.encode.<encodername>.processing"  
}
```

## Encoding status success

Gridwich > Requester, uses [ResponseEncodeSuccessDTO](#).

JSON

```
{  
    "id": "GUID-string",  
    "topic": "Topic-string",  
    "subject": "Subject-string",  
    "dataVersion": "DataVersion-string",  
    "data": {  
        "operationContext": <OperationContextObject>, ,  
        "encoderContext": <EncoderContext>, ,  
        "workflowJobName": "CloudPort or Flip assigned job name for workflow  
instance, or Media Services Job Id.",  
        "outputs": [  
            { "blobUri": "StorageURL-string" }  
        ]  
    },  
    "eventType": "response.encode.<encodername>.success"  
}
```

## Encoding status canceled

Gridwich > Requester, uses [ResponseEncodeCanceledDTO](#).

JSON

```
{  
  "id": "GUID-string",  
  "topic": "Topic-string",  
  "subject": "Subject-string",  
  "dataVersion": "DataVersion-string",  
  "data": {  
    "operationContext": <OperationContextObject>,  
    "encoderContext": <EncoderContext>,  
    "workflowJobName": "CloudPort or Flip assigned job name for workflow  
instance, or Media Services Job Id."  
  },  
  "eventType": "response.encode.<encodername>.canceled"  
}
```

## Requester asks Gridwich to change a blob's storage tier

Requester > Gridwich uses [RequestBlobTierChangeDTO](#).

JSON

```
{  
  "id": "GUID-string",  
  "topic": "Topic-string",  
  "subject": "Subject-string",  
  "dataVersion": "DataVersion-string",  
  "data": {  
    "operationContext": <OperationContextObject>,  
    "blobUri": "StorageURL-string",  
    "accessTier": "string",  
    "rehydratePriority": "string"  
  },  
  "eventType": "request.blob.tier.change"  
}
```

- The `accessTier` property is `Hot`, `Cool`, or `Archive`.
- The `rehydratePriority` property is `Standard` or `High`.

Gridwich > Requester uses [ResponseBlobTierChangeSuccessDTO](#)

JSON

```
{  
  "id": "GUID-string",  
  "topic": "Topic-string",  
  "subject": "Subject-string",  
  "dataVersion": "DataVersion-string",  
  "data": {
```

```

    "operationContext": <OperationContextObject>,
    "blobUri": "StorageURL-string",
    "accessTier": "string",
    "rehydratePriority": "string"
  },
  "eventType": "response.blob.tier.success"
}

```

## Requester asks Gridwich to create a blob container

The request provides the Storage Account and container name.

Requester > Gridwich uses [RequestContainerCreateDTO](#).

JSON

```
{
  "id": "GUID-string",
  "topic": "Topic-string",
  "subject": "Subject-string",
  "dataVersion": "DataVersion-string",
  "data": {
    "operationContext": <OperationContextObject>,
    "storageAccountName": "string", // e.g. mySA1
    "containerName": "string"      // e.g. mycontainer
  },
  "eventType": "request.blob.container.create"
}
```

Gridwich > Requester uses [ResponseContainerCreatedSuccessDTO](#).

JSON

```
{
  "id": "GUID-string",
  "topic": "Topic-string",
  "subject": "Subject-string",
  "dataVersion": "DataVersion-string",
  "data": {
    "operationContext": <OperationContextObject>,
    "storageAccountName": "string",
    "containerName": "string"
  },
  "eventType": "response.blob.container.create.success"
}
```

## Requester asks Gridwich to delete a blob container

The request provides the Storage Account and container name.

**Requester** > **Gridwich** uses `RequestContainerDeleteDTO` .

JSON

```
{  
  "id": "GUID-string",  
  "topic": "Topic-string",  
  "subject": "Subject-string",  
  "dataVersion": "DataVersion-string",  
  "data": {  
    "operationContext": <OperationContextObject>,  
    "storageAccountName": "string",  
    "containerName": "string"  
  },  
  "eventType": "request.blob.container.delete"  
}
```

**Gridwich** > **Requester** uses `ResponseContainerDeleteSuccessDTO` .

JSON

```
{  
  "id": "GUID-string",  
  "topic": "Topic-string",  
  "subject": "Subject-string",  
  "dataVersion": "DataVersion-string",  
  "data": {  
    "operationContext": <OperationContextObject>,  
    "storageAccountName": "string",  
    "containerName": "string"  
  },  
  "eventType": "response.blob.container.delete.success"  
}
```

## Requester asks Gridwich to change the public access of a container

The request provides the container name and an `accessType` of `Blob`, `BlobContainer`, or `None`.

**Requester** > **Gridwich** uses `RequestContainerAccessChangeDTO` .

JSON

```
{  
  "id": "GUID-string",  
  "accessType": "string",  
  "containerName": "string",  
  "blobName": "string",  
  "publicAccess": "string",  
  "blobType": "string",  
  "blobProperties": {  
    "name": "string",  
    "value": "string"  
  }  
}
```

```

"topic": "Topic-string",
"subject": "Subject-string",
"dataVersion": "DataVersion-string",
"data": {
    "operationContext": <OperationContextObject>,
    "storageAccountName": "string",
    "containerName": "string",
    "accessType": "string"
},
"eventType": "request.blob.container.access.change"
}

```

Gridwich > Requester uses ResponseContainerAccessChangeSuccessDTO [↗](#).

JSON

```
{
    "id": "GUID-string",
    "topic": "Topic-string",
    "subject": "Subject-string",
    "dataVersion": "DataVersion-string",
    "data": {
        "operationContext": <OperationContextObject>,
        "storageAccountName": "string",
        "containerName": "string",
        "accessType": "string"
    },
    "eventType": "response.blob.container.access.change.success"
}
```

## Requester asks Gridwich to publish content via Azure Media Services

The request is to create or delete a content asset locator.

### Create content asset locator

Requester > Gridwich uses RequestMediaServicesLocatorCreateDTO [↗](#).

JSON

```
{
    "id": "GUID-string",
    "topic": "Topic-string",
    "subject": "Subject-string",
    "dataVersion": "DataVersion-string",
    "data": {
        "operationContext": <OperationContextObject>,

```

```

    "containerUri": "",

    "streamingPolicyName": "clearStreamingOnly",
    "contentKeyPolicyName": null,
    or
    "streamingPolicyName": "cencDrmStreaming",
    "contentKeyPolicyName": "cencDrmKey",
    or
    "streamingPolicyName": "multiDrmStreaming",
    "contentKeyPolicyName": "multiDrmKey",

    "timeBasedfilter":{
        "startSeconds": 01.234,
        "endSeconds": 12.345,
    },
    "generateAudioFilters": true
},
"eventType": "request.mediaservices.locator.create"
}

```

The start and end times are always relative to the start of the media file, regardless of the presentation start time.

[ ] [Expand table](#)

Streaming policy	DRM technologies
cencDrmStreaming	Microsoft PlayReady + Google Widevine
multiDrmStreaming	Microsoft PlayReady + Google Widevine + Apple FairPlay

Gridwich > Requester uses [ResponseMediaServicesLocatorCreateSuccessDTO](#).

JSON
<pre>{     "id": "GUID-string",     "topic": "Topic-string",     "subject": "Subject-string",     "dataVersion": "DataVersion-string",     "data": {         "operationContext": &lt;OperationContextObject&gt;,         "locatorName": "someNameSetByGridwich",         "cencKeyId": "someKeyId for PlayReady and Widevine encryption",         "cbcKeyId": "someKeyId for FairPlay encryption",         "dashUri": "someUri which ends with manifest(format=mpd-time . . . )",         "hlsUri": "someUri which ends with manifest(format=m3u8-aapl . . .)"     } }</pre>

```
},
  "eventType": "response.mediaservices.locator.create.success"
}
```

## Delete asset locator

Requester > Gridwich uses RequestMediaServicesLocatorDeleteDTO [↗](#).

JSON

```
{
  "id": "GUID-string",
  "topic": "Topic-string",
  "subject": "Subject-string",
  "dataVersion": "DataVersion-string",
  "data": {
    "operationContext": <OperationContextObject>,
    "locatorName": "someName"
  },
  "eventType": "request.mediaservices.locator.delete"
}
```

The `locatorName` property is an opaque string generated by Gridwich.

Gridwich > Requester uses ResponseMediaServicesLocatorDeleteSuccessDTO [↗](#).

JSON

```
{
  "id": "GUID-string",
  "topic": "Topic-string",
  "subject": "Subject-string",
  "dataVersion": "DataVersion-string",
  "data": {
    "operationContext": <OperationContextObject>,
    "locatorName": "someName"
  },
  "eventType": "response.mediaservices.locator.delete.success"
}
```

## Requester asks Gridwich to rotate to a new storage key

The `Rollkey` event family differs from others in Gridwich in that, while the request accepts an `operationContext` value, none of the response events include it.

Failure events aren't of the normal `response.failure` event type, but instead have a type value of `response.rollkey.storage.failure`.

The `response.rollkey.storage.failure` failure events:

- Don't include any of the normal failure event logging information `log` data properties.
- Contain an additional data property named `error` that contains error message text. Other Gridwich failures carry that text on the `logEventMessage` data property.

These points reflect the current state of the Azure Logic App that performs the RollKey operation. The definition of the Logic App is in the [infrastructure/terraform/keyroller/main.tf](#) Terraform file.

The `keyName` corresponds to the key name that Azure Storage defines in its [Get Keys](#) operation.

## Requester > Gridwich

```
JSON

{
  "id": "GUID-string",
  "topic": "Topic-string",
  "subject": "Subject-string",
  "dataVersion": "DataVersion-string",
  "data": {
    "operationContext": <OperationContextObject>,
    "account": "storageAccountName",
    "keyName": "key1"
  },
  "eventType": "request.rollkey.storage"
}
```

## Gridwich > Requester

- Success:

```
JSON

{
  "id": "GUID-string",
  "topic": "Topic-string",
  "subject": "Subject-string",
  "dataVersion": "DataVersion-string",
  "data": {
    "account": "storageAccountName",
    "keyName": "key1"
  }
}
```

```
        },
        "eventType": "response.rollkey.storage.success"
    }
```

- Failure:

JSON

```
{
    "id": "GUID-string",
    "topic": "Topic-string",
    "subject": "Subject-string",
    "dataVersion": "1.0",
    "data": {
        "account": "storageAccountName1",
        "keyName": "key1",
        "error": "error message text"
    },
    "eventType": "response.rollkey.storage.failure"
}
```

Failure results for this operation aren't as complete as [normal Gridwich failures](#).

## Next steps

Product documentation:

- [Gridwich cloud media system](#)
- [Azure Media Services v3 overview](#)
- [What is Azure Blob storage?](#)
- [What is Azure Logic Apps?](#)

Microsoft Learn modules:

- [Explore Azure Storage services](#)
- [Introduction to Azure Logic Apps](#)

## Related resources

- [Gridwich content protection and DRM](#)
- [Gridwich operations for Azure Storage](#)
- [Logging in Gridwich](#)

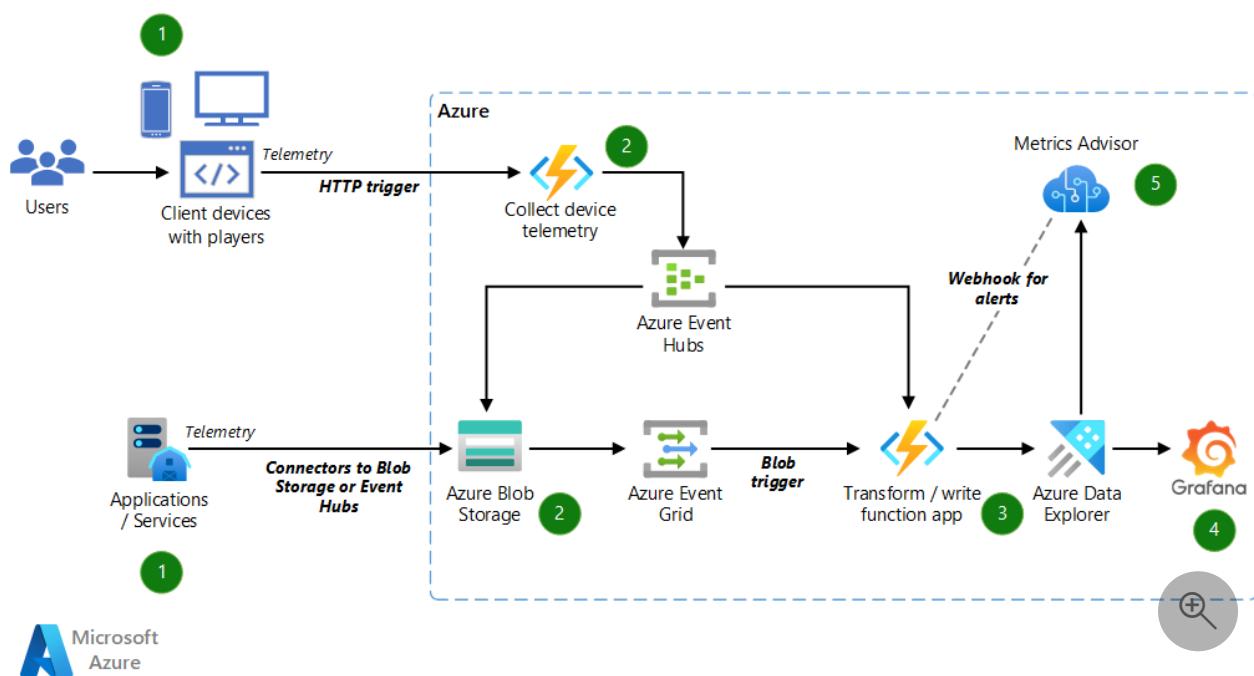
# Build real-time monitoring and observable systems for media

Azure Data Explorer   Azure Functions   Azure AI Metrics Advisor   Azure Blob Storage   Azure Event Hubs

This architecture describes a solution that provides real-time monitoring and observability of systems and end-user device telemetry data. It focuses on a use case for the media industry.

*Grafana* [↗](#) is a trademark of its respective company. No endorsement is implied by the use of this mark.

## Architecture



Download a [Visio file](#) [↗](#) of this architecture.

## Dataflow

In the observable system shown in the diagram, raw telemetry is streamed to Azure Blob Storage via HTTP and connectors. The raw telemetry is processed, transformed, normalized, and saved in Azure Data Explorer for analysis. Systems like Grafana and Azure Metrics Advisor read data from Data Explorer and provide insights to end users.

More specifically, these are the elements of the system in the diagram:

- 1. Instrumentation.** Instrumentation occurs via probes or agents that are installed in systems to monitor data. These agents come in various forms. For example, in a video-on-demand streaming platform, a company might use open-standards [dash.js](#) to collect Quality of Experience metrics from customers.
- 2. Ingestion.** This raw telemetry can come directly from end clients via HTTP calls. Alternatively, you can upload it via third-party systems to persistent storage and data lakes like Blob Storage. Blob Storage supports the ability to invoke an Azure function when a new file is uploaded. You can use this trigger mechanism to move raw telemetry to structured data warehouses.
- 3. Transformation and persistence.** You might need a transformation system to normalize your data. An Azure Functions app transforms the data as needed and then writes it to Data Explorer. Data Explorer is ideal for big data analytics because it's designed for high performance and throughput on large data sets.
- 4. Monitoring.** Azure Managed Grafana supports integration with Data Explorer. You can use the drag-and-drop features of Grafana to quickly build dashboards and charts. Grafana is a good fit for media monitoring because it provides sub-minute refreshing of dashboard tiles and can also be used for alerting.
- 5. Anomaly detection.** The Grafana dashboard provides support for manual monitoring in the NOC. However, with a large data set and a user base spread across geographies and using various devices, manual identification of issues via charts and alert rules that have hard-coded thresholds becomes inefficient. You can use AI to address this problem. Services like Metrics Advisor use machine learning algorithms to automatically understand and detect anomalies based on time-series data. In addition, the Kusto data platform has built-in anomaly detection functions that account for seasonality and baseline trends in the data.

## Components

- [Data Explorer](#) is a managed data analytics service for real-time analysis of large volumes of data. Data Explorer is a great tool for handling large datasets that require high speed and throughput of data retrieval. This architecture uses Data Explorer to store and query datasets for analysis.
- [Blob Storage](#) is used to hold raw telemetry. This telemetry can come from your applications and services or from third-party vendors. The data can be treated as transient if you don't need to perform more analysis later. The data from Blob Storage is ingested into Data Explorer clusters.
- [Azure Event Grid](#) is an event delivery system. It's used to listen to events that are published by Blob Storage. Azure Storage events allow applications to react to events like the creation and deletion of blobs. An Azure function subscribes to events that are published by Event Grid.

- [Azure Event Hubs](#) is a real-time data ingestion service that you can use to ingest millions of events per second from any source. Event hubs represent the front door, often called an event *ingestor*, for an event pipeline. An event ingestor is a component or service that's located between event publishers and event consumers. It decouples the production of an event stream from the consumption of the events.
- [Azure Functions](#) is a serverless solution that's used to parse and transform data ingested via HTTP and blob endpoints and write to the Data Explorer cluster.
- [Azure Managed Grafana](#) connects easily to Data Explorer. In this architecture, it generates charts and dashboards that visualize telemetry data. Azure Managed Grafana provides deep integration with Microsoft Entra ID so that you can implement role-based access to dashboards and views.
- [Metrics Advisor](#) is a part of Azure Applied AI Services. It uses AI to perform data monitoring and anomaly detection in time-series data. Metrics Advisor automates the process of applying models to data and provides a set of APIs and a web-based workspace for data ingestion, anomaly detection, and diagnostics. You can use it even if you have no knowledge of machine learning.

## Alternatives

[Azure Data Factory](#) and [Azure Synapse Analytics](#) provide tools and workspaces for building ETL workflows and the ability to track and retry jobs from a graphical interface. Note that Data Factory and Azure Synapse both have a minimum lag of about 5 minutes from the time of ingestion to persistence. This lag might be acceptable in your monitoring system. If it is, we recommend that you consider these alternatives.

## Scenario details

Organizations often deploy varied and large-scale technologies to solve business problems. These systems, and end-user devices, generate large sets of telemetry data.

This architecture is based on a use case for the media industry. Media streaming for live and video-on-demand playback requires near real-time identification of and response to application problems. To support this real-time scenario, organizations need to collect a massive telemetry set, which requires scalable architecture. After the data is collected, other types of analysis, like AI and anomaly detection, are needed to efficiently identify problems across so large a data set.

When large-scale technologies are deployed, the system and end-user devices that interact with them generate massive sets of telemetry data. In traditional scenarios, this data is analyzed via a data warehouse system to generate insights that can be used to

support management decisions. This approach might work in some scenarios, but it's not responsive enough for streaming media use cases. To solve this problem, real-time insights are required for the telemetry data that's generated from monitoring servers, networks, and the end-user devices that interact with them. Monitoring systems that catch failures and errors are common, but to catch them in near real-time is difficult. That's the focus of this architecture.

In a live streaming or video-on-demand setting, telemetry data is generated from systems and heterogeneous clients (mobile, desktop, and TV). The solution involves taking raw data and associating context with the data points, for example, dimensions like geography, end-user operating system, content ID, and CDN provider. The raw telemetry is collected, transformed, and saved in Data Explorer for analysis. You can then use AI to make sense of the data and automate the manual processes of observation and alerting. You can use systems like Grafana and Metrics Advisor to read data from Data Explorer to show interactive dashboards and trigger alerts.

## Considerations

These considerations implement the pillars of the Azure Well-Architected Framework, a set of guiding tenets that you can use to improve the quality of a workload. For more information, see [Microsoft Azure Well-Architected Framework](#).

## Reliability

Reliability ensures your application can meet the commitments you make to your customers. For more information, see [Overview of the reliability pillar](#).

Business-critical applications need to keep running even during disruptive events like Azure region or CDN outages. There are two primary strategies and one hybrid strategy for building redundancy into your system:

- **Active/active.** Duplicate code and functions are running. Either system can take over during a failure.
- **Active/standby.** Only one node is active/primary. The other one is ready to take over in case the primary node goes down.
- **Mixed.** Some components/services are in the active/active configuration, and some are in active/standby.

Keep in mind that not all Azure services have built-in redundancy. For example, Azure Functions runs a function app only in a specific region. [Azure Functions geo-disaster recovery](#) describes various strategies that you can implement, depending on how your functions are triggered (HTTP versus pub/sub).

The ingestion and transformation function app can run in active/active mode. You can run Data Explorer in both [active/active and active/standby configurations](#).

Azure Managed Grafana supports [availability zone redundancy](#). One strategy for creating cross-region redundancy is to set up Grafana in each region in which your Data Explorer cluster is deployed.

## Cost optimization

Cost optimization is about reducing unnecessary expenses and improving operational efficiencies. For more information, see [Overview of the cost optimization pillar](#).

The cost of this architecture depends on the number of ingress telemetry events, your storage of raw telemetry in Blob Storage and Data Explorer, an hourly cost for Azure Managed Grafana, and a static cost for the number of time-series charts in Metrics Advisor.

You can use the [Azure pricing calculator](#) to estimate your hourly or monthly costs.

## Performance efficiency

Performance efficiency is the ability of your workload to scale to meet the demands placed on it by users in an efficient manner. For more information, see [Performance efficiency pillar overview](#).

Depending on the scale and frequency of incoming requests, the function app might be a bottleneck, for two main reasons:

- **Cold start.** Cold start is a consequence of serverless executions. It refers to the scheduling and setup time that's required to spin up an environment before the function first starts running. At most, the required time is a few seconds.
- **Frequency of requests.** Say you have 1,000 HTTP requests but only a single-threaded server to handle them. You won't be able to service all 1,000 HTTP requests concurrently. To serve these requests in a timely manner, you need to deploy more servers. That is, you need to scale horizontally.

We recommend that you use Premium or Dedicated SKUs to:

- Eliminate cold start.
- Handle requirements for concurrent requests by scaling the number of servicing virtual machines up or down.

For more information, see [Select a SKU for your Azure Data Explorer cluster](#).

# Deploy this scenario

For information about deploying this scenario, see [real-time-monitoring-and-observability-for-media](#) on GitHub. This code sample includes the necessary infrastructure-as-code (IaC) to bootstrap development and Azure functions to ingest and transform the data from HTTP and blob endpoints.

## Contributors

*This article is maintained by Microsoft. It was originally written by the following contributors.*

Principal authors:

- [John Hauppa](#) | Senior Technical Program Manager
- [Uffaz Nathaniel](#) | Principal Software Engineer

Other contributors:

- [Mick Alberts](#) | Technical Writer
- [Dilmurod Makhmadaliev](#) | Software Engineer
- [Omeed Musavi](#) | Principal Software Engineer Lead
- [Ayo Mustapha](#) | Principal Technical Program Manager

*To see non-public LinkedIn profiles, sign in to LinkedIn.*

## Next steps

- [Supplementary code samples](#)
- [Azure Data Explorer documentation](#)
- [Introduction to Azure Data Explorer - Training](#)
- [Introduction to Azure Functions](#)

## Related resources

- [Monitor Media Services](#)
- [Analytics architecture design](#)
- [Big data analytics with Azure Data Explorer](#)

# HPC media rendering

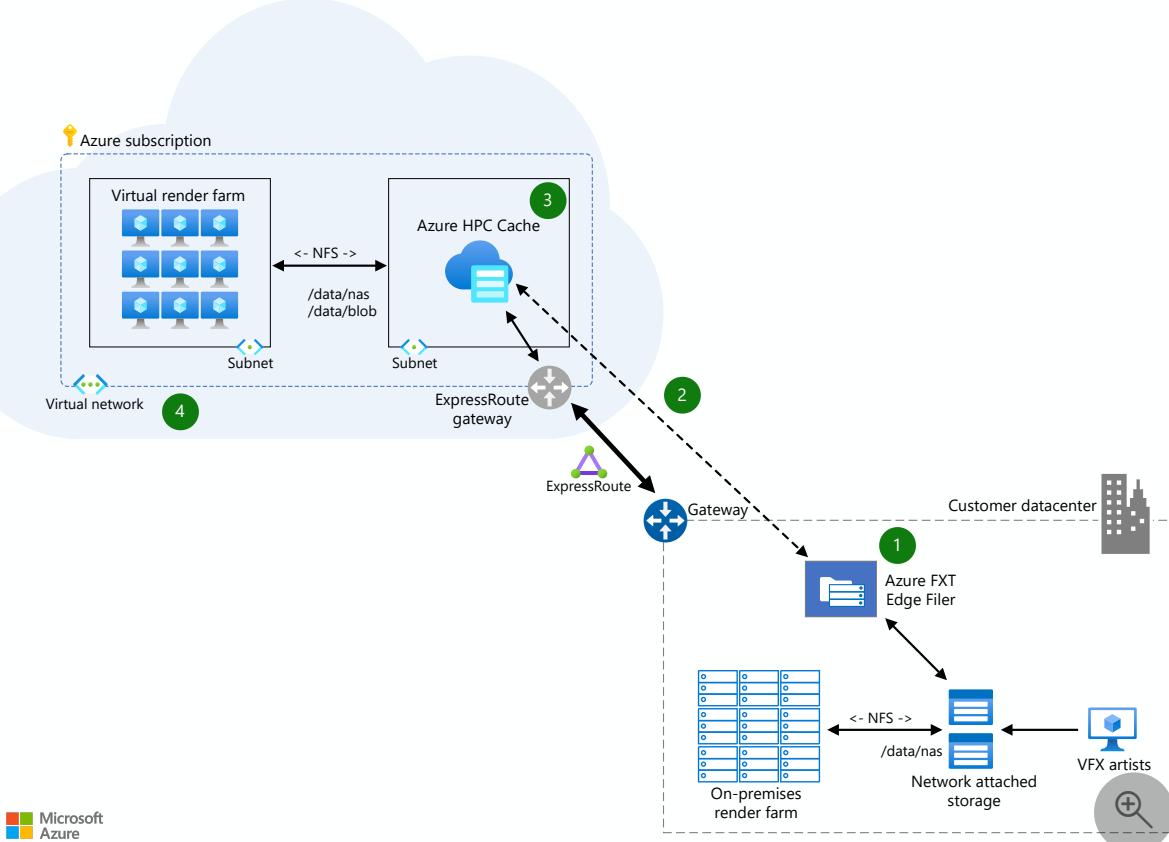
Azure Batch   Azure CycleCloud   Azure FXT Edge Filer   Azure Virtual Machine Scale Sets

## 💡 Solution ideas

This article is a solution idea. If you'd like us to expand the content with more information, such as potential use cases, alternative services, implementation considerations, or pricing guidance, let us know by providing [GitHub feedback](#).

This solution allows studios to leverage on-premises capacity to its fullest with the Azure FXT Edge Filer for NAS acceleration. When demand grows beyond on-premises capacity, burst render provides access to tens of thousands of cores using Azure Virtual Machine Scale Sets. An Express Route connection and HPC Cache minimize latency while studios securely manage storage in a single place without replication.

## Architecture



Download a [Visio file](#) of this architecture.

# Dataflow

1. Optimize access to NAS files and support remote artists with the Azure FXT Edge Filer connecting artists to low-latency storage.
2. Connecting on-premises storage resources to Azure via Azure Express Route providing a secure, private link to additional render cores.
3. Azure HPC Cache provides low-latency access to tens of thousands of compute cores with burst rendering. Azure SDK support in HPC Cache enables automation for easy infrastructure management and cost efficiencies.
4. A virtual render farm is available in Azure using Virtual Machine Scale Sets that grow as you need it and provides capacity to meet fluctuating demand.

## Components

- [N-Series VMs](#) : N-series virtual machines are ideal for compute and graphics-intensive workloads, helping customers to fuel innovation through scenarios like high-end remote visualization, deep learning, and predictive analytics.
- [H-Series VMs](#) : The H-series is a new family specifically designed to handle high performance computing workloads such as financial risk modeling, seismic and reservoir simulation, molecular modeling, and genomic research.
- Effectively manage common workloads with ease while creating and optimizing HPC clusters with Microsoft [Azure CycleCloud](#) .
- [Avere vFXT](#) : Faster, more accessible data storage for high-performance computing at the edge
- [Azure Batch](#) : Cloud-scale job scheduling and compute management

## Scenario details

### Potential use cases

Graphics designers, artists, and animation designers need high performance systems to make sure they deliver the best quality work and can accommodate change requests without waiting hours for the processing to finish. Areas that studios can see the benefits from high performance computing include:

- Animation and modeling.
- 3D Rendering.
- Compositing and color grading.

# Next steps

- [N-Series Virtual Machines Documentation](#)
- [H-Series Virtual Machines Documentation](#)
- [Azure CycleCloud Documentation](#)
- [Avere vFXT Documentation](#)
- [Azure Batch Documentation](#)

# Serverless functions architecture design

Article • 07/28/2022

Serverless architecture evolves cloud platforms toward pure cloud-native code by abstracting code from the infrastructure that it needs to run. [Azure Functions](#) is a serverless compute option that supports *functions*, small pieces of code that do single things.

Benefits of using serverless architectures with Functions applications include:

- The Azure infrastructure automatically provides all the updated servers that applications need to keep running at scale.
- Compute resources allocate dynamically, and instantly autoscale to meet elastic demands. Serverless doesn't mean "no server," but "less server," because servers run only as needed.
- Micro-billing saves costs by charging only for the compute resources and duration the code uses to execute.
- Function *bindings* streamline integration by providing declarative access to a wide variety of Azure and third-party services.

Functions are *event-driven*. An external event like an HTTP web request, message, schedule, or change in data *triggers* the function code. A Functions application doesn't code the trigger, only the response to the trigger. With a lower barrier to entry, developers can focus on business logic, rather than writing code to handle infrastructure concerns like messaging.

Azure Functions is a managed service in Azure and Azure Stack. The open source Functions runtime works in many environments, including Kubernetes, Azure IoT Edge, on-premises, and other clouds.

Serverless and Functions require new ways of thinking and new approaches to building applications. They aren't the right solutions for every problem. For example serverless Functions scenarios, see [Reference architectures](#).

## Implementation steps

Successful implementation of serverless technologies with Azure Functions requires the following actions:

- Decide and plan

*Architects and technical decision makers (TDMs)* perform [application assessment](#), conduct or attend [technical workshops and trainings](#), run [proof of concept \(PoC\)](#) or [pilot](#) projects, and conduct architectural designs sessions as necessary.

- [Develop and deploy apps](#)

*Developers* implement serverless Functions app development patterns and practices, configure DevOps pipelines, and employ site reliability engineering (SRE) best practices.

- [Manage operations](#)

*IT professionals* identify hosting configurations, future-proof scalability by automating infrastructure provisioning, and maintain availability by planning for business continuity and disaster recovery.

- [Secure apps](#)

*Security professionals* handle Azure Functions security essentials, secure the hosting setup, and provide application security guidance.

## Related resources

- To learn more about serverless technology, see the [Azure serverless documentation](#).
- To learn more about Azure Functions, see the [Azure Functions documentation](#).
- For help with choosing a compute technology, see [Choose an Azure compute service for your application](#).

# Keyword search and speech-to-text

Azure Content Delivery Network   Azure AI Search   Azure Media Player   Azure Video Indexer

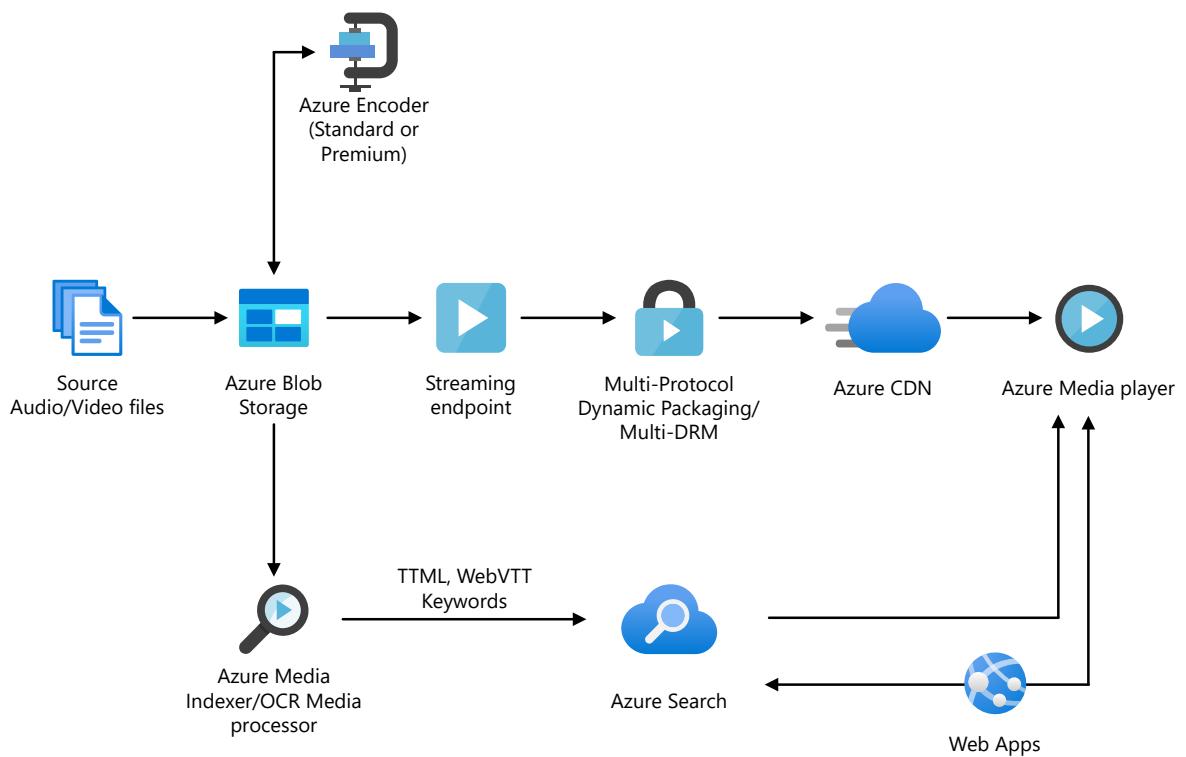
Azure App Service

## 💡 Solution ideas

This article is a solution idea. If you'd like us to expand the content with more information, such as potential use cases, alternative services, implementation considerations, or pricing guidance, let us know by providing [GitHub feedback](#).

This solution idea identifies speech in static video files to manage speech as standard content.

## Architecture



 Microsoft Azure



Download a [Visio file](#) of this architecture.

# Dataflow

- Azure Blob Storage stores large amounts of unstructured data that can be accessed from anywhere in the world via HTTP or HTTPS. You can use [Blob Storage](#) to expose data publicly to the world, or to store application data privately.
- [Azure Encoding](#) converts media files from one encoding to another.
- [Azure streaming endpoint](#) represents a streaming service that can deliver content directly to a client player application, or to a content delivery network (CDN) for further distribution.
- [Content Delivery Network](#) provides secure, reliable content delivery with broad global reach and a rich feature set.
- [Azure Media Player](#) uses industry standards, such as HTML5 (MSE/EME) to provide an enriched adaptive streaming experience. Regardless of the playback technology used, you have a unified JavaScript interface to access APIs.
- [Azure Cognitive Search](#) provides a ready-to-use service that gets populated with data and then used to add search functionality to a web or mobile application.
- [Web Apps](#) hosts the website or web application.
- [Azure Media Indexer](#) makes the content of your media files searchable and generates a full-text transcript for closed-captioning and keywords. Media files are processed individually or in batches.

# Components

- [Blob Storage](#) is a service that's part of [Azure Storage](#). Blob Storage offers optimized cloud object storage for large amounts of unstructured data.
- [Azure Media Services](#) is a cloud-based platform that you can use to stream video, enhance accessibility and distribution, and analyze video content.
- [Live and on-demand streaming](#) is a feature of Azure Media Services that delivers content to various devices at scale.
- [Azure Encoding](#) provides a way to convert files that contain digital video or audio from one standard format to another.
- [Azure Media Player](#) plays videos that are in various formats.
- [Azure Content Delivery Network](#) offers a global solution for rapidly delivering content. This service provides your users with fast, reliable, and secure access to your apps' static and dynamic web content.
- [Azure Cognitive Search](#) is a cloud search service that supplies infrastructure, APIs, and tools for searching. You can use Azure Cognitive Search to build search experiences over private, heterogeneous content in web, mobile, and enterprise applications.

- [App Service](#) provides a framework for building, deploying, and scaling web apps. The [Web Apps](#) feature is a service for hosting web applications, REST APIs, and mobile back ends.
- [Azure Media Indexer](#) provides a way to make content of your media files searchable. It can also generate a full-text transcript for closed captioning and keywords.

## Scenario details

A speech-to-text solution provides a way to identify speech in static video files so you can manage it as standard content. For instance, employees can use this technology to search within training videos for spoken words or phrases. Then they can navigate to the specific moment in the video that contains the word or phrase.

When you use this solution, you can upload static videos to an Azure website. The Azure Media Indexer uses the Speech API to index the speech within the videos and stores it in an Azure database. You can search for words or phrases by using the Web Apps feature of Azure App Service. Then you can retrieve a list of results. When you select a result, you can see the place in the video that mentions the word or phrase.

This solution is built on the Azure managed services [Content Delivery Network](#) and [Azure Cognitive Search](#).

## Potential use cases

This solution applies to scenarios that can benefit from the ability to search recorded speech. Examples include:

- Training and educational videos.
- Crime investigations.
- Customer service analysis.

## Next steps

- [How to use Azure Blob Storage](#)
- [How to encode an asset using Media Encoder](#)
- [How to manage streaming endpoints](#)
- [Using Azure Content Delivery Network](#)
- [Develop video player applications](#)
- [Create an Azure Cognitive Search service](#)
- [Run Web Apps in the cloud](#)

- Indexing media files

## Related resources

- Gridwich cloud media system
- Live stream digital media
- Video-on-demand digital media

# Live stream digital media

Azure Blob Storage

Azure Content Delivery Network

Azure Media Player

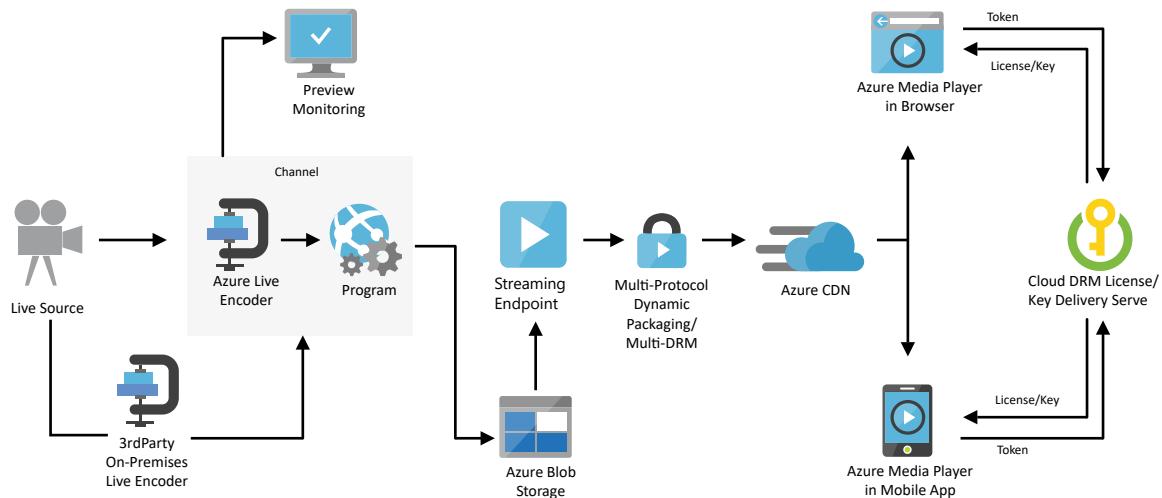
Azure Media Services

## 💡 Solution ideas

This article is a solution idea. If you'd like us to expand the content with more information, such as potential use cases, alternative services, implementation considerations, or pricing guidance, let us know by providing [GitHub feedback](#).

This solution is built on the Azure managed service: [Media Services](#) and [Content Delivery Network](#). These services run in a high-availability environment, patched and supported, allowing you to focus on your solution instead of the environment they run in.

## Architecture



Download a [Visio file](#) of this architecture.

## Components

- **Partner on-premises live encoder:** Outputs the live source for ingest into the cloud as RTMP(S), or Smooth Streaming.

- Stores large amounts of unstructured data that can be accessed from anywhere in the world via HTTP or HTTPS. You can use [Blob storage](#) to expose data publicly to the world, or to store application data privately.
- [Media Services](#): Ingest, encode, preview, store, and deliver your live streaming content. Live Events, Live Outputs, and Streaming Endpoints handle the live streaming functions, including ingestion, formatting, DVR, security, scalability, and redundancy.
- [Media Services Streaming Endpoint](#): Represents a streaming service that can deliver content directly to a client player application, or to a content delivery network (CDN) for further distribution.
- [Content Delivery Network](#): Provides secure, reliable content delivery with broad global reach and a rich feature set.
- [Azure Media Player](#): Uses industry standards such as HTML5 (MSE/EME) to provide an enriched adaptive streaming experience. Regardless of the playback technology used, developers have a unified JavaScript interface to access APIs.
- [Preview monitoring](#): Preview and validate a live stream before further processing and delivery.
- [Multi-DRM content protection](#): Delivers content securely using multi-DRM (PlayReady, Widevine, FairPlay Streaming) or AES clear key encryption.

## Solution details

A live streaming solution allows you to capture video in real-time and broadcast it to consumers in real-time. This can include streaming interviews, conferences, and sporting events online. In this solution, video is captured by a video camera and sent to a Live Event input endpoint. The Live Event receives the input stream and makes it available for streaming through a Streaming Endpoint to a web browser or mobile app. The Live Event also provides a preview monitoring endpoint to preview and validate your stream before further processing and delivery. The Live Event can also record and store the ingested content in order to be streamed later (video-on-demand).

## Potential use cases

This solution applies to anyone from large corporations to small content creators.

## Next steps

- [Overview of Media Services live transcoding](#)
- [How to use Azure Blob storage](#)
- [Overview of Media Services live streaming](#)

- Overview of Content Protection
- Using Azure Content Delivery Network
- Azure Media Services documentation
- Media services content protection ↗

## Related resources

- Performance tuning - Event streaming
- HPC media rendering
- Test Media Services V3 encoding
- Content Delivery Network analytics

# Video-on-demand digital media

Azure Blob Storage

Azure Content Delivery Network

Azure

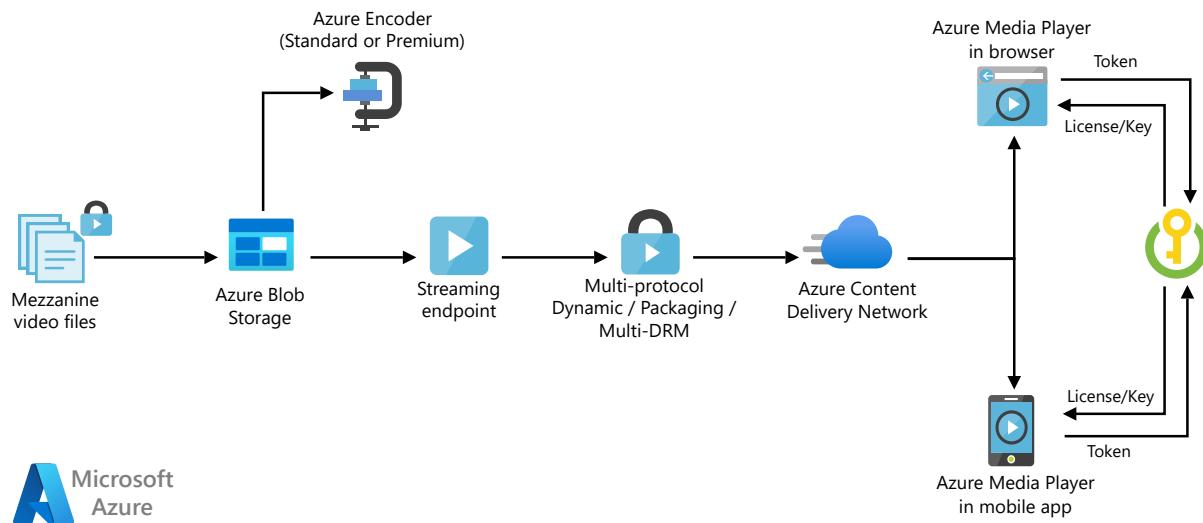
Azure Media Player

## 💡 Solution ideas

This article is a solution idea. If you'd like us to expand the content with more information, such as potential use cases, alternative services, implementation considerations, or pricing guidance, let us know by providing [GitHub feedback](#).

This solution is built on the Azure managed services: [Blob Storage](#), [Content Delivery Network](#), and [Azure Media Player](#) from [Azure Media Services](#). These services run in a high-availability environment, patched and supported, allowing you to focus on your solution instead of the environment they run in.

## Architecture



Download a [Visio file](#) of this architecture.

## Components

- **Blob Storage**: Stores large amounts of unstructured data that can be accessed from anywhere in the world via HTTP or HTTPS. You can use Blob storage to expose data publicly to the world, or to store application data privately. There are multiple options for uploading files to blob storage, including [AzCopy](#), [Media](#)

Services Azure portal, .NET SDK, or REST API, Azure CLI, Python, or one of [several Azure blob storage tools/SDKs](#).

- [Azure Media Services Encoder](#): Encoding jobs are one of the most common processing operations in Media Services. You create encoding jobs to convert media files from one encoding to another.
- [Azure Media Services Streaming Endpoint](#): A streaming service that can deliver content directly to a client player application, or to a content delivery network (CDN) for further distribution.
- [Content Delivery Network](#): Provides secure, reliable content delivery with broad global reach and a rich feature set.
- [Azure Media Player](#): Uses industry standards, such as HTML5 (MSE/EME), to provide a rich adaptive streaming experience. Regardless of the playback technology used, developers have a unified JavaScript interface to access APIs. Also, see the [Azure Media Player documentation](#).
- [Multi-DRM content protection](#): Delivers content securely using multi-DRM (PlayReady, Widevine, FairPlay Streaming) or AES Clear Key encryption.

## Scenario details

A basic video-on-demand solution that gives you the capability to stream recorded video content to any video-capable endpoint device, mobile application, or desktop browser. This content might include movies, news clips, sports segments, training videos, and customer support tutorials. Video files are uploaded to Azure Blob storage, encoded to a multi-bitrate standard format, and then distributed via all major adaptive bit-rate streaming protocols (HLS, MPEG-DASH, Smooth) to the Azure Media Player client.

You can also use other media players such as:

- [Video.js](#)
- [Shaka Player](#)
- [hls.js](#)
- [dash.js](#)
- [ExoPlayer](#)
- [AVPlayer](#)
- [THEOplayer](#)
- [NexPlayer](#)

## Potential use cases

This solution applies to television, movie, and various online streaming services.

# Next steps

- To get started with Azure Media Services, visit the [Azure Media Services](#) documentation where you will find quickstarts, tutorials, and samples.
- [Azure Media Player overview](#)
- [How to use Azure Blob storage](#)
- [How to encode an asset using Media Encoder](#)
- [How to manage streaming endpoints](#)
- [Using Azure Content Delivery Network](#)
- [Playing your content with existing players ↗](#)
- [Deliver content securely](#)

## Related resources

- [Live stream digital media](#)
- [Gridwich cloud media system](#)
- [Instant broadcasting with serverless code](#)

# Migration architecture design

Article • 02/13/2023

Azure provides resources for every stage of your cloud migration, with tools and guidance to help you move and manage your workloads.

These are just some of the key migration services available on Azure:

- [Azure Migrate](#) . Simplify migration and modernization with a unified platform.
- [Azure Database Migration Service](#) . Accelerate your data migration to Azure.
- [Azure Data Box](#) . Easily move data to Azure when busy networks aren't an option.
- [Azure App Service migration tools](#) . Quickly assess your web apps and migrate them to Azure with free, easy-to-use tools.

## Introduction to migration on Azure

If you're new to migration on Azure, the best way to learn more is with [Microsoft Learn training](#), a free online training platform. Microsoft Learn provides interactive training for Microsoft products and more.

Here are some learning paths and modules to get you started:

- [Learning path: Best practices for Azure migration and modernization](#)
- [Learning path: Migrate virtual machines and apps using Azure Migrate](#)
- [Learning path: Migrate SQL workloads to Azure](#)
- [Module: Design migrations](#)
- [Module: Applications and infrastructure migration and modernization](#)
- [Module: Migrate to Azure through repeatable processes and common tools](#)

## Path to production

For information about creating a migration plan, see [Build a migration plan with Azure Migrate](#).

## Best practices

The Cloud Adoption Framework for Azure provides proven guidance and best practices that can help you confidently adopt the cloud and achieve business outcomes. Here are some migration best practices to check out:

- Azure cloud migration best practices checklist
- Multiple datacenters
- Azure regions decision guide
- Best practices when data requirements exceed network capacity during a migration effort
- Best practices to set up networking for workloads migrated to Azure
- Deploy a migration infrastructure
- Best practices to cost and size workloads migrated to Azure
- Scale a migration to Azure
- Governance or compliance strategy

For security best practices for Azure Migrate, see [Azure security baseline for Azure Migrate](#).

## Migration architectures

The following sections provide links to reference architectures in a few high-level migration categories:

### Hyper-V migrations

- Support matrix for Hyper-V migration
- How does Hyper-V replication work?

### VMware migrations

- Support matrix for VMware migration
- Migrate workloads for Azure VMware Solution
- Azure Migrate agentless migration of VMware virtual machines
- Prepare for VMware agentless migration
- VMware agent-based migration architecture

### Mainframe migrations

- Modernize mainframe and midrange data
- General mainframe refactor to Azure
- Rehost a general mainframe on Azure
- Migrate IBM mainframe applications to Azure with TmaxSoft OpenFrame

### Oracle migrations

- Oracle database migration to Azure
- Overview of Oracle database migration
- Oracle database migration: Cross-cloud connectivity
- Oracle database migration: Lift and shift
- Oracle database migration: Refactor
- Oracle database migration: Rearchitect

## Migrations of banking systems

- Banking system cloud transformation on Azure
- Patterns and implementations for a banking cloud transformation

## Stay current with migration on Azure

Get the latest updates on [Azure migration services and features](#).

## Additional resources

### Example solutions

Following are some additional migration architectures to consider:

- Modernize .NET applications
- Migrate an e-commerce solution to Azure
- Lift and shift to containers with AKS
- Migrate an Azure Cloud Services application to Azure Service Fabric
- Migrate a monolithic application to microservices using domain-driven design
- Support matrix for migration of physical servers, AWS VMs, and GCP VMs
- Migrate a web app using Azure API Management
- JMeter implementation for a load testing pipeline

## AWS or Google Cloud professionals

### AWS

- Azure Migrate is comparable to [AWS Application Discovery Service](#). Azure Migrate assesses on-premises workloads for migration to Azure, performs performance-based sizing, and provides cost estimations.

- [Azure Database Migration Service](#) is comparable to [AWS Database Migration Service](#) . Azure Database Migration Service enables seamless migrations from multiple database sources to Azure data platforms with minimal downtime.

## Google Cloud

- [Google Cloud to Azure services comparison - Migration tools](#)

# Hadoop migration to Azure

Article • 11/08/2023

Apache Hadoop provides a distributed file system and a framework for using MapReduce techniques to analyze and transform very large data sets. An important characteristic of Hadoop is the partitioning of data and computation across many (thousands) of hosts. Computations are done in parallel close to the data. A Hadoop cluster scales computation capacity, storage capacity, and I/O bandwidth simply by adding commodity hardware.

This article is an overview of migrating Hadoop to Azure. The other articles in this section provide migration guidance for specific Hadoop components. They are:

- [Apache HDFS migration to Azure](#)
- [Apache HBase migration to Azure](#)
- [Apache Kafka migration to Azure](#)
- [Apache Sqoop migration to Azure](#)

Hadoop provides an extensive ecosystem of services and frameworks. These articles don't describe the Hadoop components and Azure implementations of them in detail. Instead, they provide high-level guidance and considerations to serve as a starting point for you to migrate your on-premises and cloud Hadoop applications to Azure.

*Apache<sup>®</sup>, Apache Spark<sup>®</sup>, Apache Hadoop<sup>®</sup>, Apache HBase<sup>®</sup>, Apache Hive<sup>®</sup>, Apache Ranger<sup>®</sup>, Apache Sentry<sup>®</sup>, Apache ZooKeeper<sup>®</sup>, Apache Storm<sup>®</sup>, Apache Sqoop<sup>®</sup>, Apache Flink<sup>®</sup>, Apache Kafka<sup>®</sup>, and the flame logo are either registered trademarks or trademarks of the Apache Software Foundation in the United States and/or other countries. No endorsement by The Apache Software Foundation is implied by the use of these marks.*

## Hadoop components

The key components of a Hadoop system are listed in the following table. For each component there's a brief description, and migration information such as:

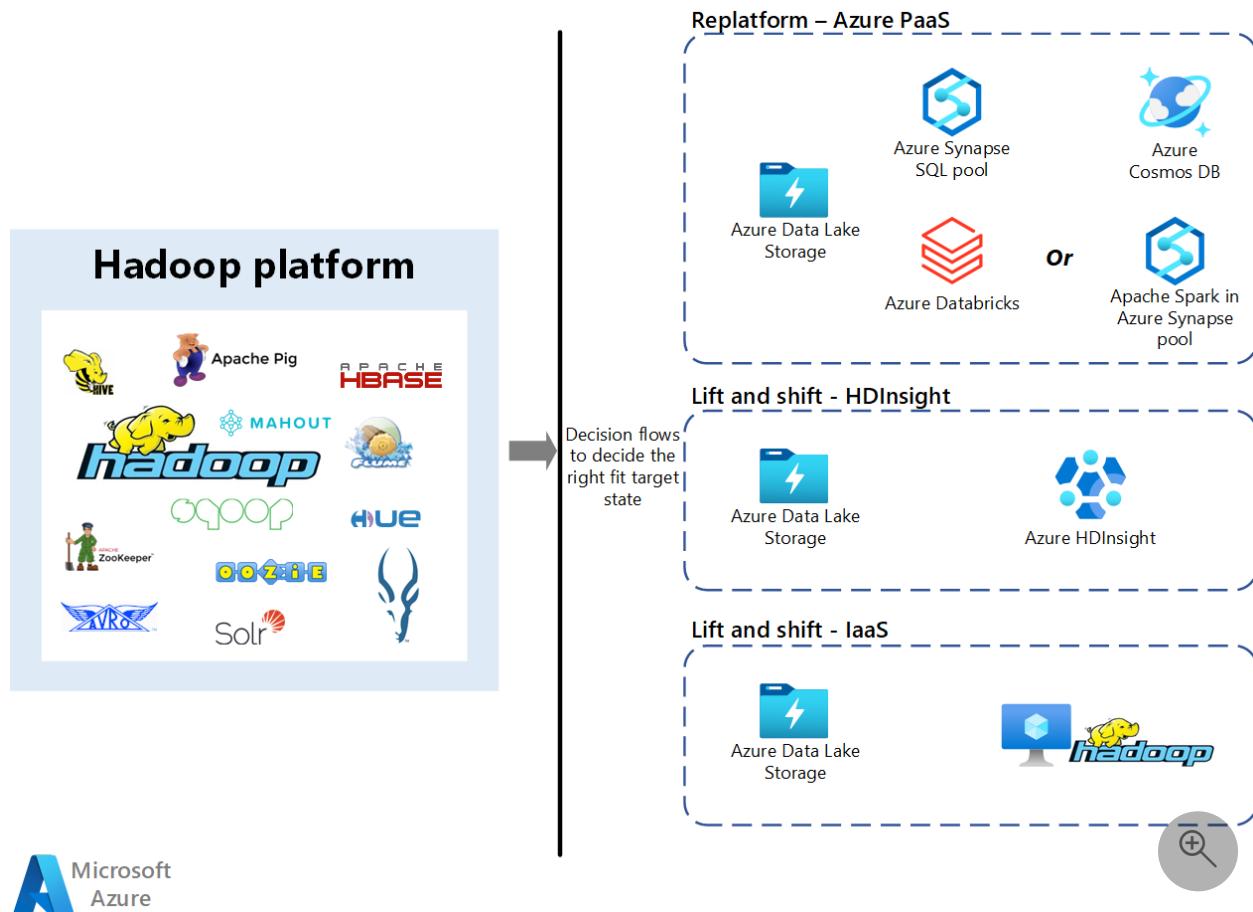
- Links to decision flowcharts for deciding on migration strategies
- A list of possible Azure target services

Component	Description	Decision flowcharts	Targeted Azure services
Apache HDFS	Distributed file system	Planning the data migration, Pre-checks prior to data migration	Azure Data Lake Storage
Apache HBase	Column-oriented table service	Choosing landing target for Apache HBase, Choosing storage for Apache HBase on Azure	HBase on a virtual machine (VM), HBase in Azure HDInsight, Azure Cosmos DB
Apache Spark	Data processing framework	Choosing landing target for Apache Spark on Azure	Spark in HDInsight, Azure Synapse Analytics, Azure Databricks
Apache Hive	Data warehouse infrastructure	Choosing landing target for Hive, Selecting target DB for Hive metadata	Hive on a VM, Hive in HDInsight, Azure Synapse Analytics
Apache Ranger	Framework for monitoring and managing data security		Enterprise Security Package for HDInsight, Microsoft Entra ID, Ranger on a VM
Apache Sentry	Framework for monitoring and managing data security	Choosing landing targets for Apache Sentry on Azure	Sentry and Ranger on a VM, Enterprise Security Package for HDInsight, Microsoft Entra ID
Apache MapReduce	Distributed computation framework		MapReduce, Spark
Apache Zookeeper	Distributed coordination service		ZooKeeper on a VM, built-in solution in platform as a service (PaaS)
Apache YARN	Resource manager for Hadoop ecosystem		YARN on a VM, built-in solution in PaaS
Apache Sqoop	Command line interface tool for transferring data between Apache Hadoop clusters and relational databases	Choosing landing targets for Apache Sqoop on Azure	Sqoop on a VM, Sqoop in HDInsight, Azure Data Factory

Component	Description	Decision flowcharts	Targeted Azure services
Apache Kafka	Highly scalable fault-tolerant distributed messaging system	Choosing landing targets for Apache Kafka on Azure	Kafka on a VM, Event Hubs for Kafka, Kafka on HDInsight
Apache Atlas	Open source framework for data governance and metadata management		Azure Purview

## Migration approaches

The following diagram shows three approaches to migrating Hadoop applications:



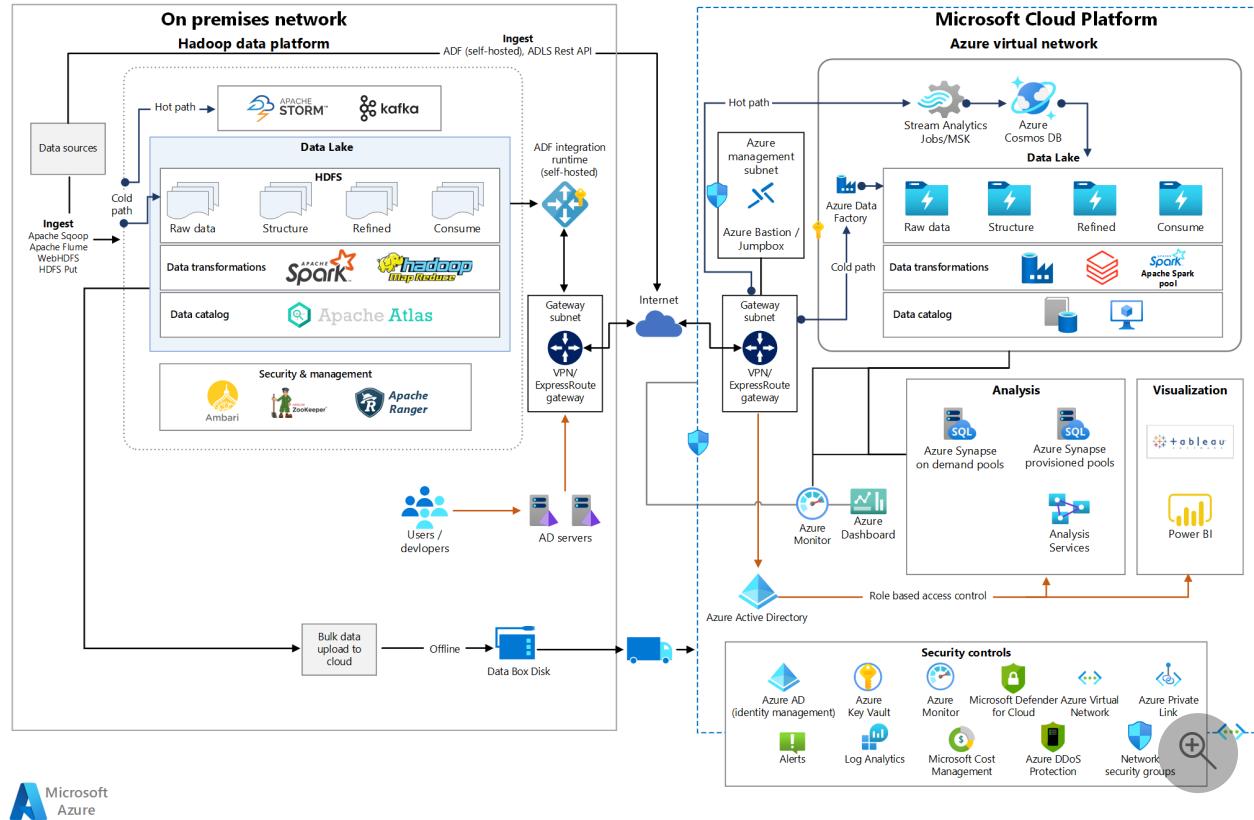
Download a [Visio file](#) of this architecture.

The approaches are:

- **Replatform by using Azure PaaS:** For more information, see [Modernize by using Azure Synapse Analytics and Databricks](#).
- **Lift and shift to HDInsight:** For more information, see [Lift and shift to HDInsight](#).
- **Lift and shift to IaaS:** For more information, see [Lift and shift to Azure infrastructure as a service \(IaaS\)](#).

# Modernize by using Azure Synapse Analytics and Databricks

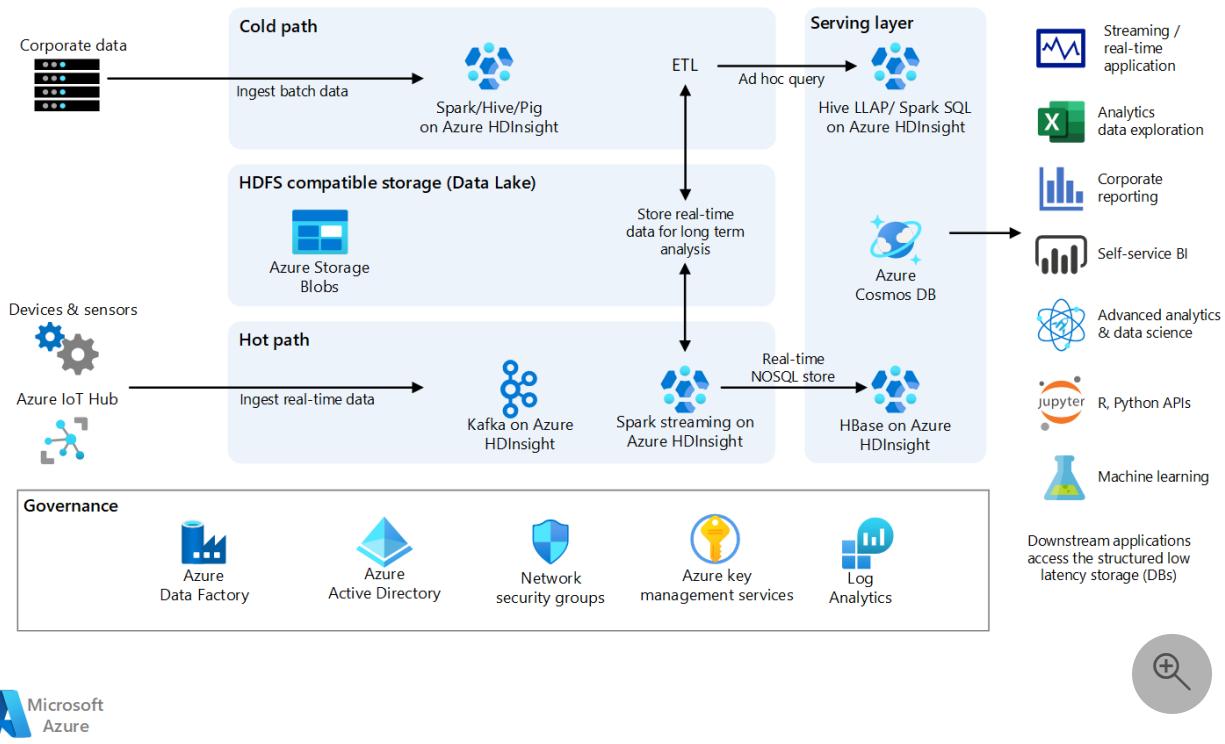
The following diagram shows this approach:



Download a [Visio file](#) of this architecture.

## Lift and shift to HDInsight

The following diagram shows this approach:

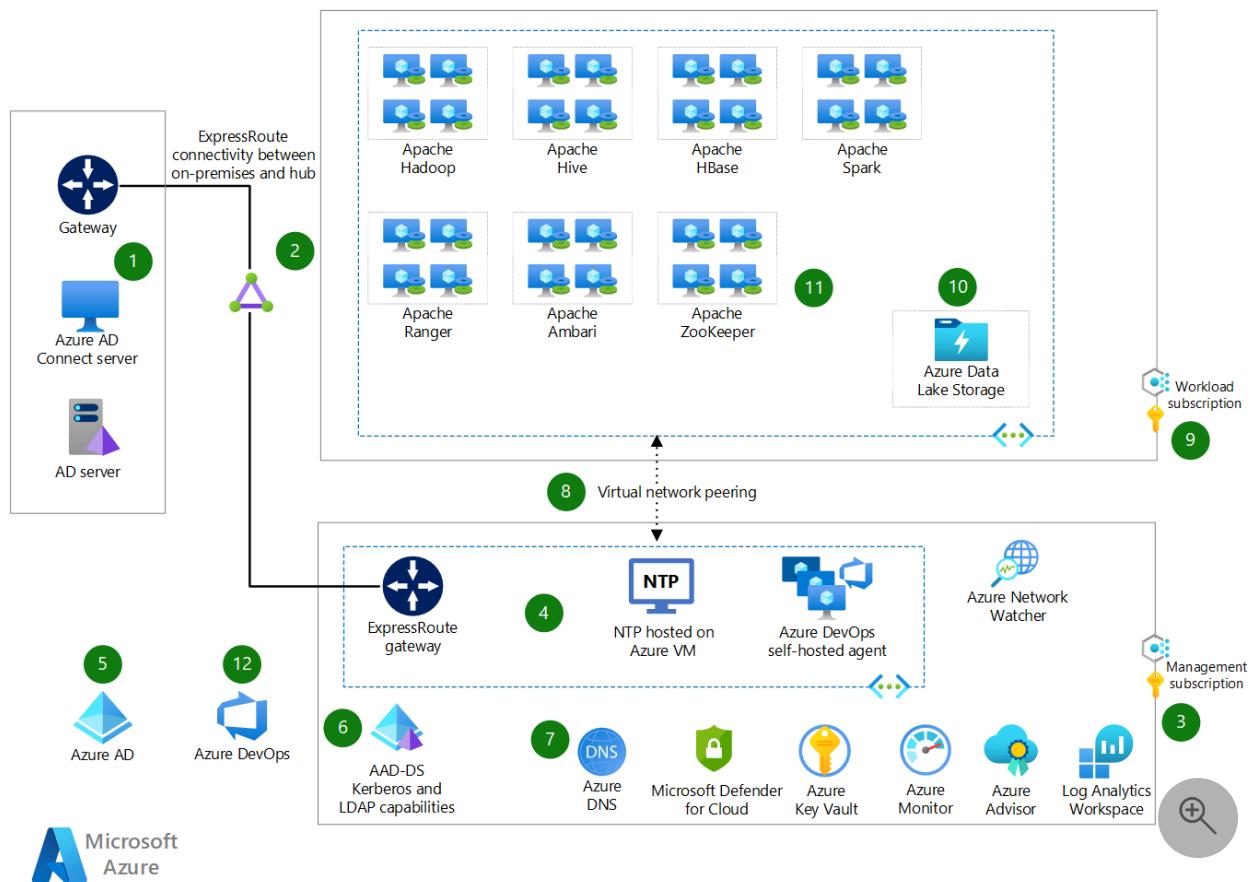


Download a [Visio file](#) of this architecture.

For more information, see [Guide to migrating Big Data workloads to Azure HDInsight](#). That article provides a link for downloading a migration guide and also provides an email address that you can use to ask questions or make suggestions.

## Lift and shift to Azure infrastructure as a service (IaaS)

The following pattern presents a point of view on how to deploy OSS on Azure IaaS with a tight integration back to on-premises systems such as Active Directory, Domain Controller, and DNS. The deployment follows enterprise-scale landing zone guidance from Microsoft. Management capabilities such as monitoring, security, governance, and networking are hosted within a management subscription. The workloads, all IaaS-based, are hosted in a separate subscription. For more information about enterprise-scale landing zones, see [What is an Azure landing zone?](#).



Download a [Visio file](#) of this architecture.

1. On-premises Active Directory synchronizes with Microsoft Entra ID by using Microsoft Entra Connect hosted on-premises.
2. Azure ExpressRoute provides secure and private network connectivity between on-premises and Azure.
3. The management (or hub) subscription provides networking and management capabilities for the deployment. This pattern is in line with enterprise-scale landing zone guidance from Microsoft.
4. The services hosted inside the hub subscription provide network connectivity and management capabilities.
  - **NTP (hosted on Azure VM)** is required to keep the clocks synchronized across all virtual machines. When you run multiple applications, such as HBase and ZooKeeper, you should run a Network Time Protocol (NTP) service or another time-synchronization mechanism on your cluster. All nodes should use the same service for time synchronization. For instructions on setting up NTP on Linux, see [14.6. Basic NTP configuration](#).
  - **Azure Network Watcher** provides tools to monitor, diagnose, and manage resources in an Azure virtual network. Network Watcher is designed to monitor and repair the network health of IaaS products, including VMs, virtual networks, application gateways, and load balancers.

- **Azure Advisor** analyzes your resource configuration and usage telemetry and then recommends solutions to improve the cost effectiveness, performance, reliability, and security of your Azure resources.
- **Azure Monitor** provides a comprehensive solution for collecting, analyzing, and acting on telemetry from your cloud and on-premises environments. It helps you understand how your applications are performing so that you can proactively identify issues that affect the applications and the resources they depend on.
- **Log Analytics Workspace** is a unique environment for Azure Monitor log data. Each workspace has its own data repository and configuration. Data sources and solutions are configured to store their data in a particular workspace. You need a Log Analytics workspace if you intend to collect data from the following sources:
  - Azure resources in your subscription
  - On-premises computers that are monitored by System Center Operations Manager
  - Device collections from System Center Configuration Manager
  - Diagnostics or log data from Azure Storage
- **Azure DevOps Self-Hosted Agent** hosted on Azure virtual Machine Scale Sets gives you flexibility over the size and the image of machines on which agents run. You specify a virtual machine scale set, a number of agents to keep on standby, a maximum number of virtual machines in the scale set. Azure Pipelines manages the scaling of your agents for you.

5. The **Microsoft Entra ID** tenant is synchronized with the on-premises Active Directory via Microsoft Entra Connect synchronization services. For more information, see [Microsoft Entra Connect Sync: Understand and customize synchronization](#).
6. **Microsoft Entra Domain Services (Microsoft Entra Domain Services)** provides LDAP and Kerberos capabilities on Azure. When you first deploy Microsoft Entra Domain Services, an automatic one-way synchronization is configured and started in order to replicate the objects from Microsoft Entra ID. This one-way synchronization continues to run in the background to keep the Microsoft Entra Domain Services managed domain up-to-date with any changes from Microsoft Entra ID. No synchronization occurs from Microsoft Entra Domain Services back to Microsoft Entra ID.
7. Services such as **Azure DNS**, **Microsoft Defender for Cloud**, and **Azure Key Vault** sit inside the management subscription and provide service/IP address resolution, unified infrastructure security management, and certificate and key management capabilities, respectively.

8. [Virtual Network Peering](#) provides connectivity between virtual networks deployed in two subscriptions: management (hub) and workload (spoke).
9. In line with enterprise-scale landing zones, workload subscriptions are used for hosting application workloads.
10. [Azure Data Lake Storage](#) is a set of capabilities that are built on Azure Blob Storage to do big data analytics. In the context of big data workloads, Data Lake Storage can be used as secondary storage for Hadoop. Data written to Data Lake Storage can be consumed by other Azure services that are outside of the Hadoop framework.
11. **Big data workloads** are hosted on a set of independent Azure virtual machines. Refer to guidance for [HDFS](#), [HBase](#), [Hive](#), [Ranger](#), and [Spark](#) on Azure IaaS for more information.
12. [Azure DevOps](#) is a software as a service (SaaS) offering that provides an integrated set of services and tools to manage your software projects, from planning and development through testing and deployment.

## End state reference architecture

One of the challenges of migrating workloads from on-premises Hadoop to Azure is deploying to achieve the desired end state architecture and application. The project that's described in [Hadoop Migration on Azure PaaS](#) is intended to reduce the significant effort that's usually needed to deploy the PaaS services and the application.

In that project, we look at the end state architecture for big data workloads on Azure and list the components that are used in a Bicep template deployment. With Bicep we deploy only the modules that we need to deploy architecture. We cover the prerequisites for the template and the various methods of deploying the resources on Azure, such as One-click, Azure CLI, GitHub Actions, and Azure DevOps Pipeline.

## Contributors

*This article is maintained by Microsoft. It was originally written by the following contributors.*

Principal authors:

- [Namrata Maheshwary](#) | Senior Cloud Solution Architect
- [Raja N](#) | Director, Customer Success
- [Hideo Takagi](#) | Cloud Solution Architect
- [Ram Yerrabotu](#) | Senior Cloud Solution Architect

Other contributors:

- [Ram Baskaran](#) | Senior Cloud Solution Architect
- [Jason Bouska](#) | Senior Software Engineer
- [Eugene Chung](#) | Senior Cloud Solution Architect
- [Pawan Hosatti](#) | Senior Cloud Solution Architect - Engineering
- [Daman Kaur](#) | Cloud Solution Architect
- [Danny Liu](#) | Senior Cloud Solution Architect - Engineering
- [Jose Mendez](#) | Senior Cloud Solution Architect
- [Ben Sadeghi](#) | Senior Specialist
- [Sunil Sattiraju](#) | Senior Cloud Solution Architect
- [Amanjeet Singh](#) | Principal Program Manager
- [Nagaraj Seeplapudur Venkatesan](#) | Senior Cloud Solution Architect - Engineering

*To see non-public LinkedIn profiles, sign in to LinkedIn.*

## Next steps

### Azure product introductions

- [Introduction to Azure Data Lake Storage Gen2](#)
- [What is Apache Spark in Azure HDInsight?](#)
- [What is Apache Hadoop in Azure HDInsight?](#)
- [What is Apache HBase in Azure HDInsight?](#)
- [What is Apache Kafka in Azure HDInsight?](#)
- [Overview of enterprise security in Azure HDInsight](#)

### Azure product reference

- [Microsoft Entra documentation](#)
- [Azure Cosmos DB documentation](#)
- [Azure Data Factory documentation](#)
- [Azure Databricks documentation](#)
- [Azure Event Hubs documentation](#)
- [Azure Functions documentation](#)
- [Azure HDInsight documentation](#)
- [Microsoft Purview data governance documentation](#)
- [Azure Stream Analytics documentation](#)
- [Azure Synapse Analytics](#)

## Other

- Enterprise Security Package for Azure HDInsight
- Develop Java MapReduce programs for Apache Hadoop on HDInsight
- Use Apache Sqoop with Hadoop in HDInsight
- Overview of Apache Spark Streaming
- Structured Streaming tutorial
- Use Azure Event Hubs from Apache Kafka applications

## Related resources

- Apache HDFS migration to Azure
- Apache HBase migration to Azure
- Apache Kafka migration to Azure
- Apache Sqoop migration to Azure

# Apache Hadoop Distributed File System (HDFS) migration to Azure

Azure HDInsight   Azure Cosmos DB   Azure Data Lake Storage   Azure Synapse Analytics

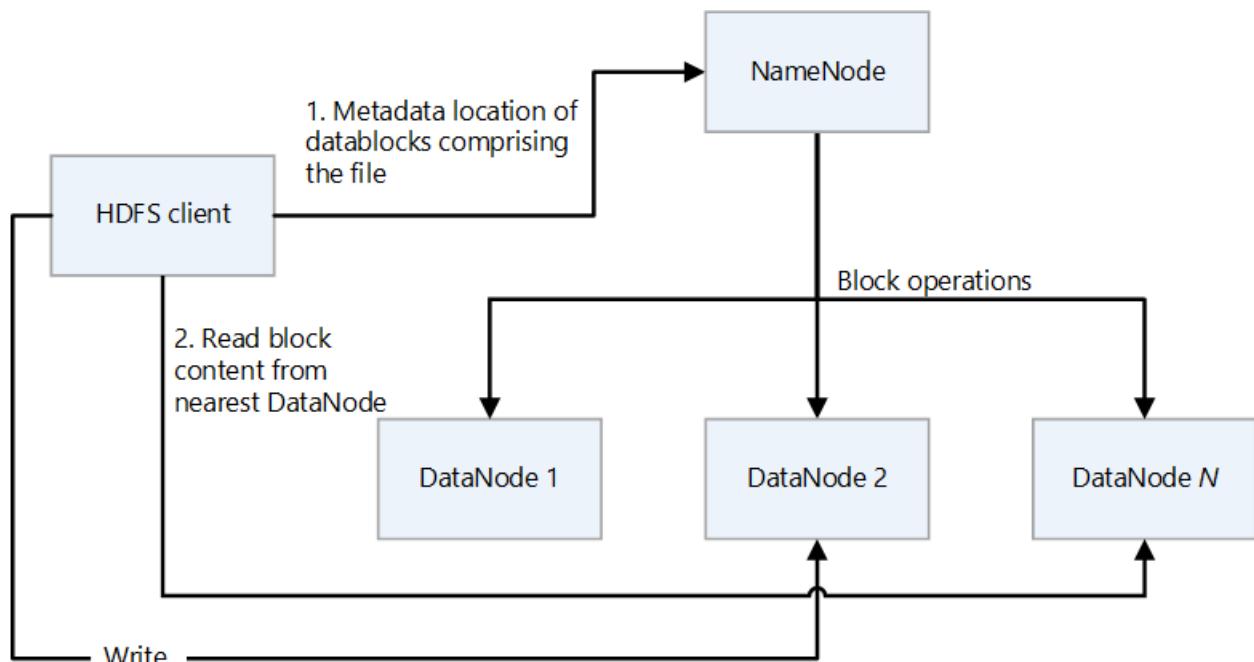
Azure Stream Analytics

The [Hadoop Distributed File System \(HDFS\)](#) is a Java-based distributed file system that provides reliable, scalable data storage that can span large clusters of commodity servers. This article provides an overview of HDFS and a guide to migrating it to Azure.

*Apache®, Apache Spark®, Apache Hadoop®, Apache Hive®, and the flame logo are either registered trademarks or trademarks of the Apache Software Foundation in the United States and/or other countries. No endorsement by The Apache Software Foundation is implied by the use of these marks.*

## HDFS Architecture and Components

HDFS has a primary/secondary design. In the following diagram, NameNode is the primary and the DataNodes are the secondaries.



- **NameNode** manages access to files and to the file system namespace, which is a hierarchy of directories.
  - Files and directories are nodes on NameNode. They have attributes such as permissions, modification and access times, and size quotas for namespace and disk space.

- A file comprises multiple blocks. The default block size is 128 megabytes. A non-default block size can be set for a cluster by modifying the `hdfs-site.xml` file.
  - Each block of the file is independently replicated at multiple DataNodes. The default value for the replication factor is three, but every cluster can have its own non-default value. The replication factor can be changed at any time. A change causes a cluster re-balancing.
  - NameNode maintains the namespace tree and the mapping of file blocks to DataNodes (the physical locations of file data).
  - When an HDFS client reads a file:
    1. It contacts the NameNode for the locations of the data blocks of the file.
    2. It reads block contents from the nearest DataNode.
  - HDFS keeps the entire namespace in RAM.
- **DataNodes** are the secondary nodes that perform read and write operations on the file system, and perform block operations such as creation, replication, and deletion.
    - A DataNode contains metadata files that hold the checksums of the stored files. For each block replica that's hosted by a DataNode, there's a corresponding metadata file that contains metadata about the replica, including its checksum information. The metadata file has the same base name as the block file, and extension `.meta`.
    - The DataNode contains the data file that holds the block's data.
    - When a DataNode reads a file, it fetches the block locations and replica locations from the NameNode and tries to read from the nearest location.
    - An HDFS cluster can have thousands of DataNodes and tens of thousands of HDFS clients per cluster. Each DataNode can execute multiple application tasks concurrently.
    - An end-to-end checksum calculation is performed as part of the HDFS write pipeline when a block is written to DataNodes.
  - **HDFS Client** is the client that applications use to access files.
    - It's a code library that exports the HDFS file system interface.
    - It supports operations to read, write, and delete files, and operations to create and delete directories.
    - It performs the following steps when an application reads a file:
      - It gets from the NameNode a list of DataNodes and locations that hold the file blocks. The list includes the replicas.
      - It uses the list to get the requested blocks from the DataNodes.
    - HDFS provides an API that exposes the locations of file blocks. This allows applications like the MapReduce framework to schedule a task to run where the

data is, in order to optimize read performance.

## Feature map

The Azure Blob Filesystem (ABFS) driver provides an interface that makes it possible for Azure Data Lake Storage to act as an HDFS file system. The following table compares the core functionality of the ABFS driver and Data Lake Storage to that of HDFS.

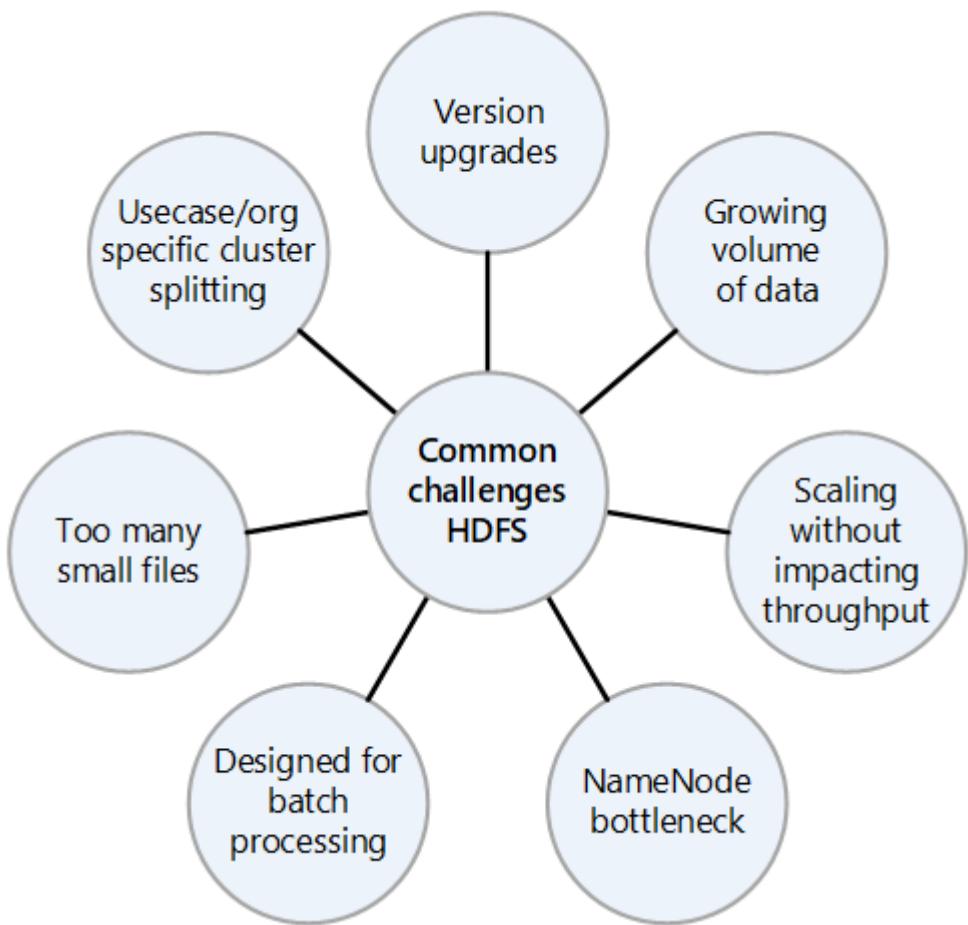
[\[+\] Expand table](#)

Feature	The ABFS driver and Data Lake Storage	HDFS
<b>Access that's compatible with Hadoop</b>	You can manage and access data just as you would with HDFS. The ABFS driver is available in all Apache Hadoop environments, including Azure HDInsight and Azure Databricks.	A MapR cluster can access an external HDFS cluster with the <code>hdfs://</code> or <code>webhdfs://</code> protocols
<b>POSIX permissions</b>	The security model for Data Lake Gen2 supports access control list (ACL) and POSIX permissions along with some extra granularity that's specific to Data Lake Storage Gen2. Settings can be configured by using admin tools or frameworks like Apache Hive and Apache Spark.	Jobs that require file system features like strictly atomic directory renames, fine-grained HDFS permissions, or HDFS symlinks can only work on HDFS.
<b>Cost effectiveness</b>	Data Lake Storage offers low-cost storage capacity and transactions. Azure Blob Storage lifecycles help to lower costs by adjusting billing rates as data moves through its lifecycle.	
<b>Optimized driver</b>	The ABFS driver is optimized for big data analytics. The corresponding REST APIs are	

Feature	The ABFS driver and Data Lake Storage	HDFS
	provided through the distributed file system (DFS) endpoint, <code>dfs.core.windows.net</code> .	
Block Size	<b>Blocks</b> is equivalent to a single <b>Append</b> API invocation (the <b>Append</b> API creates a new block) and is limited to 100 MB per invocation. However, the write pattern supports calling <b>Append</b> many times per file (even in parallel) to a maximum of 50,000 and then calling <b>Flush</b> (equivalent to <b>PutBlockList</b> ). This is the way the maximum files size of 4.75TB is achieved.	HDFS stores the data in a data block. You set the block size by setting a value in the <code>hdfs-site.xml</code> file in the Hadoop directory. The default size is 128 MB.
Default ACLs	Files don't have default ACLs and aren't enabled by default.	Files don't have default ACLs.
Binary Files	Binary files can be moved to Azure Blob Storage in a non-hierarchical namespace. Objects in Blob Storage are accessible via the Azure Storage REST API, Azure PowerShell, Azure CLI, or an Azure Storage client library. Client libraries are available for different languages, including .NET, Java, Node.js, Python, Go, PHP, and Ruby	Hadoop provides the ability to read and write binary files. SequenceFile is a flat file that consists of a binary key and value pairs. The SequenceFile provides Writer, Reader, and Sorter classes for writing, reading, and sorting. Convert the image or video file into a SequenceFile and store it in HDFS. Then use the HDFS SequenceFileReader/Writer methods or the put command: <code>bin/hadoop fs -put /src_image_file /dst_image_file</code>
Permission inheritance	Data Lake Storage uses the POSIX-style model and behaves the same as Hadoop if ACLs control access to an object. For more information,	Permissions for an item are stored on the item itself, not inherited after the item exists. Permissions are only inherited if default permissions are set on the parent item before the child item is created.

Feature	The ABFS driver and Data Lake Storage	HDFS
	see <a href="#">Access control lists (ACLs)</a> in <a href="#">Data Lake Storage Gen2</a> .	
<b>Data replication</b>	Data in an Azure Storage account is replicated three times in the primary region. Zone-redundant storage is the recommended replication option. It synchronously replicates across three Azure availability zones in the primary region.	By default a file's replication factor is three. For critical files or files that are accessed often, a higher replication factor improves fault tolerance and increases read bandwidth.
<b>Sticky bit</b>	In the context of Data Lake Storage, it's unlikely that the sticky bit is required. Briefly, if the sticky bit is enabled on a directory, a child item can only be deleted or renamed by the user that owns the child item. The sticky bit isn't shown in the Azure portal.	The sticky bit can be set on directories to prevent anyone except the superuser, directory owner, or file owner from deleting or moving files within the directory. Setting the sticky bit for a file has no effect.

## Common challenges of an on-premises HDFS



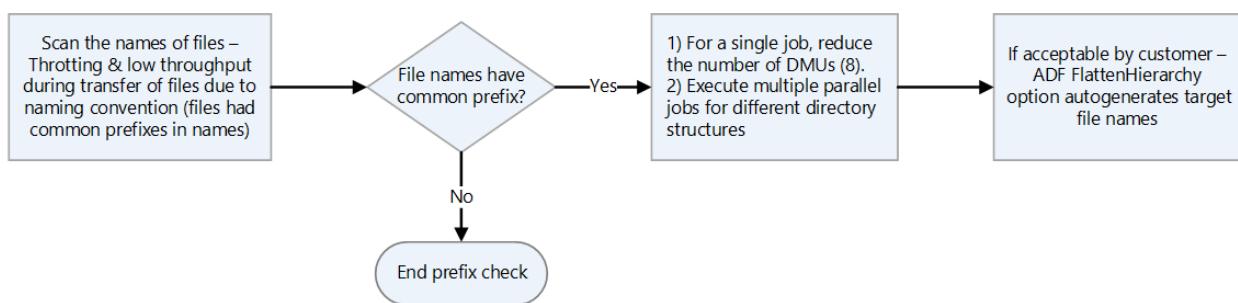
The many challenges presented by an on-premises HDFS implementation can be reasons to consider the advantages of migrating to the cloud:

- Frequent HDFS version upgrades
- Increasing amounts of data
- Having many small files that increase the pressure on the NameNode, which controls the metadata of all the files in the cluster. More files often means more read traffic on the NameNode when clients read the files, and more calls when clients are writing.
- If multiple teams in the organization require different datasets, splitting the HDFS clusters by use case or organization isn't possible. The result is that data duplication increases, which increases costs and reduces efficiency.
- The NameNode can become a performance bottleneck as the HDFS cluster is scaled up or out.
- Prior to Hadoop 2.0, all client requests to an HDFS cluster first pass through the NameNode, because all the metadata is stored in a single NameNode. This design makes the NameNode a possible bottleneck and single point of failure. If the NameNode fails, the cluster is unavailable.

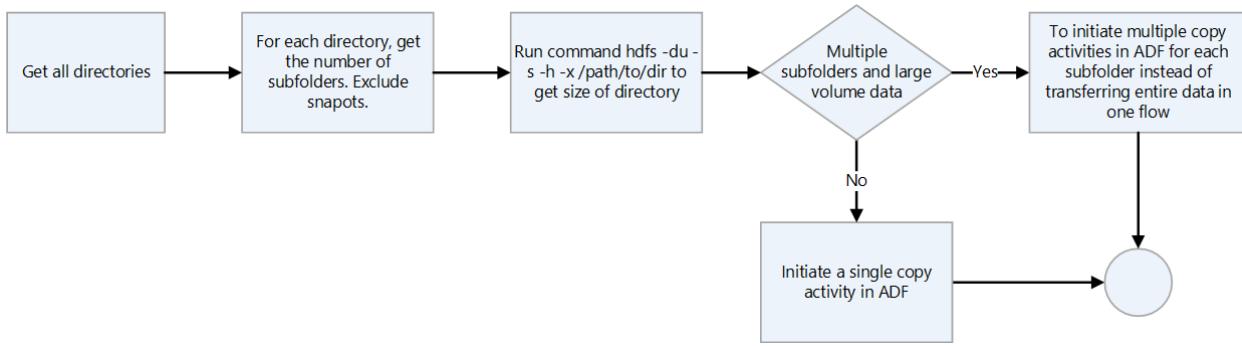
## Migration considerations

Here are some things that are important to consider when you plan a migration of HDFS to Data Lake Storage:

- Consider aggregating data that's in small files into a single file on Data Lake Storage.
- List all the directory structures in HDFS and replicate similar zoning in Data Lake Storage. You can obtain the directory structure of HDFS by using the `hdfs -ls` command.
- List all the roles that are defined in the HDFS cluster so that you can replicate them in the target environment.
- Note the data lifecycle policy of the files that are stored in HDFS.
- Keep in mind that some system features of HDFS aren't available on Data Lake Storage, including:
  - Strictly atomic renaming of directories
  - Fine-grained HDFS permissions
  - HDFS symlinks
- Azure Storage has geo-redundant replication, but it's not always wise to use it. It does provide data redundancy and geographic recovery, but a failover to a more distant location can severely degrade performance and incur additional costs. Consider whether the higher availability of the data is worth it.
- If files have names with the same prefixes, HDFS treats them as a single partition. Therefore, if you use Azure Data Factory, all data movement units (DMUs) write to a single partition.



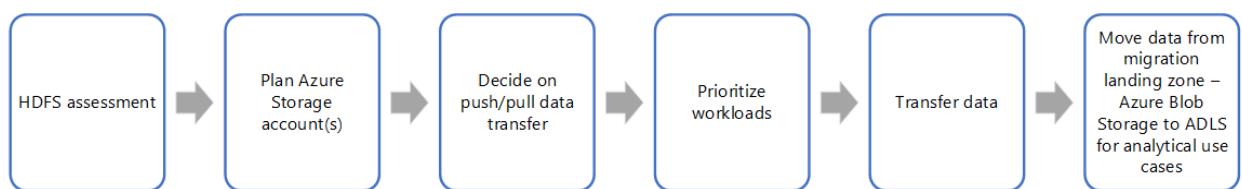
- If you use Data factory for data transfer, scan through each directory, excluding snapshots, and check the directory size by using the `hdfs du` command. If there are multiple subdirectories and large amounts of data, initiate multiple copy activities in Data Factory. For example, use one copy per subdirectory rather than transferring the entire directory by using a single copy activity.



- Data platforms are often used for longer term retention of information that may have been removed from systems of record. You should plan to create tape backups or snapshots of the archived data. Consider replicating the information to a recovery site. Usually data is archived either for compliance or for historical data purposes. Before you archive data you should have a clear reason for keeping it. Also, decide when archived data is to be removed and establish processes to remove it at that time.
- The low cost of the Archive access tier of Data Lake Storage makes it an attractive option for archiving data. For more information, see [Archive access tier](#).
- When an HDFS client uses the ABFS driver to access Blob Storage, there can be instances where the method that's used by the client isn't supported and AzureNativeFileSystem throws an UnsupportedOperationException. For example, `append(Path f, int bufferSize, Progressable progress)` isn't currently supported. To check on issues related to the ABFS driver, see [Hadoop features and fixes](#).
- There's a backported version of the ABFS driver for use on older Hadoop clusters. For more information, see [Backport for ABFS Driver](#).
- In an Azure virtual networking environment, the DistCp tool doesn't support Azure ExpressRoute private peering with an Azure Storage virtual network endpoint. For more information, see [Use Azure Data Factory to migrate data from an on-premises Hadoop cluster to Azure Storage](#).

## Migration approach

The typical approach to migrating HDFS to Data Lake Storage uses these steps:



## HDFS assessment

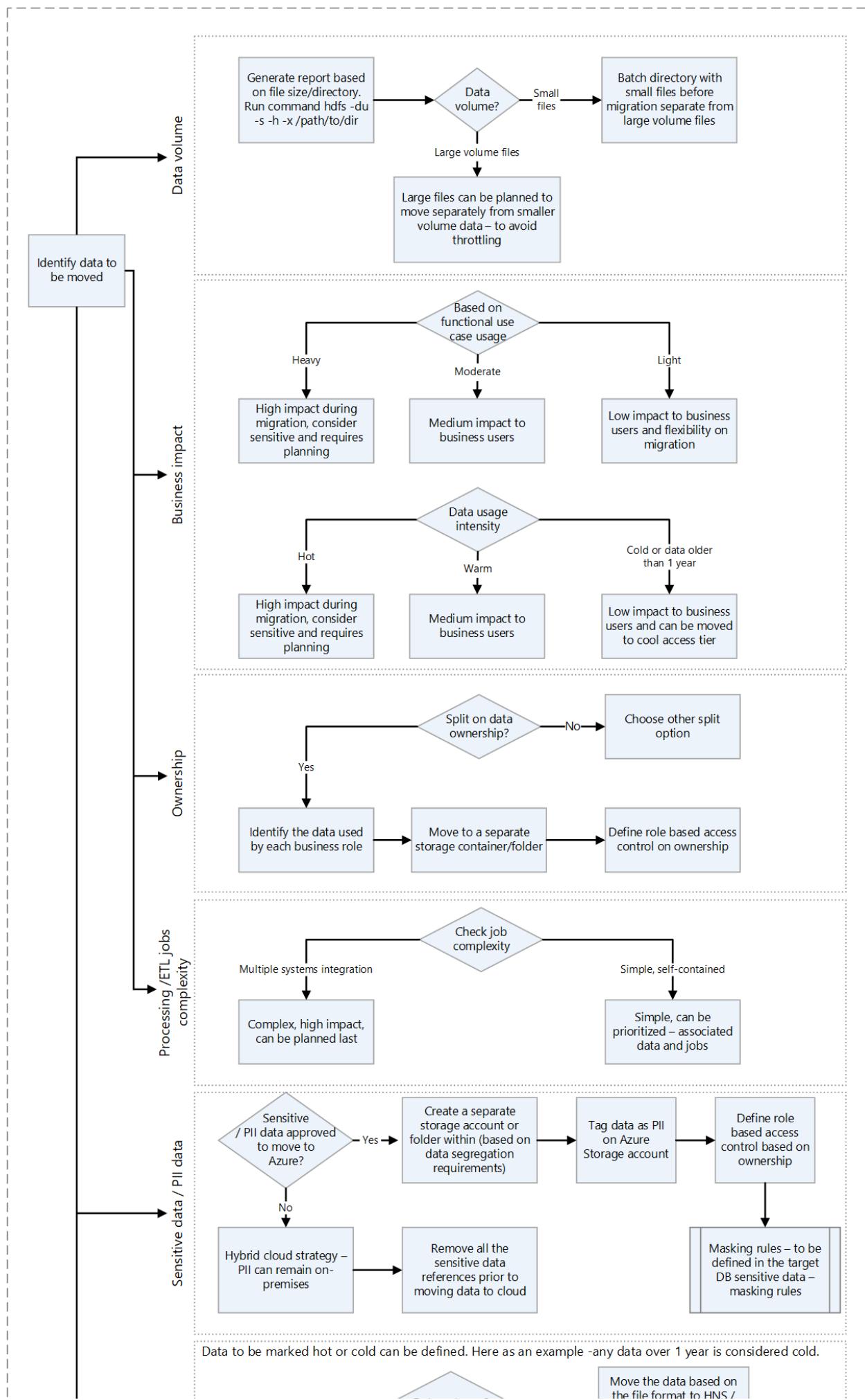
On-premises assessment scripts provide information that helps you to determine which workloads can be migrated to Azure and whether the data should be migrated all at once or a piece at a time. Third-party tools like Unravel can provide metrics and support auto-assessment of the on-premises HDFS. Some important factors to consider when planning include:

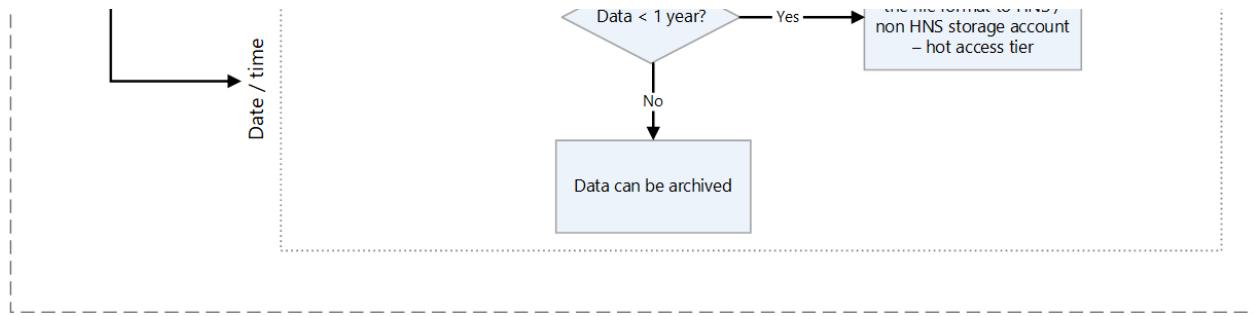
- Data volume
- Business impact
- Ownership of data
- Processing complexity
- Extract, transfer, and load (ETL) complexity
- Personally identifiable information (PII) and other sensitive data

Based on such factors, you can formulate a plan to move data to Azure that minimizes downtime and business disruption. Perhaps sensitive data can remain on-premises. Historical data can be moved and tested prior to moving an incremental load.

The following decision flow helps decide the criteria and commands to run to get the right information.

## 1. Data migration strategy/planning





HDFS commands for getting assessment metrics from HDFS include:

- List all the directories in a location:

```
hdfs dfs -ls books
```

- Recursively list all files in a location:

```
hdfs dfs -ls -R books
```

- Get the size of the HDFS directory and files:

```
hadoop fs -du -s -h command
```

The `hadoop fs -du -s -h` command displays the size of the HDFS files and directory. Since the Hadoop file system replicates every file, the actual physical size of the file is the number of file replicas multiplied by the size of one replica.

- Determine whether ACLs are enabled. To do this, obtain the value of `dfs.namenode.acls.enabled` in `Hdfs-site.xml`. Knowing the value helps in planning access control on the Azure Storage account. For information about the contents of this file, see [Default file settings](#).

Partner tools such as Unravel provide assessment reports for planning data migration. The tools must run in the on-premises environment or connect to the Hadoop cluster to generate reports.

- The following Unravel report provides statistics, per directory, about the small files in the directory:

Small File Report completed successfully.

Cluster: default [304s...200s]

Report Created at: 10/21/2020 10:06:17

Search...

Path	Files	Avg File Size	Total File Size	Min File Size	Max File Size
/	21079	5 B	121.99 kB	0 B	512 B
/warehouse/tablespace	19602	1 B	20.19 kB	0 B	165 B
/warehouse	19602	1 B	20.19 kB	0 B	165 B
/warehouse/tablespace/managed/hive	19595	1 B	19.23 kB	1 B	48 B
/warehouse/tablespace/managed	19595	1 B	19.23 kB	1 B	48 B
/warehouse/tablespace/managed/hive/tpcds_bin_partitioned_orc_5.db	11737	1 B	11.48 kB	1 B	1 B
/warehouse/tablespace/managed/hive/tpcds_bin_partitioned_orc_2.db	7809	1 B	2.63 kB	1 B	1 B
/warehouse/tablespace/managed/hive/tpcds_bin_partitioned_orc_5.db/web_returns	2159	1 B	2.11 kB	1 B	1 B
/warehouse/tablespace/managed/hive/tpcds_bin_partitioned_orc_2.db/web_returns	2140	1 B	2.09 kB	1 B	1 B

- The following report provides statistics, per directory, about the files in the directory:

File Reports completed successfully.

Cluster: default [304s...200s]

Search...

Path	Files	Avg File Size	Total File Size	Min File Size	Max File Size
/	36654	2.64 MB	100.87 GB	0 B	7.09 GB
/user	38360	390.71 kB	14.29 GB	0 B	223.05 MB
/user/hive	36653	319.97 kB	11.00 GB	0 B	26.93 MB
/user/hive/warehouse		319.97 kB	11.00 GB	0 B	26.93 MB
/user/hive/warehouse/tpcds_bin_partitioned_orc_30.db	11789	622.28 kB	7.00 GB	413 B	26.93 MB
/user/hive/warehouse/tpcds_bin_partitioned_orc_15.db	11769	309.30 kB	—	413 B	26.04 MB
/user/hive/warehouse/tpcds_bin_partitioned_orc_2.db	11727	47.71 kB	546.43 MB	413 B	12.58 MB
/user/hive/warehouse/tpcds_bin_partitioned_orc_30.db/web_returns	2176	54.53 kB	115.88 MB	1.55 kB	2.78 MB
/user/hive/warehouse/tpcds_bin_partitioned_orc_15.db/web_returns	2166	29.83 kB	60.99 MB	1.58 kB	1.35 MB
/user/hive/warehouse/tpcds_bin_partitioned_orc_2.db/web_returns	6.13 kB	12.82 MB	1.55 kB	180.02 kB	

## Transfer data

Data must be transferred to Azure as outlined in your migration plan. Transferring requires the following activities:

### 1. Identify all of the ingestion points.

If, because of security requirements, data can't be landed to the cloud directly, then on-premises can serve as an intermediate landing zone. You can build pipelines in Data Factory to pull the data from on-premises systems, or use AZCopy scripts to push the data to the Azure Storage account.

Common ingestion sources include:

- SFTP server
- File ingestion

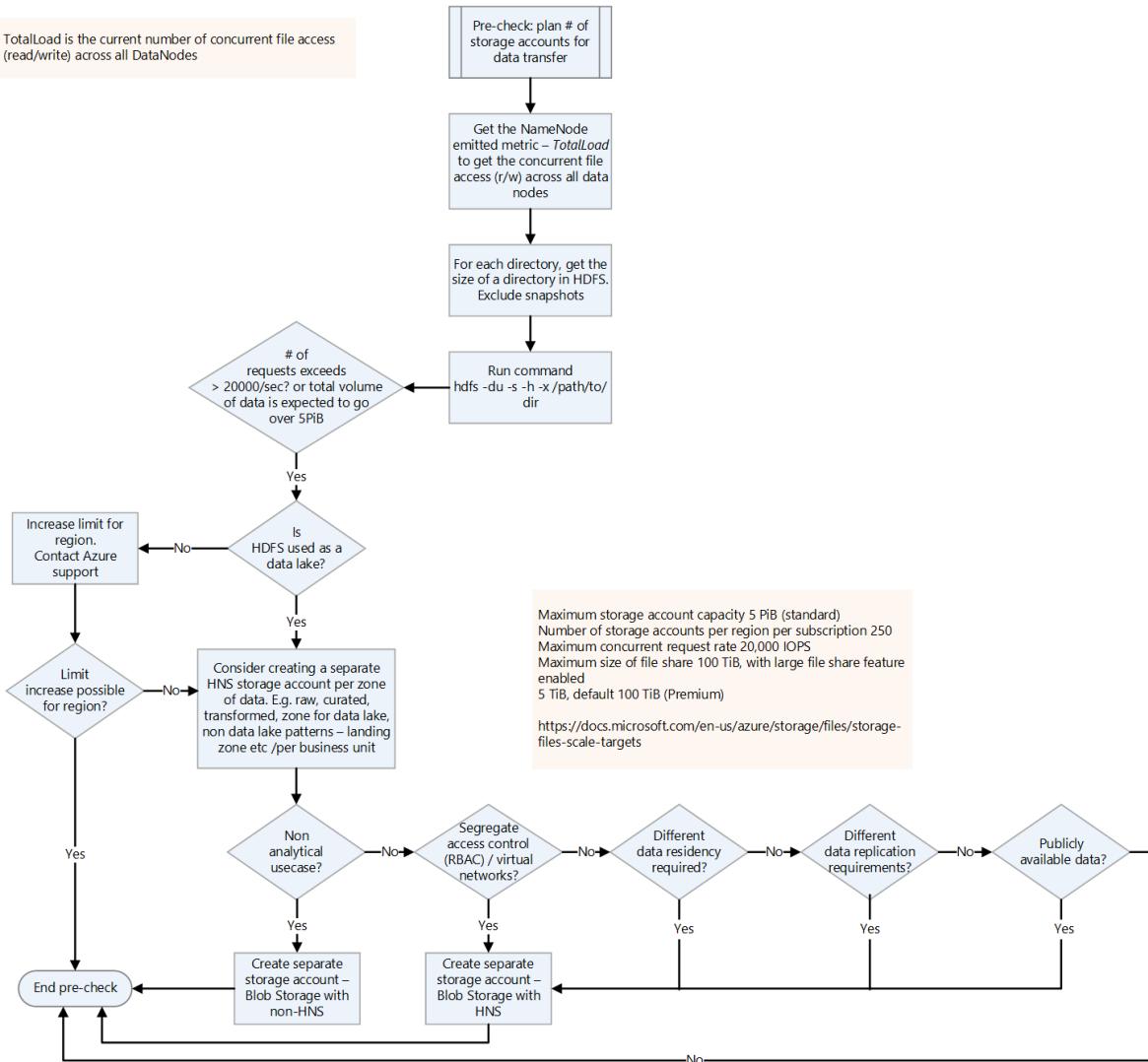
- c. Database ingestion
- d. Database dump
- e. Change data capture
- f. Streaming ingestion

## **2. Plan the number of storage accounts required.**

To plan the number of storage accounts required, understand the total load on the current HDFS. You can use the TotalLoad metric, which is the current number of concurrent file accesses across all DataNodes. Set the limit on the storage account in the region according to the TotalLoad value on-premises and the expected growth on Azure. If it's possible to increase the limit, a single storage account may suffice. However for a data lake, it's best to keep a separate storage account for each zone, to prepare for future data volume growth. Other reasons to keep a separate storage account include:

- Access control
- Resiliency requirements
- Data replication requirements
- Exposing the data for public usage

When you enable a hierarchical namespace on a storage account, you can't change it back to a flat namespace. Workloads such as backups and VM image files don't gain any benefit from a hierarchical namespace.

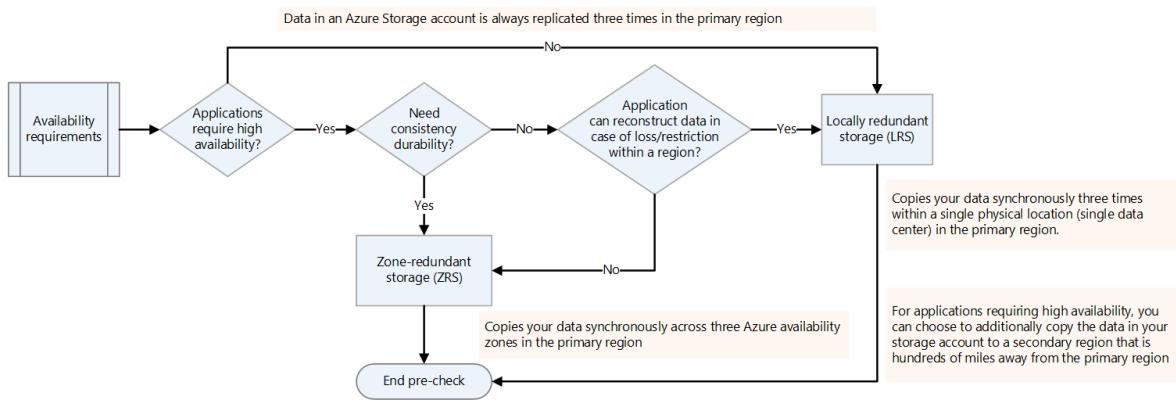


For information about securing the traffic between your virtual network and the storage account over a private link, see [Securing Storage Accounts](#).

For information about default limits for Azure Storage accounts, see [Scalability and performance targets for standard storage accounts](#). The *ingress limit* applies to the data that's sent to a storage account. The *egress limit* applies to the data that's received from a storage account

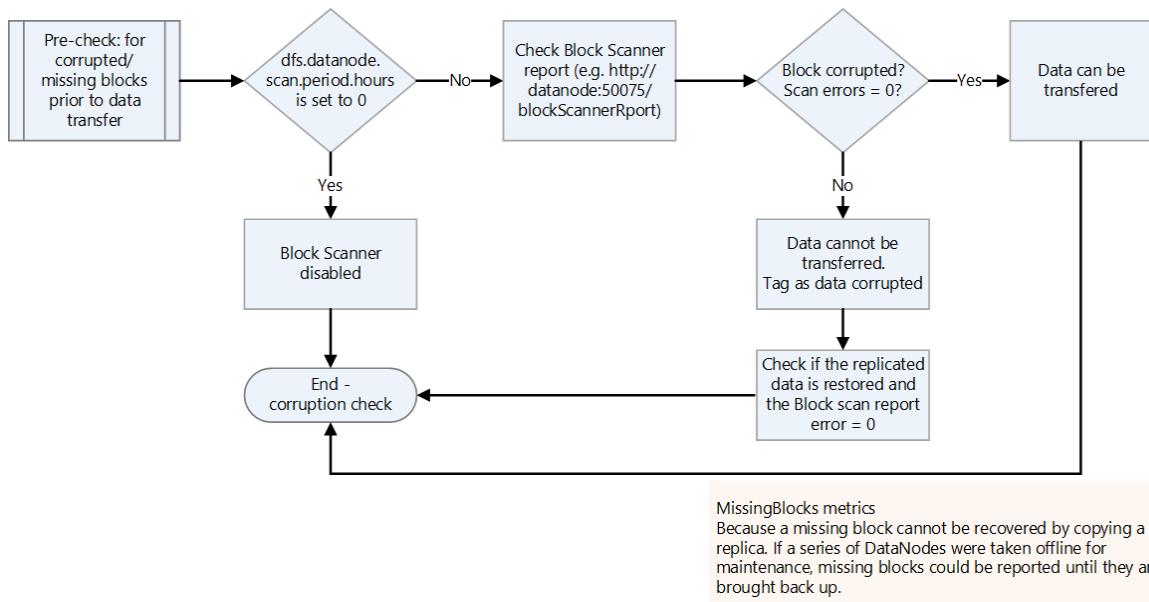
### 3. Decide on availability requirements.

You can specify the replication factor for Hadoop platforms in `hdfs-site.xml` or specify it per file. You can configure replication on Data Lake Storage according to the nature of the data. If an application requires that the data be reconstructed in case of a loss, then zone-redundant storage (ZRS) is an option. In Data Lake Storage ZRS, data is copied synchronously across three availability zones in the primary region. For applications that require high availability and that can run in more than one region, copy the data to a secondary region. This is geo-redundant replication.



#### 4. Check for corrupted or missing blocks.

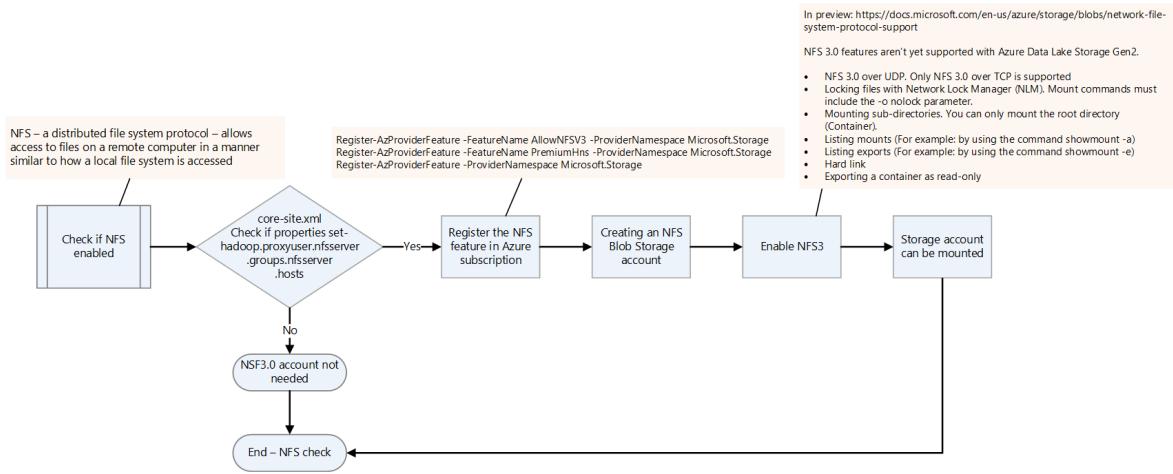
Check the block scanner report for corrupted or missing blocks. If there are any, wait for the file to be restored before transferring it.



#### 5. Check if NFS is enabled.

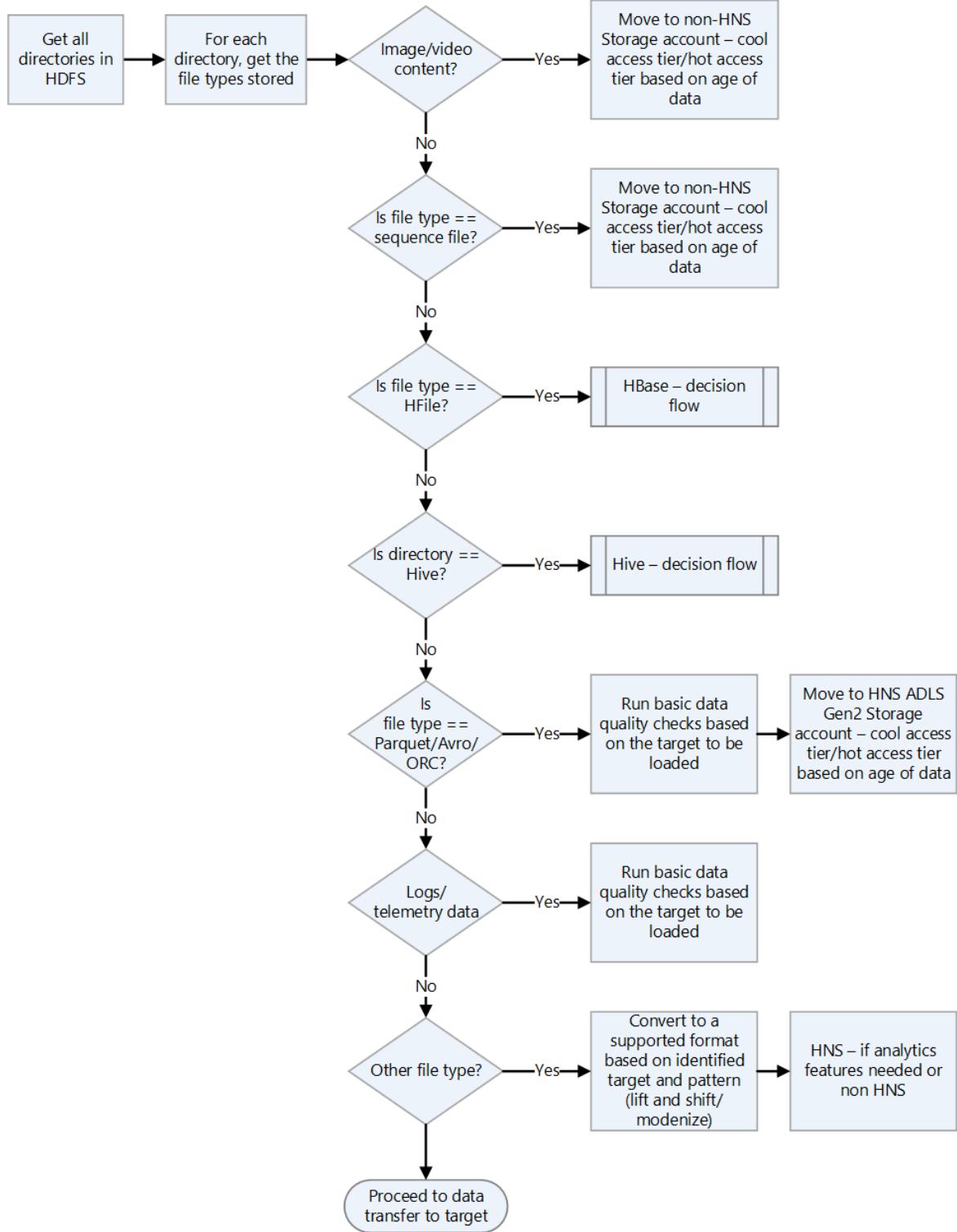
Check if NFS is enabled on the on-premises Hadoop platform by checking the core-site.xml file. It has the nfsserver.groups and nfsserver.hosts properties.

The NFS 3.0 feature is in preview in Data Lake Storage. A few features may not be supported yet. For more information, see [Network File System \(NFS\) 3.0 protocol support for Azure Blob Storage](#).



## 6. Check Hadoop file formats.

Use the following decision flow chart for guidance on how to handle file formats.



## 7. Choose an Azure solution for data transfer.

Data transfer can be online over the network or offline by using physical devices. Which method to use depends on data volume, network bandwidth, and frequency of the data transfer. Historical data has to be transferred only once. Incremental loads require repeated ongoing transfers.

Data transfer methods are discussed in the list that follows. For more information about choosing types of data transfer, see [Choose an Azure solution for data transfer](#).

### a. Azcopy

Azcopy is a command-line utility that can copy files from HDFS to a storage account. This is an option for high-bandwidth transfers (over 1 GBPS).

Here's a sample command to move an HDFS directory:

Bash

```
*azcopy copy "C:\local\path"  
"https://account.blob.core.windows.net/mycontainer1/?sv=2018-03-  
28&ss=bjqt&srt=sco&sp=rwddgcup&se=2019-05-01T05:01:17Z&st=2019-04-  
30T21:01:17Z&spr=https&sig=MGCXiyEzbttkr3ewJIh2AR8KrggSy1DGM9ovN734  
bQF4%3D" --recursive=true*
```

### b. DistCp

[DistCp](#) is a command-line utility in Hadoop that can do distributed copy operations in a Hadoop cluster. DistCp creates several map tasks in the Hadoop cluster to copy data from the source to the sink. This push approach is good when there's adequate network bandwidth, and it doesn't require extra compute resources to be provisioned for data migration. However, if the source HDFS cluster is already running out of capacity and additional compute can't be added, then consider using Data Factory with the DistCp copy activity to pull rather than push the files.

Bash

```
*hadoop distcp -D fs.azure.account.key.<account  
name>.blob.core.windows.net=<Key>  
wasb://<container>@<account>.blob.core.windows.net<path to wasb  
file> hdfs://<hdfs path>*
```

### c. Azure Data Box for large data transfers

Azure Data Box is a physical device that's ordered from Microsoft. It provides large-scale data transfers, and it's an offline data transfer option when network bandwidth is limited and data volume is high (for example, when volume is between a few terabytes and a petabyte).

You connect a Data Box to the LAN to transfer data to it. You then ship it back to the Microsoft data center, where the data is transferred by Microsoft engineers to the configured storage account.

There are multiple Data Box options that differ by the data volumes that they can handle. For more information about the Data Box approach, see [Azure Data Box documentation - Offline transfer](#).

#### d. Data Factory

Data Factory is a data-integration service that helps create data-driven workflows that orchestrate and automate data movement and data transformation. You can use it when there's sufficient network bandwidth available and there's a requirement to orchestrate and monitor data migration. You can use Data Factory for regular incremental loadings of data when the incremental data arrives on the on-premises system as a first hop and can't be directly transferred to the Azure storage account because of security restrictions.

For more information about the various transfer approaches, see [Data transfer for large datasets with moderate to high network bandwidth](#).

For information about using Data Factory to copy data from HDFS, see [Copy data from the HDFS server using Azure Data Factory or Synapse Analytics](#)

#### e. Partner solutions such as WANdisco LiveData migration

The WANdisco LiveData Platform for Azure is one of Microsoft's preferred solutions for migrations from Hadoop to Azure. You access its capabilities by using the Azure portal and the Azure CLI. For more information, see [Migrate your Hadoop data lakes with WANdisco LiveData Platform for Azure](#).

## Contributors

*This article is maintained by Microsoft. It was originally written by the following contributors.*

Principal authors:

- [Namrata Maheshwary](#) | Senior Cloud Solution Architect
- [Raja N](#) | Director, Customer Success
- [Hideo Takagi](#) | Cloud Solution Architect
- [Ram Yerrabotu](#) | Senior Cloud Solution Architect

Other contributors:

- [Ram Baskaran](#) | Senior Cloud Solution Architect
- [Jason Bouska](#) | Senior Software Engineer

- [Eugene Chung](#) | Senior Cloud Solution Architect
- [Pawan Hosattu](#) | Senior Cloud Solution Architect - Engineering
- [Daman Kaur](#) | Cloud Solution Architect
- [Danny Liu](#) | Senior Cloud Solution Architect - Engineering
- [Jose Mendez](#) Senior Cloud Solution Architect
- [Ben Sadeghi](#) | Senior Specialist
- [Sunil Sattiraju](#) | Senior Cloud Solution Architect
- [Amanjeet Singh](#) | Principal Program Manager
- [Nagaraj Seeplapudur Venkatesan](#) | Senior Cloud Solution Architect - Engineering

*To see non-public LinkedIn profiles, sign in to LinkedIn.*

## Next steps

### Azure product introductions

- [Introduction to Azure Data Lake Storage Gen2](#)
- [What is Apache Spark in Azure HDInsight](#)
- [What is Apache Hadoop in Azure HDInsight?](#)
- [What is Apache HBase in Azure HDInsight](#)
- [What is Apache Kafka in Azure HDInsight](#)

### Azure product reference

- [Microsoft Entra documentation](#)
- [Azure Cosmos DB documentation](#)
- [Azure Data Factory documentation](#)
- [Azure Databricks documentation](#)
- [Azure Event Hubs documentation](#)
- [Azure Functions documentation](#)
- [Azure HDInsight documentation](#)
- [Microsoft Purview data governance documentation](#)
- [Azure Stream Analytics documentation](#)
- [Azure Synapse Analytics](#)

## Other

- [Enterprise Security Package for Azure HDInsight](#)
- [Develop Java MapReduce programs for Apache Hadoop on HDInsight](#)
- [Use Apache Sqoop with Hadoop in HDInsight](#)

- Overview of Apache Spark Streaming
- Structured Streaming tutorial
- Use Azure Event Hubs from Apache Kafka applications

## Related resources

- [Hadoop migration to Azure](#)
- [Apache HBase migration to Azure](#)
- [Apache Kafka migration to Azure](#)
- [Apache Sqoop migration to Azure](#)

# Apache HBase migration to Azure

Azure HDInsight

Azure Cosmos DB

Azure Data Lake Storage

Azure Synapse Analytics

Azure Stream Analytics

Apache [HBase](#) is a Java-based, NoSQL column-store, distributed application that's built on top of the Hadoop Distributed File System (HDFS). It's modeled after Google's Bigtable and brings most of the Bigtable capabilities to the Hadoop ecosystem.

HBase is a distributed system. From a [CAP theorem](#) perspective, it's designed for consistency and partitioning. In terms of workload profile, it's designed to serve as a datastore for data-intensive applications that require low-latency and near real-time random reads and writes.

This article discusses components and principles that play a role in planning and building an HBase cluster on Azure for a migration. The discussion is especially pertinent for these migration targets:

- Apache HBase in Azure HDInsight
- Azure IaaS by doing a lift and shift to virtual machines (VMs)
- Azure Cosmos DB.

*Apache, Apache Spark, Apache Hadoop, Apache HBase, Apache Ranger, Apache ZooKeeper, Apache Sqoop, Apache Kafka, and the flame logo are either registered trademarks or trademarks of the Apache Software Foundation in the United States and/or other countries. No endorsement by The Apache Software Foundation is implied by the use of these marks.*

## HBase components, core concepts, and architecture

HBase follows a leader/follower model. This section discusses the components and nodes that are in an HBase deployment.

## Components

The main HBase components are the Master server, the ZooKeeper nodes, and the RegionServers.

## Master server

The [Master server](#) is responsible for all metadata operations for an HBase cluster. This includes creation and deletion of objects, monitoring of RegionServers, and other operations. There are usually two Master servers deployed for high availability.

## ZooKeeper nodes

[ZooKeeper](#) is a centralized service for maintaining configuration information, naming, providing distributed synchronization, and providing group services. These capabilities are needed for coordination in a distributed application environment such as HBase.

## RegionServers

[RegionServers](#) are responsible for serving and managing regions or partitions. A RegionServer does most of the processing of a client read or write request. In a distributed deployment of HBase, a RegionServer runs on an HDFS DataNode.

## Core concepts

It's important that you understand the core concepts of the HBase architecture and data model so that you can optimize an HBase deployment.

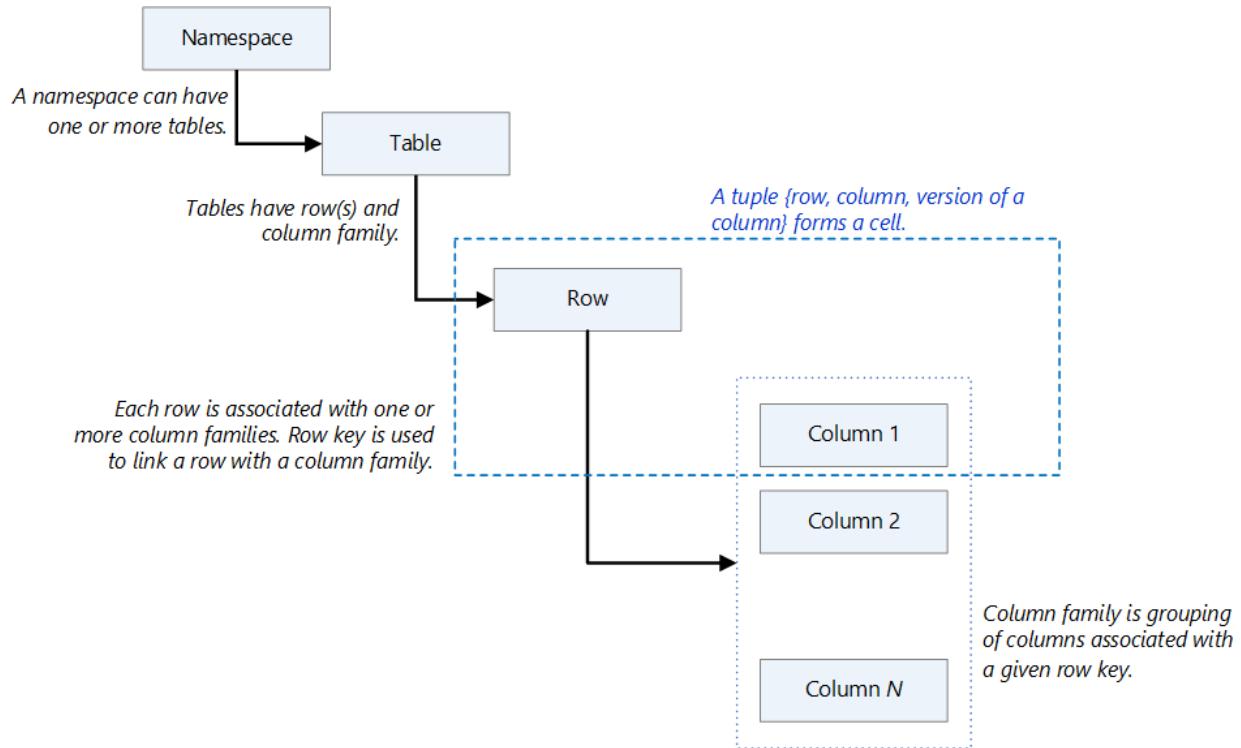
- [Data model](#)
- [Write path](#)
- [Read path](#)
- [Off-heap read and write paths](#)

## Data model

- **Namespace:** A logical grouping of tables, like a database in a relational database system. Namespaces make possible several features that are related to multi-tenancy.
- **Table:** A grouping or collection of rows. Tables are stored in regions or partitions that are spread across RegionServers.
- **Row:** A row consists of a row key and a grouping of columns that's called a *column family*. Rows are sorted and stored based on the row key.
- **Column family:** A grouping of columns that have the same prefix.
- **Cell:** A data location that's uniquely identified by a {row, column, version} tuple.
- **Data model operations:** There are four primary data model operations:
  - **Get** returns attributes for a specified row.

- **Put** either adds new rows to a table or updates existing rows.
- **Scan** allows iteration over multiple rows for specified attributes.
- **Delete** removes a row from a table. A marker, called a *tombstone*, is placed on records to mark them for deletion. The tombstones and deleted rows are removed during major compactions.

The following diagram illustrates these concepts.

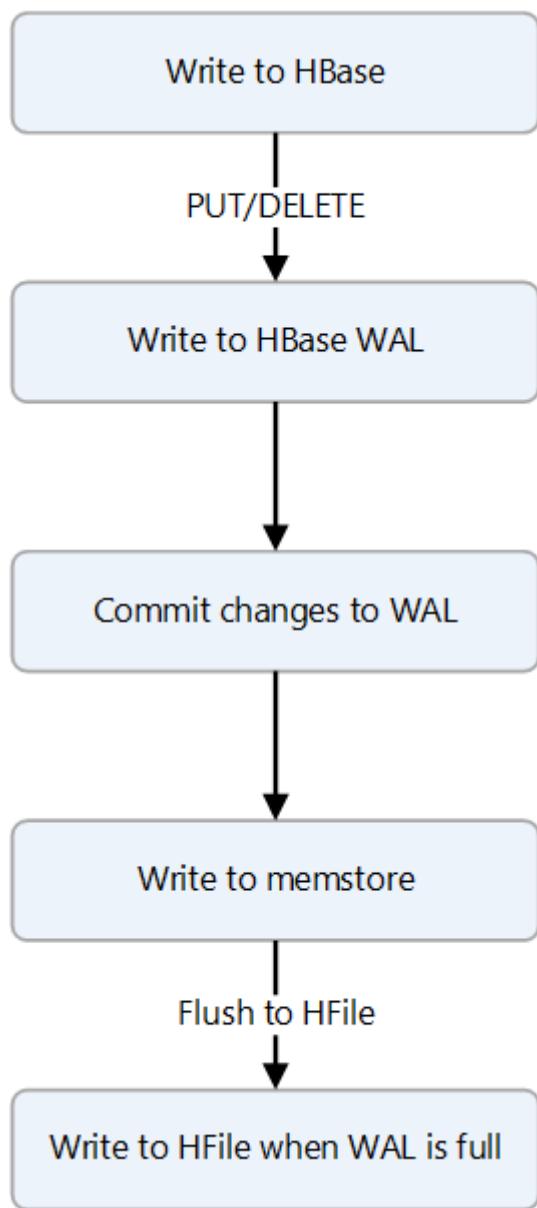


## Write path

HBase uses a combination of data structures that reside in memory and in persistent storage to deliver fast writes. When a write occurs, the data is first written to a write-ahead log (WAL), which is a data structure that's stored on persistent storage. The role of the WAL is to track changes so that logs can be replayed in case of a server failure. The WAL is only used for resiliency. After data is committed to the WAL, it's written to MemStore, which is an in-memory data structure. At this stage, a write is complete.

For long-term data persistence, HBase uses a data structure called an *HBase file* (HFile). An HFile is stored on HDFS. Depending on MemStore size and the data flush interval, data from MemStore is written to an HFile. For information about the format of an HFile, see [Appendix G: HFile format](#).

The following diagram shows the steps of a write operation.



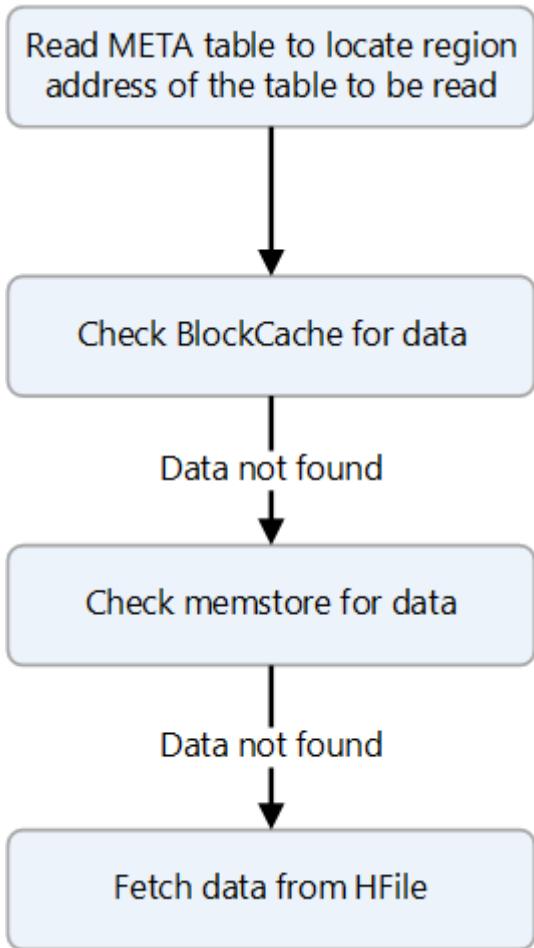
To summarize, the components on the write path are:

- The **WAL** is a data structure that's stored on persistent storage.
- **MemStore** is an in-memory, on-heap data structure.
- **HFile** is an HBase file that's stored on HDFS and used for data persistence.

## Read path

HBase uses several data structures to deliver fast random and sequential reads. HBase tries to fulfill read requests from data that's cached in BlockCache and, failing that, from MemStore. Both are stored on-heap. If the data isn't available from cache, it's fetched from an HFile and cached in BlockCache.

The following diagram shows the steps of a read operation.



### ① Note

For situations that require low read latency, there's an option to cache data in BucketCache, which is an in-memory data structure and is usually off-heap. Data that's stored in BucketCache needn't be stored in BlockCache, so heap activity is reduced and reads are faster. For more information, see [BucketCache ↗](#).

## Off-heap read and write paths

To reduce read and write latencies, HBase versions 2.x and later have a pool of off-heap buffers that are used for writing and reading. The workflow for writing and reading data does its best to avoid on-heap memory allocations in order to reduce the amount of work that garbage collection must do to complete reads and writes. The buffers must be optimized, because their use depends greatly on factors such as the number of regions and RegionServers, memory size, and premium storage that's attached to the HBase cluster on Azure. These parameters are not necessarily the same on the migrated system as on the source system.

## Challenges of running HBase on-premises

These common challenges of on-premises HBase deployments can be reasons for wanting, or needing, to migrate HBase to the cloud:

- Achieving scalability, which can be difficult depending on hardware and data center capacity.
- Having to replace hardware or migrate applications because of end of support for aging infrastructure.
- Achieving high availability and disaster recovery, for reasons such as:
  - Lack of data-center sites
  - Failure of the HBase cluster when the Master, a single point of failure, fails.
- Achieving high productivity, because an on-premises Hadoop ecosystem is complex, hard to manage, and prone to failures.
- The lack of native tools for:
  - Cost transparency
  - Monitoring
  - DevOps
  - Automation

## Considerations for an HBase migration

- If HBase infrastructure as a service (IaaS) migration is your first workload on Azure, we strongly recommend that you invest time and effort needed to build a strong foundation for hosting workloads on Azure. You do this by using Cloud Adoption Framework enterprise-scale landing zone guidance. Enterprise-scale is an architectural approach and a reference implementation that makes it possible to effectively construct and operationalize landing zones on Azure, at scale. For more information, see [What is an Azure landing zone?](#).
- We strongly recommend that all workloads that you run on Azure are designed and deployed according to the [Well-Architected Framework](#), which is a set of guiding tenets that you can use to improve the quality of a workload. The framework consists of five pillars of architecture excellence: Cost Optimization, Operational Excellence, Performance Efficiency, Reliability, and Security.
- When you design and choose Azure compute and storage, consider service limits. Compute and storage have limits that can affect the sizing of infrastructure for a data-intensive application such as HBase. For more information, see [Azure subscription and service limits, quotas, and constraints](#).
- A subscription should be used as a unit of scale. You add more instances of a service to scale out as required. Taking from Cloud Adoption Framework enterprise-scale design principles, use the subscription as a unit of management and scale. Align subscriptions to business needs and priorities, and support

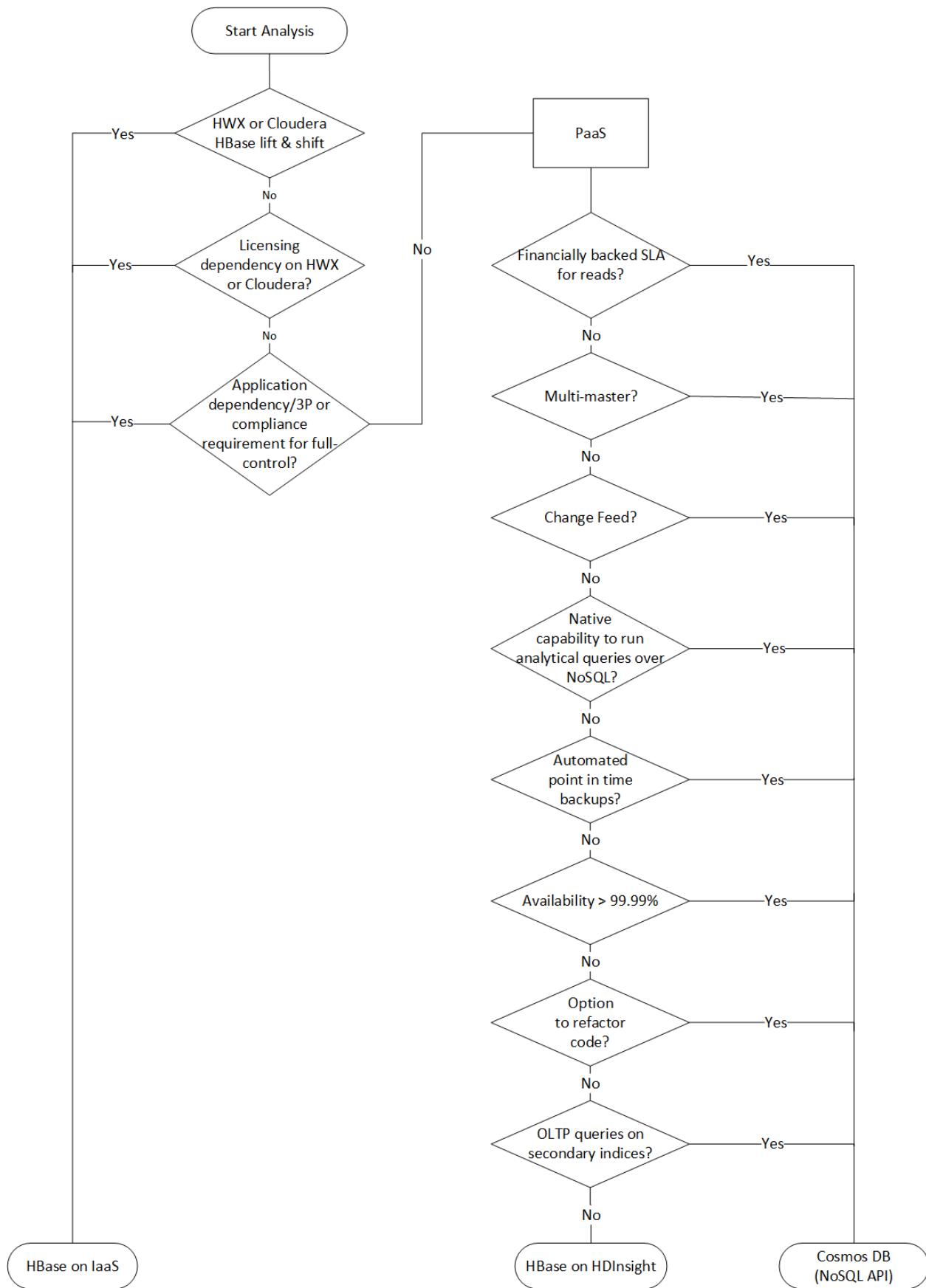
business areas and portfolio owners to encourage migrating current applications and developing new ones.

- An HBase deployment on Azure can use various types of storage for caching and persistent storage. Evaluate the options when you deploy HBase solutions on Azure.
- After you migrate to Azure IaaS, you need to optimize performance and right-size the infrastructure. There are many factors that affect performance, including the size of the infrastructure, the storage types, and the distribution of regions. Even if you minimize application changes when you migrate, an Azure environment is fundamentally different from an on-premises one. There are Azure features and limits to consider in order to meet performance requirements.

## Migration approaches

Azure has several landing targets for Apache HBase. Depending on requirements and product features, you can choose between Azure IaaS (lift and shift to VMs), HBase in HDInsight, and Azure Cosmos DB (NoSQL API).

Here's a decision flowchart for selecting a target environment:



These targets are discussed in the following sections:

- Migrate to Azure IaaS
- Migrate to HBase in HDInsight
- Migrate to Azure Cosmos DB (NoSQL API)

# Migrate to Azure IaaS

Migrating HBase to Azure IaaS requires the following activities:

- Assess the existing deployment and decide on requirements
- Consider VM options
- Consider storage options
- Migrate data
- Establish security
- Monitor the HBase deployment

## Assess the existing deployment and decide on requirements

The following table provides guidance on assessing the existing deployment of HBase and on establishing a set of requirements for an HBase migration to Azure.

 Expand table

Layer	Characteristic	Background
Infrastructure	Number of servers for each type of role: HBase Master, RegionServer, ZooKeeper node	Understand the scale and design of the existing solution.
	Number of cores per server	Use the <code>lscpu</code> or the <code>cat /proc/cpuinfo</code> command to list cores per server.
	Memory per server	On Linux, use <code>free -m</code> or <code>cat /proc/meminfo</code> to report on memory on each server.
	Is the existing environment virtualized, or deployed on bare-metal servers?	This information is used for sizing and for understanding the performance characteristics of the on-premises HBase environment.
	Network configuration	Understand the network bandwidth that each VM can support, and whether a special NIC configuration is used to support high

Layer	Characteristic	Background
		<p>bandwidth between HBase servers.</p> <p>Use the following commands to extract details of VM network configuration: <code>ifconfig -a</code> or <code>ethtool &lt;name of the interface&gt;</code>.</p>
	Storage configuration	<p>What is the total size of the data including replicas? Usually, the default configuration of HDFS replicates data three times. The HDFS CLI can be used to extract the total size of data persisted by HBase:</p> <pre>hdfs dfs -du -h hdfs://&lt;data node address&gt;/HBase</pre> <p>Also, establish storage performance targets (IOPS and throughput). This information is used to provision storage and to understand the level of throughput required to support the Azure deployment.</p>
Operating system	Version and distribution type	<p>The following command prints out details of the Linux distribution and version:</p> <pre>uname -a</pre>
	Kernel parameters	See the note that follows this table.
Application	The versions of HBase and Hadoop distributions, such as Hortonworks and Cloudera, that are in use	<p>The distribution is usually one of the following: HortonWorks, Cloudera, MapR, or an open-source version of Hadoop and HBase.</p> <p>To find out the versions of HBase and Hadoop, use the following commands: <code>hbase version</code> and <code>hdfs version</code>.</p>
	HBase-specific information: the number of tables, the metadata for each table (regions, column-family)	<p>You can extract information related to HBase deployment by using the Ambari UI. If that's not available, you can use CLI:</p> <pre>scan 'hbase:meta'{FILTER=&gt;"PrefixFilter('tableName')", COLUMNS=&gt;['info:regioninfo']}{'list_regions &lt;table name&gt;'}</pre> <p>To list all the regions associated with a given table, use:</p> <pre>list_regions &lt;table name&gt;</pre>
	JAVA (JDK) version	<code>java -version</code>

Layer	Characteristic	Background
	HBase garbage collection configuration	What garbage collection strategy is used? The most common are concurrent mark sweep (CMS) and garbage first (G1). The strategy is specified in the <code>hbase-env.sh</code> configuration file. Recent research shows G1 to be more efficient for large heap sizes.
Security & administration	The ways that HBase is accessed	<p>How do users access the data in HBase? Is it by using APIs or directly by using the HBase shell?</p> <p>How do applications consume data?</p> <p>How is data written to HBase and how far away is HBase? Is it in the same data center?</p>
	User provisioning	<p>How are users authenticated and authorized? The possibilities include:</p> <ul style="list-style-type: none"> <li>• Ranger</li> <li>• Knox</li> <li>• Kerberos</li> </ul>
	Encryption	Is there a requirement to encrypt data? In transport? At rest? What encryption solutions are used?
	Tokenization	Is there a requirement to tokenize data? If so, how? Popular tokenization applications include Protegy and Vormetric.
	Compliance	Are there any special regulatory requirements applicable to HBase workloads, such as CI-DSS or HIPAA?
	Management policies for keys, certificates, and secrets	What policies and tools are used to manage secrets?
High availability and disaster recovery	What is the service level agreement (SLA) and what are the recovery point objectives (RPO) and recovery time objectives (RTO) of	The answers affect the design of the target deployment on Azure. For instance, should there be a hot standby or an active-active regional deployment?

Layer	Characteristic	Background
	the source HBase deployment?	
	What are the business continuity (BC) and data recovery (DR) strategies for HBase workloads?	Describe BC and DR strategies and the impact if HBase isn't available.
Data	Data size and growth	How much data is migrated to HBase initially? What is the expected growth after 6, 12, and 18 months? This information is used for capacity planning and for sizing the cluster. Eventually, it's also used to optimize deployment costs.
	Ingestion	How is data written to HBase?
	Consumption	How is HBase data used? Is it by using APIs, or by a compute engine such as HDInsight Spark or Databricks Spark?
	Access pattern	Is traffic on HBase read-heavy or write-heavy? This affects the fine-tuning of the HBase configuration parameters that are defined in the <code>hbase-site.xml</code> and <code>hdfs-site.xml</code> files.

## ① Note

Kernel-level parameters may have been applied to improve HBase performance on the source system. Because the performance characteristics of the migrated system aren't the same, we recommend against changing default parameters to match the source system except to follow recommendations from your operating system vendor or your application vendor. In most cases, it's best to adjust kernel parameters during the performance optimization phase of the migration.

### Linux memory and block device parameters

```
cat /sys/kernel/mm/transparent_hugepage/enabled
cat /sys/kernel/mm/transparent_hugepage/defrag
cat /sys/block/sda/queue/scheduler
```

```

cat /sys/class/block/sda/queue/rotational
cat /sys/class/block/sda/queue/read_ahead_kb
cat /proc/sys/vm/zone_reclaim_mode

Linux network stack parameters
sudo sysctl -a \| grep -i
"net.core.rmem_max\|net.core.wmem_max\|net.core.rmem_default\|net.core.wmem_de-
fault\|net.core.optmem_max\|net.ipv4.tcp_rmem\|net.ipv4.tcp_wmem" |

```

There are several partner solutions that can assist with assessment. [Unravel](#) has solutions that can help you fast-track assessment for data migrations to Azure.

## Consider VM options

[Azure Virtual Machines](#) is one of several types of on-demand, scalable compute resources that Azure offers. Typically, you choose a VM when you need more control over the computing environment than other choices offer.

As you design your migration and select VMs, consider the following:

- The names of your application resources
- The location where the resources are stored
- The size of the VM
- The maximum number of VMs that can be created
- The operating system that the VM runs
- The configuration of the VM after it starts
- The related resources that the VM needs

For more information, see [What do I need to think about before creating a VM?](#).

Azure VM families are optimized to suit different use cases and to provide a balance of compute (virtual cores) and memory.

[\[+\] Expand table](#)

Type	Size	Description
<a href="#">Entry Level</a>	A, Av2	A-series VMs have CPU performance and memory configurations that are best suited for entry level workloads such as development and test. They are a low-cost option for getting started with Azure.

Type	Size	Description
General Purpose	D, DSv2, Dv2	These VMs have balanced CPU-to-memory ratios. They're ideal for testing and development, for small to medium databases, and for web servers that have low to medium traffic.
Compute Optimized	F	These VMs have a high CPU-to-memory ratio. They're good for medium traffic web servers, network appliances, batch processes, and application servers.
Memory Optimized	Esv3, Ev3	These VMs have a high memory-to-CPU ratio. They're good for relational database servers, medium to large caches, and in-memory analytics.

For more information, see [Virtual machines in Azure](#).

HBase is designed to use memory and premium storage (such as SSDs) in order to optimize database performance.

- It ships with features like BucketCache, which can significantly improve read performance. BucketCache is stored off-heap, so we recommend VMs that have higher memory-to-CPU ratios.
- The HBase write path writes changes to a WAL, which is a data structure that's persisted on a storage medium. Storing the WAL on a fast storage medium such as SSDs improves write performance.
- HBase is designed to scale out as performance and storage requirements grow.

For more information about sizing Azure VMs, see [Sizes for virtual machines in Azure](#).

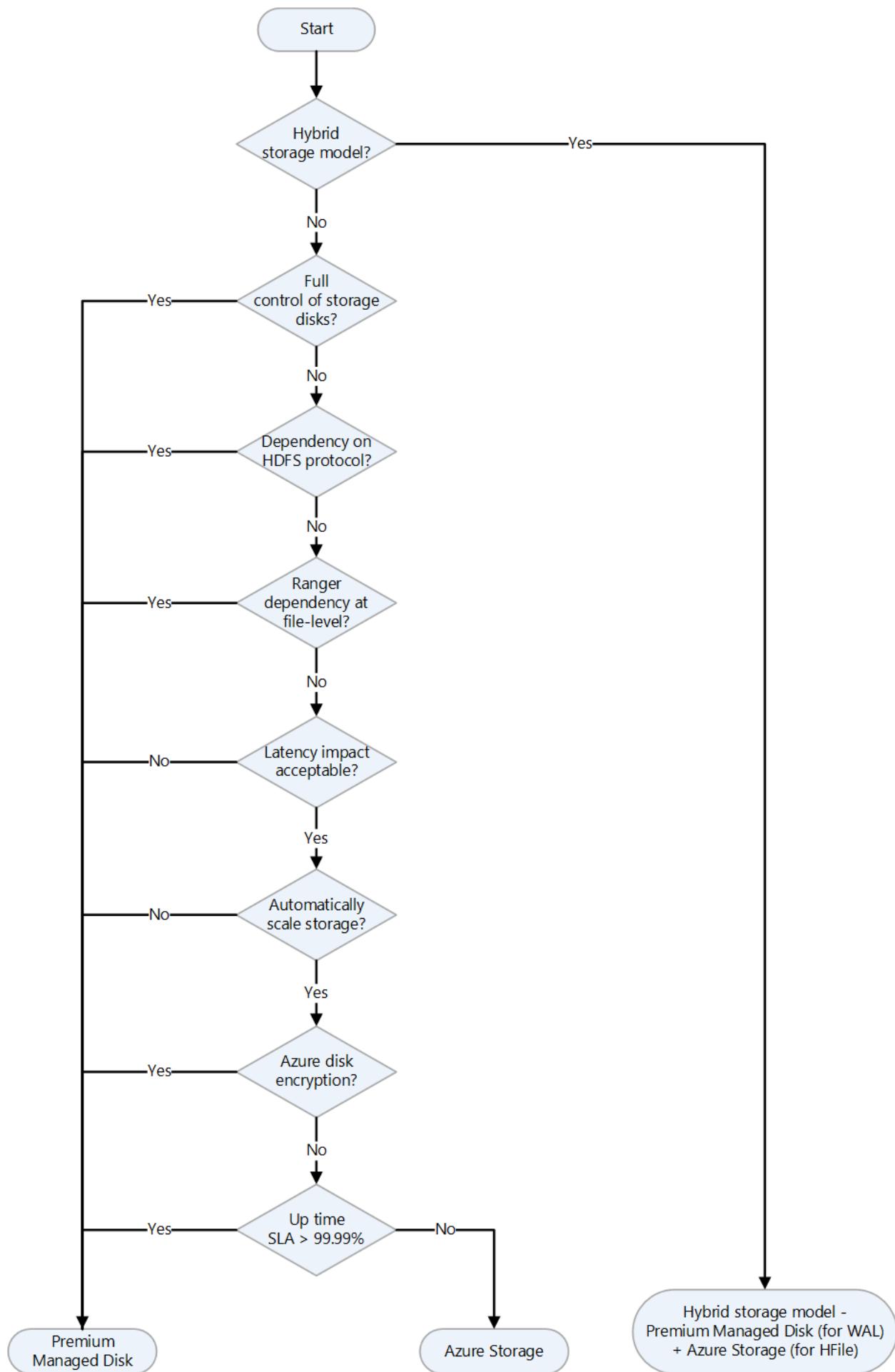
Based on considerations of compute and memory needs, we recommend using the following Azure compute family types for the various HBase node types:

- **HBase Master** – For enterprise deployments, we recommend at least two Masters for high availability. For a large HBase cluster, a DS5\_v2 Azure VM with 16 virtual CPUs (vCPUs) and 56 GiB of memory should suffice for most deployments. For medium-sized clusters, we recommend at least 8 vCPUs and 20 to 30 GiB of memory.
- **HDFS NameNode** – We recommend hosting HDFS NameNodes on different VMs than those that the Masters use. Two NameNodes should be deployed for high availability. For the Masters, we recommend DS5\_v2 VMs for large production-grade clusters.

- **HBase RegionServer** – We recommend using Azure VMs with a high memory-to-CPU ratio. HBase has several features that use memory to improve read and write times. VMs such as DS14\_v2 or DS15\_v2 are good starting points. HBase is designed to scale out by adding RegionServers to improve performance.
- **ZooKeeper** – HBase relies on ZooKeeper for operations. An Azure VM with 4 to 8 vCPUs and 4 to 8 GiB memory is a good starting point. Ensure that there's local storage available. ZooKeeper nodes should be deployed on their own set of VMs.

## Consider storage options

Azure offers several storage options that are suitable for hosting an IaaS deployment of HBase. The following flowchart uses features of various options to select a storage option. Each storage option on Azure has different performance, availability, and cost targets.



See these articles for additional information:

- [Azure Managed Disks](#)
- [Azure Premium SSD](#)
- [Azure Disk Encryption](#)

In a hybrid storage model, we use a mix of local storage and remote storage to strike a balance between performance and cost. The most common pattern is to place the HBase WAL, which is on the write path, to locally attached Azure Premium SSD Managed Disk. The long term data or HFiles are stored on Premium SSD or Standard SSD depending on cost and performance targets.

#### Note

The open source version of Apache Ranger can't apply policies and access control at the file level for Azure Storage (Azure Blob Storage or Azure Data Lake Storage). This capability is supported by the Ranger version that ships with [Cloudera Data Platform \(CDP\)](#).

There are two key factors that influence the sizing of HBase storage: data volume and throughput of reads and writes. These factors also affect the choice of Azure VM size and numbers, and Azure Storage (Managed Disk or Data Lake Storage).

- **Volume of data**

This is the data that must be persisted on HBase. The data is persisted to underlying storage. When we refer to volume of data, for sizing and planning purposes, volume includes the raw data and 3x replication. Total storage size is the metric that we use to drive volume.

- **Throughput of reads and writes**

This is the rate at which that HBase writes and reads data. IOPS and I/O size are the two metrics that drive this.

If you're planning a greenfield deployment of HBase on Azure IaaS and there's no reference point in terms of existing deployments, our recommendation is to go with the following sizing and then add RegionServers as the volume of data or the need for higher throughput grows. Ds-series and Es-series VMs are well suited for an HBase RegionServer. For HBase Master and ZooKeeper nodes, we recommend using smaller Ds-series VMs.

For better sizing accuracy and for establishing a performance baseline, we recommend that you run tests on Azure IaaS by using the HBase dataset, model, and workload pattern. If moving the data isn't possible, we recommend using benchmarking tools

such as the Yahoo! Cloud Serving Benchmark (YCSB) to generate synthetic data and simulate sequential and random I/O traffic. The intent of this exercise is to gain an understanding of the level of performance to expect by using a combination of Azure compute and Premium SSD. The tests should include day-to-day workload patterns and special cases such as workloads that can cause a spike in resource usage, like month-end and year-end activities. For example, a retailer that uses HBase observes spikes in resource usage during holiday periods, whereas a provider of financial services observes spikes during key financial periods.

Inputs from assessment activities and performance baselines should provide a fairly accurate view of sizing on Azure IaaS. Due to the nature of the workloads, there's room to optimize operations by scaling-out or scaling-in clusters after going live. We recommend that you familiarize yourself with various [cost optimization](#) techniques for optimizing costs and operations.

## Migrate data

### Note

We recommend against directly copying HFiles to migrate data files from one HBase deployment to another. Instead, use one of the out-of-the-box HBase features.

The following table shows data migration approaches for various situations.

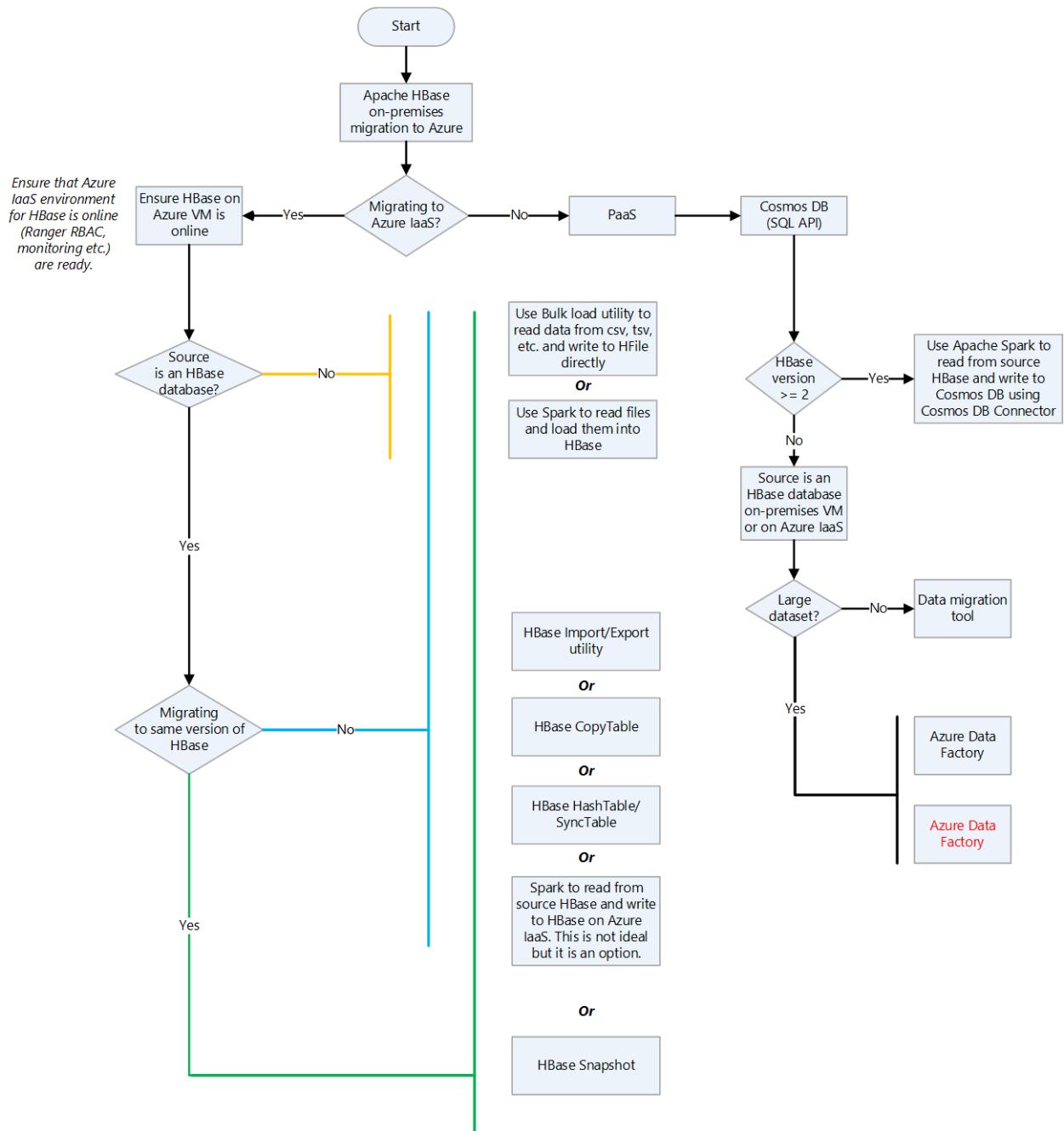
 [Expand table](#)

Pattern	Migration approach	Considerations
<b>Bulk load scenarios where source data isn't being read from an HBase instance.</b>  For example, the source data is in a file format such as CSV, TSV, or Parquet, or it's	Build a pipeline using tools such as WANDisco and Databricks to read the source data and write it to HBase.  If the data is sitting on a file system or on HDFS, then use tools such as WANDisco, HDInsight Spark, or Databricks Spark to read from the source and write to HBase on Azure.  At a high-level, migration	Need for separate infrastructure for the migration tool runtime.  Handling encryption and tokenization requirements during data migration.  Network latency between source and target (Azure HBase).

Pattern	Migration approach	Considerations
in a database or proprietary format.	<p>pipelines can be built—one per target table on HBase—that extract data from the source and write it first to Azure HDFS. Then a separate pipeline can be built to read from Azure HDFS and write to Azure HBase.</p>	
<p><b>The source is an HBase instance but it's not the same HBase version as that of the Azure target HBase.</b></p>	<p>Since the source is also an HBase datastore, consider direct HBase cluster-to-cluster data migration options such as:</p> <ul style="list-style-type: none"> <li>• HBase CopyTable</li> <li>• Spark on HDInsight or Databricks Spark</li> <li>• HBase Export Utility and HBase Import Utility</li> <li>• The HashTable/SyncTable tool</li> </ul> <p><i>Note</i> - CopyTable supports full and delta table copy features.</p>	<p>The same considerations as for bulk loads, plus a few related to specific migrations tools.</p> <p>For HBase CopyTable, consider the HBase versions on the source and target deployments.</p> <p>Clusters must be online for both the source and target.</p> <p>Additional resources are needed on the source side to support additional read traffic on the source HBase instance.</p> <p>The CopyTable feature, by default, only copies the latest version of a row cell. It also copies all cells within a specified time range. There might be changes on the source HBase while CopyTable runs. When this happens, the changes are either completely copied or completely ignored.</p> <p>Spark on HDInsight and Databricks Spark require additional resources or a separate cluster for migrating data, but it's a tried and tested approach.</p> <p>The HBase Export Utility, by default, always copies the latest version of a cell to the HBase.</p>

Pattern	Migration approach	Considerations
		<p>target.</p> <p>HashTable/SyncTable is more efficient than the CopyTable feature.</p>
<b>The source is an HBase database with the same HBase version as that of the HBase target.</b>	<p>The same options that are used when the versions differ</p> <p>HBase Snapshots</p>	<p>The considerations that were already mentioned for the case where the versions differ.</p> <p>For HBase Snapshots, the considerations are as follows.</p> <ul style="list-style-type: none"> <li>• Snapshots doesn't create a copy of the data, but it does create a reference back to HFiles. The referenced HFiles are archived separately in case compaction is triggered on the parent table that's referenced in a snapshot.</li> <li>• The footprint on the source and target HBases when a snapshot restore is triggered.</li> <li>• Keeping the data source and target HBases in-sync during migration and then planning for the final cutover.</li> <li>• Network latency between the source and target.</li> </ul>

Here's a decision flowchart to aid you in choosing data migration techniques when you migrate HBase to Azure:



Further reading:

- [Bulk Loading \(Apache HBase Reference Guide\)](#) ↗
- [Import \(Apache HBase Reference Guide\)](#) ↗
- [CopyTable \(Apache HBase Reference Guide\)](#) ↗
- [HashTable/SyncTable \(Apache HBase Reference Guide\)](#) ↗
- [HBase Snapshots \(Apache HBase Reference Guide\)](#) ↗
- [Core \(SQL\) API](#)
- [Tutorial: Use data migration tool to migrate your data to Azure Cosmos DB](#)
- [Copy data from HBase using Azure Data Factory or Synapse Analytics](#)

## Establish security

For an HBase cluster to operate, it must be able to communicate with other VMs in the cluster. This includes VMs that host Master, RegionServers, and ZooKeeper.

There are various ways to make it possible for servers to authenticate to each other seamlessly. The most common patterns are:

- Kerberized Linux servers that are domain-joined to a Windows domain controller
- A Kerberized Linux environment that uses Microsoft Entra Domain Services (Microsoft Entra Domain Services)
- A standalone MIT Kerberos domain controller
- Authorization by using Apache Ranger

## **Kerberized Linux servers domain-joined to a Windows domain controller**

Linux servers hosting Apache Hadoop are domain-joined to an Active Directory domain. In this setup, we see that there's no need to have a separately hosted Kerberos, as this capability sits within a Windows domain controller.

Considerations:

- Location of the domain controller.
- Roles assigned to the domain controller.

If the domain controller is located on-premises or outside of an Azure region or in a non-Azure cloud, latency must be considered for operations that require interaction with the domain controller. One option is to host a second [domain controller on Azure](#). The Azure-based domain controller is then used for all authentication and authorization scenarios for workloads that run on Azure. We recommend against assigning [operations masters roles](#) to the domain controllers that are deployed in Azure. In such a scenario, the primary domain controller is hosted on-premises.

## **A Kerberized Linux environment that uses Microsoft Entra Domain Services (Microsoft Entra Domain Services)**

[Microsoft Entra Domain Services](#) provides managed domain services such as domain join, group policy, lightweight directory access protocol (LDAP), and Kerberos and NTLM authentication. You can use Microsoft Entra Domain Services without the need to deploy, manage, and patch domain controllers in the cloud.

Considerations:

- Regional availability of Microsoft Entra Domain Services

- Networking requirements for Microsoft Entra Domain Services
- High availability, disaster recovery, and uptime SLA for Microsoft Entra Domain Services

## A standalone MIT Kerberos domain controller

There are some deployments of Hadoop that use a standalone Kerberos domain controller—such as the MIT Kerberos domain controller—that's deployed on a separate set of Azure VMs for high availability. For information about deployment on Linux servers, see [Installing KDCs](#).

Considerations for deployment of an MIT Kerberos domain controller:

- Managing the controller
- High availability and disaster recovery
- Well-Architected Framework guidelines

## Authorization by using Apache Ranger

[Apache Ranger](#) provides comprehensive security across the Apache Hadoop ecosystem. In the context of Apache HBase, Ranger is used to build and deploy policy-based authorization.

## Monitor the HBase deployment

For a lift and shift migration to Azure IaaS, you can use the same monitoring techniques that you used on the source system. For other migrations, there are several options available for monitoring a full HBase stack on Azure IaaS:

- [Apache Ambari](#) for monitoring the Hadoop and HBase stack
- [Java Management Extensions \(JMX\)](#) monitoring and [Azure Monitor](#)
- Infrastructure (VM, storage disks, and networking) logging and metrics

## Apache Ambari for monitoring the Hadoop and HBase stack

Apache Ambari is a project for managing distributed applications such as Hadoop and HBase. It uses the Ambari Metrics System to provide metrics in Ambari-managed clusters. Ambari can report on metrics that are specific to HBase, such as cluster load, network usage, memory usage, and HBase Master heap. Ambari-managed clusters ship with dashboards for monitoring clusters.

For more information, see [Apache Ambari](#). For detailed guidance on deploying Ambari-managed Hadoop and HBase clusters, see [3. Install Options](#).

## Java Management Extensions (JMX) monitoring and Azure Monitor

HBase and Hadoop processes run in a Java virtual machine (JVM). A JVM has built-in instrumentation to provide monitoring data via the Java Management Extensions (JMX) API. You can instrument applications to provide data via the JMX API.

In addition to exporting HBase metrics to the standard output options that the Hadoop metrics package supports, you can also export via the JMX API. This makes it possible to view HBase stats in JConsole and other JMX clients.

You can use a Log Analytics agent to capture custom JSON data sources and store the output in Log Analytics for reporting by Azure Monitor:

- After Linux servers have been deployed, can install and configure the Log Analytics agent for Linux. For more information, see [Monitor virtual machines with Azure Monitor](#).
- Configure the Log Analytics agent to [collect custom JSON data](#). Many JMX endpoints provide JSON that can be collected and parsed using various FluentD plugins.
- Here's a sample of how to configure input and output plugins to collect metrics for writing to HBase RegionServer WALs.

XML

```
<source>
  type exec
  command 'curl -XGET http://<regionServerName>:16030/jmx?
qry=Hadoop:service=hbase,name=RegionServer,sub=WAL'
  format json
  tag oms.api.metrics_regionserver
  run_interval 1m
</source>

<filter oms.api.metrics_regionserver>
  type filter_flatten
  select record['beans'][0]
</filter>

<match oms.api.metrics*>
  type out_oms_api
  log_level info

  buffer_chunk_limit 5m
```

```

buffer_type file
buffer_path /var/opt/microsoft/omsagent/state/out_oms_api*.buffer
buffer_queue_limit 10
flush_interval 20s
retry_limit 10
retry_wait 5s
max_retry_wait 5m

compress true
</match>

```

By using the example above, you can create input and output plugins for the list below. The list contains JMX endpoints that you can query to extract metrics for HBase Master and RegionServer.

XML

```

curl -XGET http://<region_server>:16030/jmx?
qry=Hadoop:service=hbase,name=RegionServer,sub=Server
curl -XGET http://<region_server>:16030/jmx?
qry=Hadoop:service=hbase,name=RegionServer,sub=Replication
curl -XGET http://<rest_server>:8085/jmx?qry=Hadoop:service=hbase,name=REST
curl -XGET http://<rest_server>:8085/jmx?
qry=Hadoop:service=hbase,name=JvmMetrics
curl -XGET http://<region_server>:16030/jmx?
qry=Hadoop:service=hbase,name=RegionServer,sub=WAL
curl -XGET http://<region_server>:16030/jmx?
qry=Hadoop:service=hbase,name=RegionServer,sub=IPC
curl -XGET http://<region_server>:16030/jmx?
qry=Hadoop:service=hbase,name=JvmMetrics
curl -XGET http://<region_server>:16030/jmx?
qry=java.lang:type=OperatingSystem
curl -XGET http://<HBase_master>:16010/jmx?
qry=Hadoop:service=hbase,name=Master,sub=AssignmentManger
curl -XGET http://<HBase_master>:16010/jmx?
qry=Hadoop:service=hbase,name=Master,sub=IPC
curl -XGET http://<HBase_master>:16010/jmx?
qry=java.lang:type=OperatingSystem
curl -XGET http://<HBase_master>:16010/jmx?
qry=Hadoop:service=hbase,name=Master,sub=Balancer
curl -XGET http://<HBase_master>:16010/jmx?
qry=Hadoop:service=hbase,name=JvmMetrics
curl -XGET http://<HBase_master>:16010/jmx?
qry=Hadoop:service=hbase,name=Master,sub=Server
curl -XGET http://<HBase_master>:16010/jmx?
qry=Hadoop:service=hbase,name=Master,sub=FileSystem

```

After it's configured, a source appears under the Custom Logs blade. In the snippet above, we use the name oms.api.metrics\_regionservers for the input. Log Analytics uses the following format for displaying the custom table name with a suffix\_CL.

Name	Type
log_regionserver_CL	Ingestion API
metrics_azure_iaas_regionserver_CL	Ingestion API
metrics_azure_iaas_regionserver_ne...	Ingestion API
<b>metrics_regionserver_CL</b>	<b>Ingestion API</b>
metrics_regionserver_ipc_CL	Ingestion API
metrics_regionserver_jvm_CL	Ingestion API
metrics_regionserver_os_CL	Ingestion API
metrics_regionserver_replication_CL	Ingestion API
metrics_regionserver_wal_CL	Ingestion API

## Infrastructure (VM, storage disks and networking) logging and metrics

### ① Note

For an HBase migration from a non-Azure cloud deployment that uses a native monitoring solution, we recommend using the native Azure monitoring solution in the new environment. Application and infrastructure monitoring together provide a complete picture.

Linux distributions ship with several tools, such as sar, for capturing and reporting on metrics. Although they're good for monitoring the health of an individual VM, you can't rely on them for a large enterprise-grade deployment of Apache HBase. We recommend that you use Azure Monitor instead. It provides dashboards for monitoring all the VMs.

Azure Monitor relies on [Log Analytics agents](#). There should be an agent on every VM. The agent captures the data that's written to Syslog and the performance data from

individual VMs. It sends the data to Azure Log Analytics for storage and indexing. Azure Monitor dashboards then pull data from a configured Log Analytics Workspace and present administrators a view of overall health of all the VMs. This is a native option that can be enabled seamlessly for Linux-based Azure VMs.

For instructions on setting up Azure Monitor to collect data from Linux, see [Monitor virtual machines with Azure Monitor](#). After data has been written to Log Analytics, you can use Kusto to analyze it. For more information, see [Log Analytics tutorial](#).

## Migrate to HBase in HDInsight

You can download a detailed guide to migrating HBase to an HDInsight HBase cluster. The download page is [Guide to Migrating Big Data Workloads to Azure HDInsight](#).

## Migrate to Azure Cosmos DB (NoSQL API)

The guide to migrating to Azure Cosmos NoSQL API is [Migrate data from Apache HBase to Azure Cosmos DB NoSQL API account](#).

## Contributors

*This article is maintained by Microsoft. It was originally written by the following contributors.*

Principal authors:

- [Namrata Maheshwary](#) | Senior Cloud Solution Architect
- [Raja N](#) | Director, Customer Success
- [Hideo Takagi](#) | Cloud Solution Architect
- [Ram Yerrabotu](#) | Senior Cloud Solution Architect

Other contributors:

- [Ram Baskaran](#) | Senior Cloud Solution Architect
- [Jason Bouska](#) | Senior Software Engineer
- [Eugene Chung](#) | Senior Cloud Solution Architect
- [Pawan Hosatti](#) | Senior Cloud Solution Architect - Engineering
- [Daman Kaur](#) | Cloud Solution Architect
- [Danny Liu](#) | Senior Cloud Solution Architect - Engineering
- [Jose Mendez](#) | Senior Cloud Solution Architect
- [Ben Sadeghi](#) | Senior Specialist
- [Sunil Sattiraju](#) | Senior Cloud Solution Architect

- [Amanjeet Singh](#) | Principal Program Manager
- [Nagaraj Seeplapudur Venkatesan](#) | Senior Cloud Solution Architect - Engineering

*To see non-public LinkedIn profiles, sign in to LinkedIn.*

## Next steps

### Azure product introductions

- [Introduction to Azure Data Lake Storage Gen2](#)
- [What is Apache Spark in Azure HDInsight](#)
- [What is Apache Hadoop in Azure HDInsight?](#)
- [What is Apache HBase in Azure HDInsight](#)
- [What is Apache Kafka in Azure HDInsight](#)

### Azure product reference

- [Microsoft Entra documentation](#)
- [Azure Cosmos DB documentation](#)
- [Azure Data Factory documentation](#)
- [Azure Databricks documentation](#)
- [Azure Event Hubs documentation](#)
- [Azure Functions documentation](#)
- [Azure HDInsight documentation](#)
- [Microsoft Purview data governance documentation](#)
- [Azure Stream Analytics documentation](#)
- [Azure Synapse Analytics](#)

## Other

- [Enterprise Security Package for Azure HDInsight](#)
- [Develop Java MapReduce programs for Apache Hadoop on HDInsight](#)
- [Use Apache Sqoop with Hadoop in HDInsight](#)
- [Overview of Apache Spark Streaming](#)
- [Structured Streaming tutorial](#)
- [Use Azure Event Hubs from Apache Kafka applications](#)

## Related resources

- [Hadoop migration to Azure](#)

- Apache HDFS migration to Azure
- Apache Kafka migration to Azure
- Apache Sqoop migration to Azure

# Apache Kafka migration to Azure

Azure HDInsight

Azure Cosmos DB

Azure Data Lake Storage

Azure Synapse Analytics

Azure Stream Analytics

Apache Kafka [↗](#) is a highly scalable and fault tolerant distributed messaging system that implements a publish-subscribe architecture. It's used as an ingestion layer in real-time streaming scenarios, such as IoT and real-time log monitoring systems. It's also used increasingly as the immutable append-only data store in Kappa architectures.

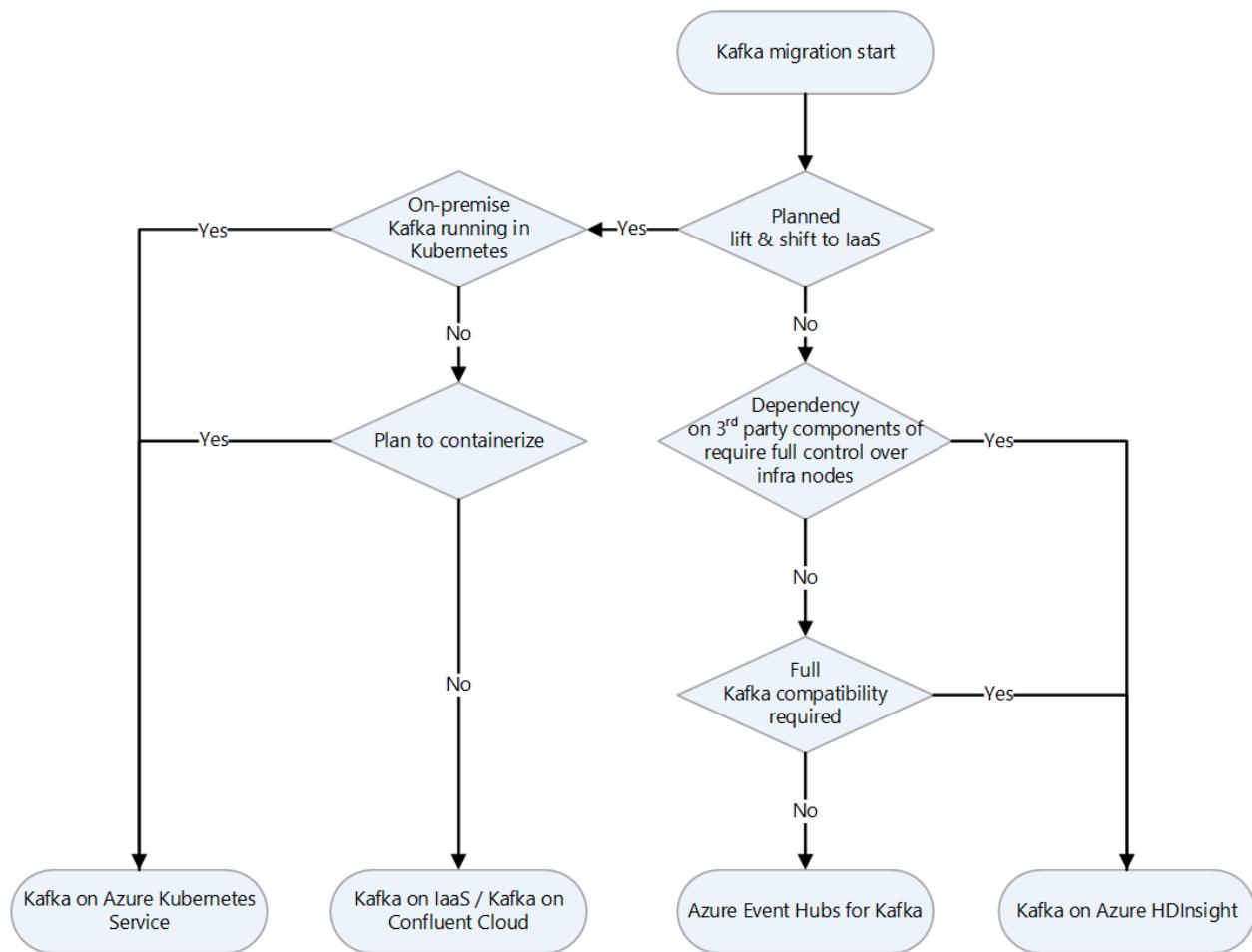
*Apache [↗](#)®, Apache Spark® [↗](#), Apache Hadoop® [↗](#), Apache HBase [↗](#), Apache Storm® [↗](#), Apache Swoop® [↗](#), Apache Kafka® [↗](#), and the flame logo are either registered trademarks or trademarks of the Apache Software Foundation in the United States and/or other countries. No endorsement by The Apache Software Foundation is implied by the use of these marks.*

## Migration approach

This article presents various strategies for migrating Kafka to Azure:

- Migrate Kafka to Azure infrastructure as a service (IaaS)
- Migrate Kafka to Azure Event Hubs for Kafka
- Migrate Kafka on Azure HDInsight
- Use AKS with Kafka on HDInsight

Here's a decision flowchart for deciding which to use:



## Migrate Kafka to Azure infrastructure as a service (IaaS)

For one way to migrate Kafka to Azure IaaS, see [Kafka on Ubuntu VMs](#).

## Migrate Kafka to Azure Event Hubs for Kafka

Event Hubs provides an endpoint that's compatible with the Apache Kafka producer and consumer APIs. This endpoint can be used by most Apache Kafka client applications, so it's an alternative to running a Kafka cluster on Azure. The endpoint supports clients that use versions 1.0 and later of the APIs. For more information about this feature, see [Azure Event Hubs for Apache Kafka overview](#).

To learn how to migrate your Apache Kafka applications to use Azure Event Hubs, see [Migrate to Azure Event Hubs for Apache Kafka Ecosystems](#).

## Kafka and Event Hubs feature differences

[Expand table](#)

How are Kafka and Event Hubs similar?	How are Kafka and Event Hubs different?
Both use partitions.	There are differences in these areas:
Partitions are independent.	<ul style="list-style-type: none"> <li>• PaaS vs. software</li> </ul>
Both use a client-side cursor concept.	<ul style="list-style-type: none"> <li>• Partitioning</li> </ul>
Both can scale to very high workloads.	<ul style="list-style-type: none"> <li>• APIs</li> </ul>
Conceptually they are nearly the same.	<ul style="list-style-type: none"> <li>• Runtime</li> </ul>
Neither uses the HTTP protocol for receiving.	<ul style="list-style-type: none"> <li>• Protocols</li> </ul>
	<ul style="list-style-type: none"> <li>• Durability</li> </ul>
	<ul style="list-style-type: none"> <li>• Security</li> </ul>
	<ul style="list-style-type: none"> <li>• Throttling</li> </ul>

## Partitioning differences

↔ [Expand table](#)

Kafka	Event Hubs
Scale is managed by partition count.	Scale is managed by throughput units.
You must load-balance partitions across machines.	Load balancing is automatic.
You must manually re-shard by using split and merge.	Repartitioning isn't required.

## Durability differences

[Expand table](#)

Kafka	Event Hubs
Volatile by default	Always durable
Replicated after ACK	Replicated before ACK
Depends on disk and quorum	Provided by storage

## Security differences

[Expand table](#)

Kafka	Event Hubs
SSL and SASL	SAS and SASL/PLAIN RFC 4618
File-like ACLs	Policy
Optional transport encryption	Mandatory TLS
User based	Token based (unlimited)

## Other differences

[Expand table](#)

Kafka	Event Hubs
Kafka doesn't throttle.	Event Hubs supports throttling.
Kafka uses a proprietary protocol.	Event Hubs uses AMQP 1.0 protocol.
Kafka doesn't use HTTP for send.	Event Hubs uses HTTP Send and Batch Send.

# Migrate Kafka on Azure HDInsight

You can migrate Kafka to Kafka on Azure HDInsight. For more information, see [What is Apache Kafka in Azure HDInsight?](#).

## Use AKS with Kafka on HDInsight

See [Use Azure Kubernetes Service with Apache Kafka on HDInsight](#).

## Kafka Data Migration

You can use Kafka's [MirrorMaker tool](#) to replicate topics from one cluster to another. This technique can help you migrate data after a Kafka cluster is provisioned. For more information, see [Use MirrorMaker to replicate Apache Kafka topics with Kafka on HDInsight](#).

Here's a migration approach that uses mirroring:

- Move producers first and then move consumers. When you migrate the producers you prevent production of new messages on the source Kafka.
- After the source Kafka consumes all remaining messages, you can migrate the consumers.

Here are the implementation steps:

1. Change the Kafka connection address of the producer client to point to the new Kafka instance.
2. Restart the producer business services and send new messages to the new Kafka instance.
3. Wait for the data in the source Kafka to be consumed.
4. Change the Kafka connection address of the consumer client to point to the new Kafka instance.
5. Restart the consumer business services to consume messages from the new Kafka instance.
6. Verify that consumers succeed in getting data from the new Kafka instance.

## Monitor the Kafka cluster

You can use Azure Monitor logs to analyze logs that are generated by Apache Kafka on HDInsight. For more information, see: [Analyze logs for Apache Kafka on HDInsight](#).

# Apache Kafka Streams API

The Kafka Streams API makes it possible to process data in near real-time, and it provides the ability to join and aggregate data. There are many more features of the API worth knowing about. For more information, see [Introducing Kafka Streams: Stream Processing Made Simple - Confluent](#).

## The Microsoft and Confluent partnership

Confluent provides a cloud-native service for Apache Kafka. Microsoft and Confluent have a strategic alliance. For more information, see:

- [Confluent and Microsoft Announce Strategic Alliance](#)
- [Introducing seamless integration between Microsoft Azure and Confluent Cloud](#)

## Contributors

*This article is maintained by Microsoft. It was originally written by the following contributors.*

Principal authors:

- [Namrata Maheshwary](#) | Senior Cloud Solution Architect
- [Raja N](#) | Director, Customer Success
- [Hideo Takagi](#) | Cloud Solution Architect
- [Ram Yerrabotu](#) | Senior Cloud Solution Architect

Other contributors:

- [Ram Baskaran](#) | Senior Cloud Solution Architect
- [Jason Bouska](#) | Senior Software Engineer
- [Eugene Chung](#) | Senior Cloud Solution Architect
- [Pawan Hosattu](#) | Senior Cloud Solution Architect - Engineering
- [Daman Kaur](#) | Cloud Solution Architect
- [Danny Liu](#) | Senior Cloud Solution Architect - Engineering
- [Jose Mendez](#) | Senior Cloud Solution Architect
- [Ben Sadeghi](#) | Senior Specialist
- [Sunil Sattiraju](#) | Senior Cloud Solution Architect
- [Amanjeet Singh](#) | Principal Program Manager
- [Nagaraj Seepalapudur Venkatesan](#) | Senior Cloud Solution Architect - Engineering

*To see non-public LinkedIn profiles, sign in to LinkedIn.*

# Next steps

## Azure product introductions

- [Introduction to Azure Data Lake Storage Gen2](#)
- [What is Apache Spark in Azure HDInsight?](#)
- [What is Apache Hadoop in Azure HDInsight?](#)
- [What is Apache HBase in Azure HDInsight?](#)
- [What is Apache Kafka in Azure HDInsight?](#)
- [Overview of enterprise security in Azure HDInsight](#)

## Azure product reference

- [Microsoft Entra documentation](#)
- [Azure Cosmos DB documentation](#)
- [Azure Data Factory documentation](#)
- [Azure Databricks documentation](#)
- [Azure Event Hubs documentation](#)
- [Azure Functions documentation](#)
- [Azure HDInsight documentation](#)
- [Microsoft Purview data governance documentation](#)
- [Azure Stream Analytics documentation](#)
- [Azure Synapse Analytics](#)

## Other

- [Enterprise Security Package for Azure HDInsight](#)
- [Develop Java MapReduce programs for Apache Hadoop on HDInsight](#)
- [Use Apache Sqoop with Hadoop in HDInsight](#)
- [Overview of Apache Spark Streaming](#)
- [Structured Streaming tutorial](#)
- [Use Azure Event Hubs from Apache Kafka applications](#)

## Related resources

- [Hadoop migration to Azure](#)
- [Apache HDFS migration to Azure](#)
- [Apache HBase migration to Azure](#)
- [Apache Sqoop migration to Azure](#)

# Apache Swoop migration to Azure

Azure HDInsight

Azure Cosmos DB

Azure Data Lake Storage

Azure Synapse Analytics

Azure Stream Analytics

Apache Swoop [↗](#) is a tool for transferring data between Apache Hadoop clusters and relational databases. It has a command-line interface.

You can use Swoop to import data to HDFS from relational databases such as MySQL, PostgreSQL, Oracle, and SQL Server, and to export HDFS data to such databases. Swoop can use MapReduce and Apache Hive to convert data on Hadoop. Advanced features include incremental loading, formatting by using SQL, and updating datasets. Swoop operates in parallel to achieve high-speed data transfer.

## ⓘ Note

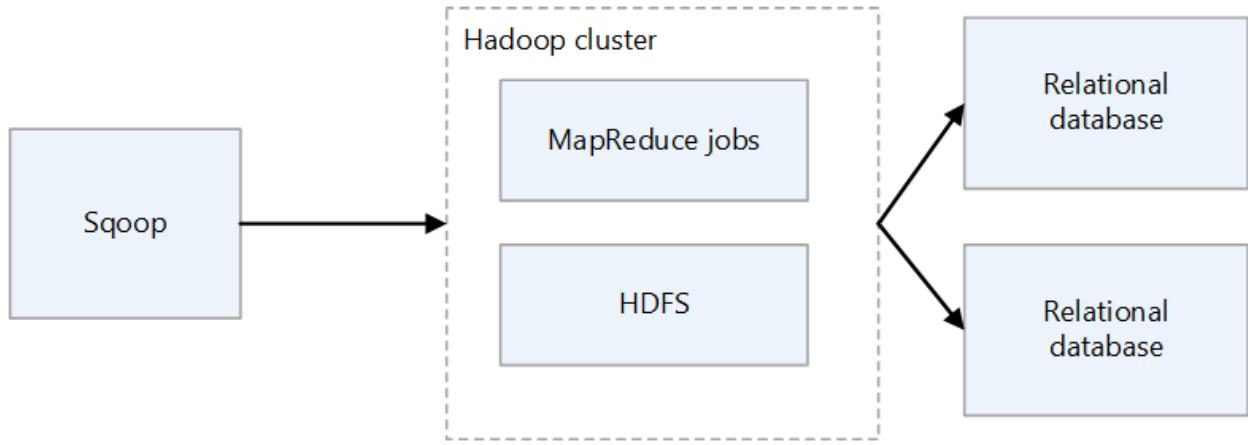
The Swoop project has retired. Swoop was moved into the Apache Attic in June, 2021. The website, downloads, and issue tracker all remain open. See [Apache Swoop in the Apache Attic](#) [↗](#) for more information.

*Apache [↗](#) ®, Apache Spark [↗](#) ®, Apache Hadoop [↗](#) ®, Apache HBase [↗](#) , Apache Hive [↗](#) , Apache Ranger [↗](#) ®, Apache Storm [↗](#) ®, Apache Swoop [↗](#) ®, Apache Kafka [↗](#) ®, and the flame logo are either registered trademarks or trademarks of the Apache Software Foundation in the United States and/or other countries. No endorsement by The Apache Software Foundation is implied by the use of these marks.*

## Swoop architecture and components

There are two versions of Swoop: Swoop1 and Swoop2. Swoop1 is a simple client tool, whereas Swoop2 has a client/server architecture. They aren't compatible with one another, and they differ in usage. Swoop2 isn't feature complete, and isn't intended for production deployment.

### Swoop1 architecture



## Sqoop1 import and export

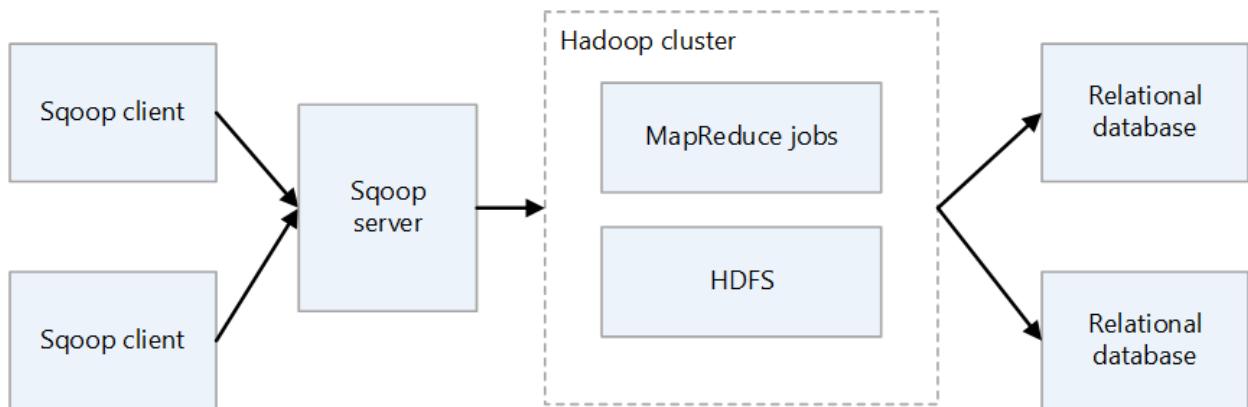
- **Import**

Reads data from relational databases and outputs data to HDFS. Each record in the relational databases table is output as a single row in HDFS. Text, SequenceFiles, and Avro are the file formats that can be written to HDFS.

- **Export**

Reads data from HDFS and transfers it to relational databases. The target relational databases support both insert and update.

## Sqoop2 architecture



- **Sqoop server**

Provides an entry point for Sqoop clients.

- **Sqoop client**

Interacts with the Sqoop server. The client can be on any node, provided that the client can communicate with the server. Because the client needs only to

communicate with the server, there's no need to make settings as you would with MapReduce.

## Challenges of Sqoop on-premises

Here are some common challenges of an on-premises Sqoop deployment:

- It can be difficult to scale, depending on the hardware and datacenter capacity.
- It can't be easily scaled on demand.
- When support ends for aging infrastructure, you can be forced to replace and upgrade.
- There's a lack of native tools to provide:
  - Cost transparency
  - Monitoring
  - DevOps
  - Automation

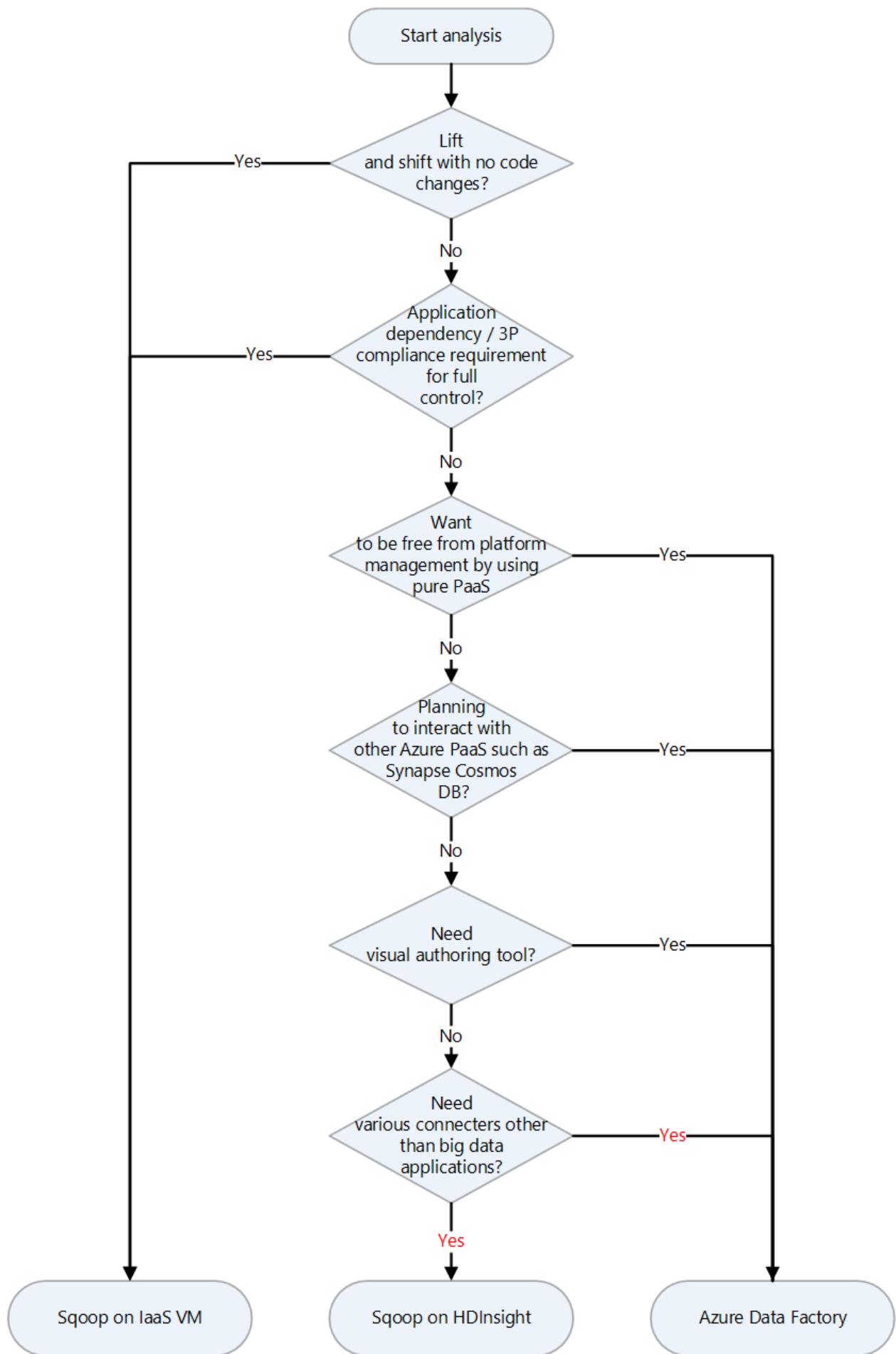
## Considerations

- When you migrate Sqoop to Azure, if your data source remains on-premises, you need to consider its connectivity. You can establish a VPN connection over the internet between Azure and your existing on-premises network, or you can use Azure ExpressRoute to make a private connection.
- When you migrate Sqoop to Azure HDInsight, consider your Sqoop version. HDInsight supports only Sqoop1, so if you're using Sqoop2 in your on-premises environment, you have to replace it with Sqoop1 on HDInsight, or keep Sqoop2 independent.
- When you migrate Sqoop to Azure Data Factory, you need to consider data file formats. Data Factory doesn't support the SequenceFile format. The lack of support can be a problem if your Sqoop implementation imports data in SequenceFile format. For more information, see [File format](#).

## Migration approach

Azure has several migration targets for Apache Sqoop. Depending on requirements and product features, you can choose between Azure IaaS virtual machines (VMs), Azure HDInsight, and Azure Data Factory.

Here's a decision chart for selecting a migration target:



The migration targets are discussed in the following sections:

- Lift and shift migration to Azure IaaS

- [Migrate to HDInsight](#)
- [Migrate to Data Factory](#)

## Lift and shift migration to Azure IaaS

If you choose Azure IaaS VMs as the migration destination for your on-premises Sqoop, you can do a lift and shift migration. You use the same version of Sqoop to create a completely controllable environment. Therefore, you don't have to make any changes to the Sqoop software. Sqoop works with a Hadoop cluster and is usually migrated along with a Hadoop cluster. The following articles are guides for a lift and shift migration of a Hadoop cluster. Choose the article that's applies to the service to be migrated.

- [Apache Hive](#)
- [Apache Ranger](#)
- [Apache HBase](#)

## Preparation for migration

To prepare for migration, you plan for migration and you establish a network connection.

## Plan for migration

Gather the following information to prepare for the migration of your on-premises Sqoop. The information helps you to determine the size of the destination virtual machine and to plan the software components and network configurations.

[\[+\] Expand table](#)

Item	Background
Current host size	Get information about the CPU, memory, disk, and other components of the host or virtual machine on which the Sqoop client or server is running. You use this information to estimate the base size required for your Azure virtual machine.
Host and application metrics	Get the resource usage information (CPU, memory, disk, and other components) of the machine that runs the Sqoop client and estimate the resources actually used. If you're using less resources than what's allocated to your host, consider downsizing when you migrate to Azure. After you identify the required amount of resources, select the

Item	Background
	type of virtual machine to migrate to by referring to <a href="#">Azure virtual machine size</a> .
Sqoop version	Check the version of the on-premises Sqoop to determine which version of Sqoop to install on the Azure virtual machine. If you're using a distribution such as Cloudera or Hortonworks, the version of the component depends on the version of that distribution.
Running jobs and scripts	Identify the jobs that run Sqoop and the methods of scheduling them. The jobs and methods are candidates for migration.
Databases to connect	Identify the databases that Sqoop connects to, as specified by import and export commands in the Sqoop jobs. After you identify them, you need to see if you can connect to those databases after you migrate Sqoop to your Azure virtual machines. If some of the databases you connect to are still on-premises, you need a network connection between on-premises and Azure. For more information, see the <a href="#">Establish a network connection</a> section.
Plugins	Identify the Sqoop plugins that you use and determine whether you can migrate them.
High availability, business continuity, disaster recovery	Determine whether the troubleshooting techniques that you use on-premises can be used on Azure. For example, if you have an active/standby configuration on two nodes, prepare two Azure virtual machines for Sqoop clients that have the same configuration. The same applies when configuring disaster recovery.

## Establish a network connection

If some of the databases that you connect to remain on-premises, you need a network connection between on-premises and Azure.

There are two main options for connecting on-premises and Azure on a private network:

- **VPN Gateway**

You can use Azure VPN Gateway to send encrypted traffic between your Azure virtual network and your on-premises location over the public internet. This technique is inexpensive and easy to set up. However, due to the encrypted connection over the internet, the communication bandwidth isn't guaranteed. If

you need to guarantee bandwidth, you should choose ExpressRoute, which is the second option. For more information about the VPN option, see [What is VPN Gateway?](#) and [VPN Gateway design](#).

- **ExpressRoute**

ExpressRoute can connect your on-premises network to Azure or to Microsoft 365 by using a private connection that's provided by a connectivity provider.

ExpressRoute doesn't go through the public internet, so it's more secure, more reliable, and has more consistent latencies than connections over the internet. In addition, the bandwidth options of the line that you purchase can guarantee stable latencies. For more information, see [What is Azure ExpressRoute?](#).

If these private connection methods don't meet your needs, consider Azure Data Factory as a migration destination. The self-hosted integration runtime in Data Factory makes it possible for you to transfer data from on-premises to Azure without having to configure a private network.

## Migrate data and settings

When you migrate on-premises Sqoop to Azure virtual machines, include the following data and settings:

- **Sqoop config files:** It depends on your environment, but the following files are often included:
  - `sqoop-site.xml`
  - `sqoop-env.xml`
  - `password-file`
  - `oraoop-site.xml`, if you use Oraoop
- **Saved jobs:** If you saved jobs in the Sqoop metastore by using the `sqoop job --create` command, you need to migrate them. The save destination of metastore is defined in `sqoop-site.xml`. If the shared metastore isn't set, look for the saved jobs in the `.sqoop` subdirectory of the home directory of the user that runs metastore.

You can use the following commands to see information about the saved jobs.

- Get the saved job list:

```
sqoop job --list
```

- View parameters for saved jobs

```
sqoop job --show <job-id>
```

- **Scripts:** If you have script files that run Sqoop, you need to migrate them.
- **Scheduler:** If you schedule the execution of Sqoop, you need to identify its scheduler, such as a Linux cron job or a job management tool. Then you need to consider whether the scheduler can be migrated to Azure.
- **Plugins:** If you're using any custom plugins in Sqoop, for example a connector to an external database, you need to migrate them. If you created a patch file, apply the patch to the migrated Sqoop.

## Migrate to HDInsight

HDInsight bundles Apache Hadoop components and the HDInsight platform into a package that's deployed on a cluster. Instead of migrating Sqoop itself to Azure, it's more typical to run Sqoop on an HDInsight cluster. For more information about using HDInsight to run open-source frameworks such as Hadoop and Spark, see [What is Azure HDInsight?](#) and [Guide to Migrating Big Data Workloads to Azure HDInsight](#).

See the following articles for the component versions in HDInsight.

- [HDInsight 4.0 component versions](#)
- [HDInsight 3.6 component versions](#)

## Migrate to Data Factory

Azure Data Factory is a fully managed, serverless, data integration service. It can be scaled on demand according to factors such as volume of data. It has a GUI for intuitive editing and development by using Python, .NET, and Azure Resource Manager templates (ARM templates).

## Connection to data sources

See the appropriate article for a list of the standard Sqoop connectors:

- [Sqoop1 connectors](#)
- [Sqoop2 connectors](#)

Data Factory has a large number of connectors. For more information, see [Azure Data Factory and Azure Synapse Analytics connector overview](#).

The following table is an example that shows the Data Factory connectors to use for Sqoop1 version 1.4.7 and Sqoop2 version 1.99.7. Be sure to refer to the latest documentation, because the list of supported versions can change.

Sqoop1 - 1.4.7	Sqoop2 - 1.99.7	Data Factory	Considerations
MySQL JDBC Connector	Generic JDBC Connector	MySQL, Azure Database for MySQL	
MySQL Direct Connector	N/A	N/A	Direct Connector uses mysqldump to input and output data without going through JDBC. The method is different in Data Factory, but the MySQL connector can be used instead.
Microsoft SQL Connector	Generic JDBC Connector	SQL Server, Azure SQL Database, Azure SQL Managed Instance	
PostgreSQL Connector	PostgreSQL, Generic JDBC Connector	Azure Database for PostgreSQL	
PostgreSQL Direct Connector	N/A	N/A	Direct Connector doesn't go through JDBC and uses the <b>COPY</b> command to input and output data. The method is different in Data Factory, but the PostgreSQL connector can be used instead.
pg_bulkload connector	N/A	N/A	Load into PostgreSQL by using pg_bulkload. The method is different in Data Factory, but the PostgreSQL connector can be used instead.

<b>Sqoop1 - 1.4.7</b>	<b>Sqoop2 - 1.99.7</b>	<b>Data Factory</b>	<b>Considerations</b>
Netezza Connector	Generic JDBC Connector	Netteza	
Data Connector for Oracle and Hadoop	Generic JDBC Connector	Oracle	
N/A	FTP Connector	FTP	
N/A	SFTP Connector	SFTP	
N/A	Kafka Connector	N/A	Data Factory can't connect directly to Kafka. Consider using Spark Streaming such as Azure Databricks or HDInsight to connect to Kafka.
N/A	Kite Connector	N/A	Data Factory can't connect directly to Kite.
HDFS	HDFS	HDFS	Data Factory supports HDFS as a source, but not as a sink.

## Connect to databases on-premises

If, after you migrate Sqoop to Data Factory, you still need to copy data between a data store in your on-premises network and Azure, consider using these methods:

- [Self-hosted integration runtime](#)
- [Managed virtual network by using a private endpoint](#)

## Self-hosted integration runtime

If you're trying to integrate data in a private network environment where there's no direct communication path from the public cloud environment, you can do the following

to improve security:

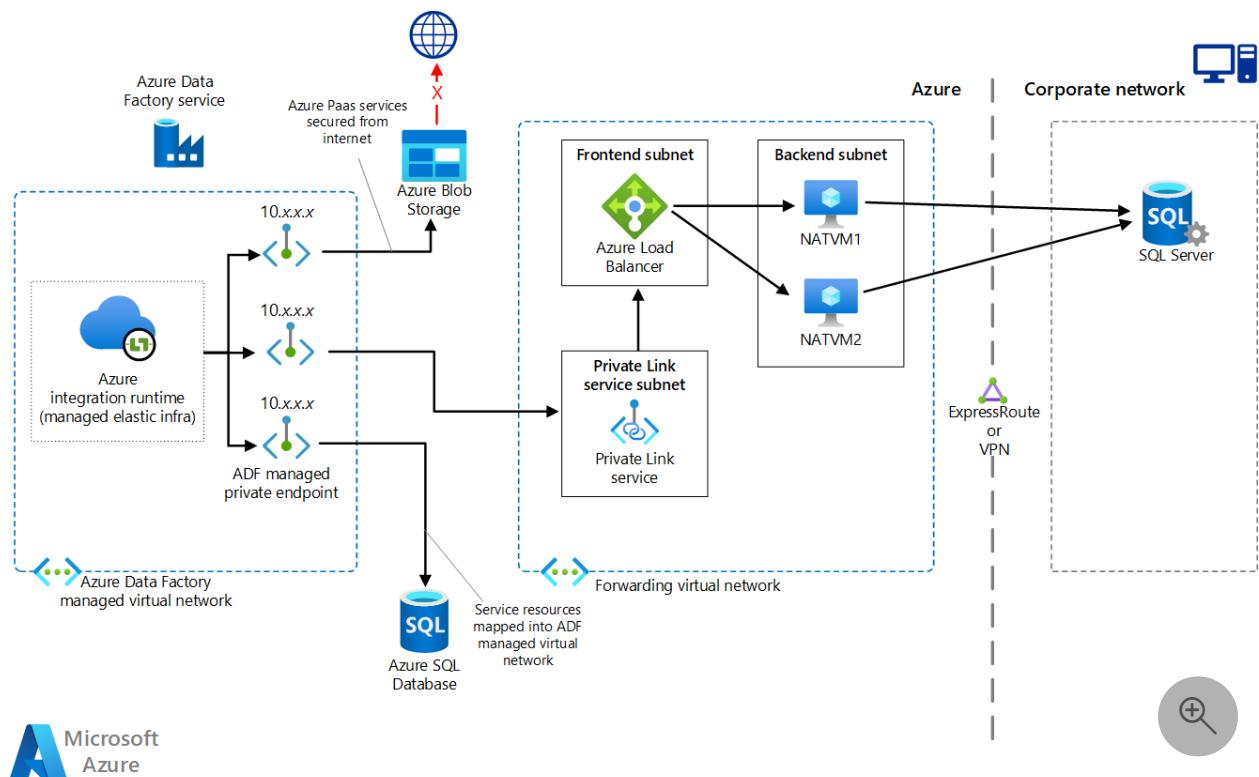
- Install a self-hosted integration runtime in the on-premises environment, either in the internal firewall or in the virtual private network.
- Make an HTTPS-based outbound connection from the self-hosted integration runtime to Azure in order to establish a connection for data movement.

Self-hosted integration runtime is only supported on Windows. You can also achieve scalability and high availability by installing and associating self-hosted integration runtimes on multiple machines. Self-hosted integration runtime is also responsible for dispatching data transformation activities to resources that aren't on-premises or in the Azure virtual network.

For information about how to set up self-hosted integration runtime, see [Create and configure a self-hosted integration runtime](#).

## Managed virtual network by using a private endpoint

If you have a private connection between on-premises and Azure (such as ExpressRoute or VPN Gateway), you can use managed virtual network and private endpoint in Data Factory to make a private connection to your on-premises databases. You can use virtual networks to forward traffic to your on-premises resources, as shown in the following diagram, to access your on-premises resources without going through the internet.



Download a [Visio file](#) of this architecture.

For more information, see [Tutorial: How to access on-premises SQL Server from Data Factory Managed VNet using Private Endpoint](#).

## Network options

Data Factory has two network options:

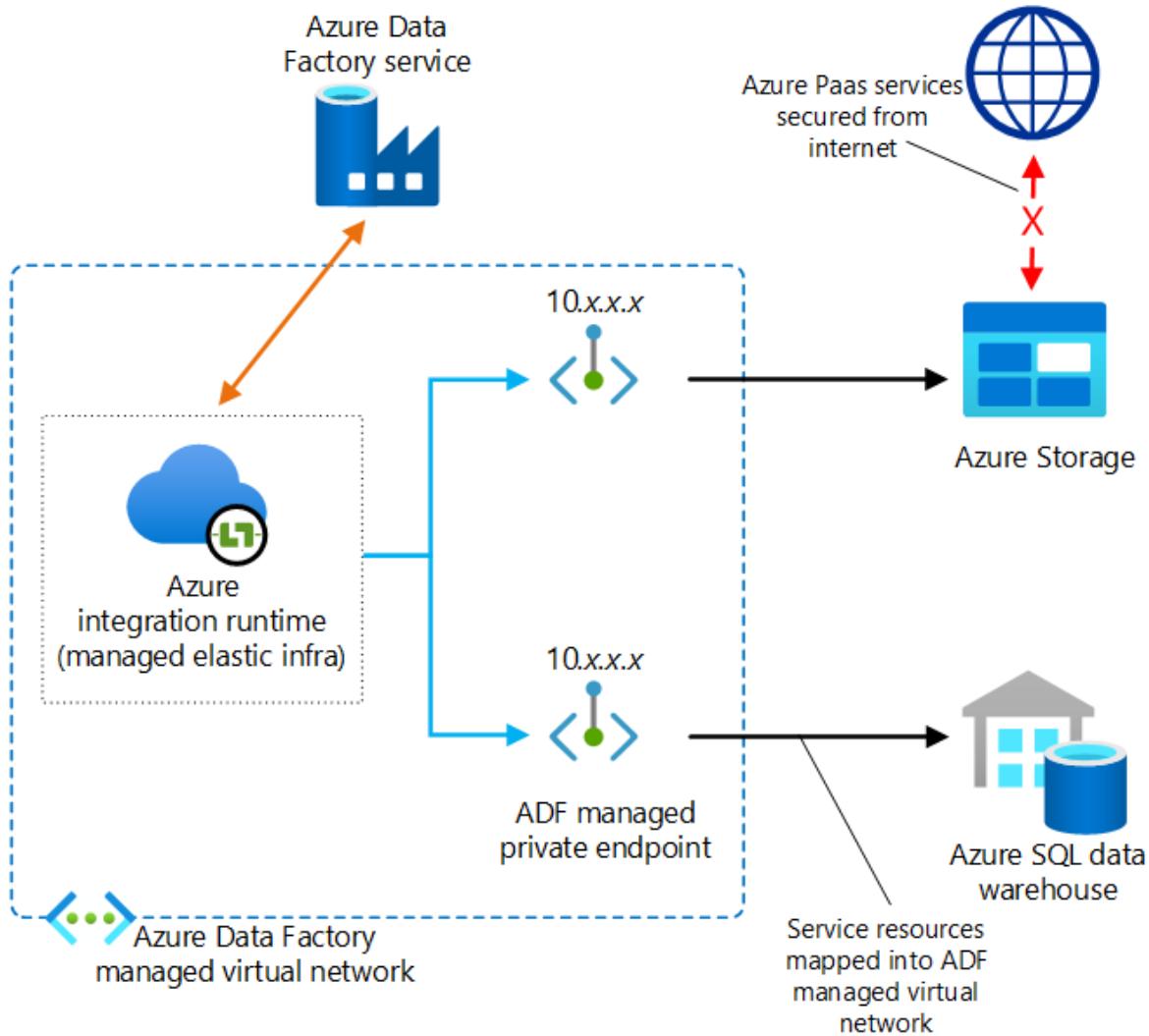
- [Managed virtual network](#)
- [Private link](#)

Both build a private network and help secure the process of data integration. They can be used at the same time.

### Managed virtual network

You can deploy the integration runtime, which is the Data Factory runtime, within a managed virtual network. By deploying a private endpoint such as a data store that connects to the managed virtual network, you can improve data integration safety within a closed private network.

↔ Command and control  
→ Data

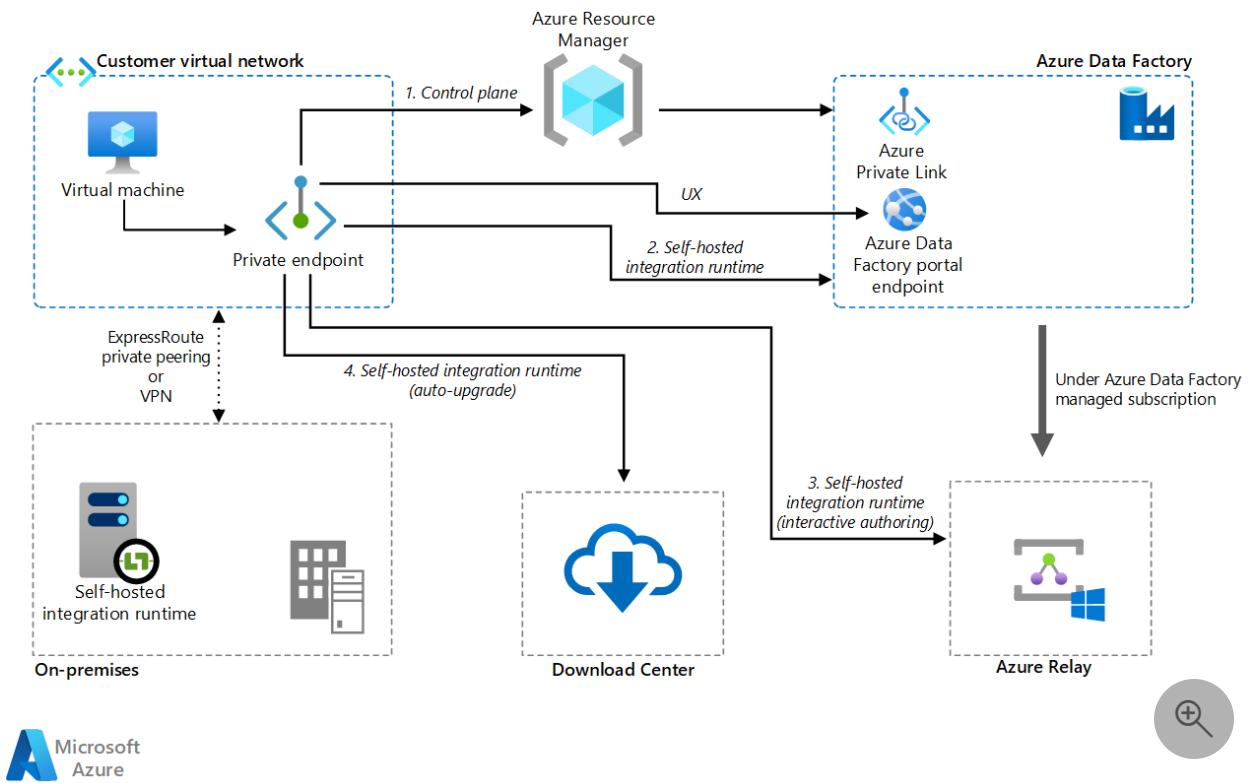


Download a [Visio file](#) of this architecture.

For more information, see [Azure Data Factory managed virtual network](#).

## Private link

You can use [Azure Private Link for Azure Data Factory](#) to connect to Data Factory.



Download a [Visio file](#) of this architecture.

For more information, see [What is a private endpoint?](#) and [Private Link documentation](#).

## Performance of data copy

Sqoop improves data transfer performance by using MapReduce for parallel processing. After you migrate Sqoop, Data Factory can adjust performance and scalability for scenarios that perform large-scale data migrations.

A *data integration unit* (DIU) is a Data Factory unit of performance. It's a combination of CPU, memory, and network resource allocation. Data Factory can adjust up to 256 DIUs for copy activities that use the Azure integration runtime. For more information, see [Data Integration Units](#).

If you use self-hosted integration runtime, you can improve performance by scaling the machine that hosts the self-hosted integration runtime. The maximum scale-out is four nodes.

For more information about making adjustments to achieve your desired performance, see [Copy activity performance and scalability guide](#).

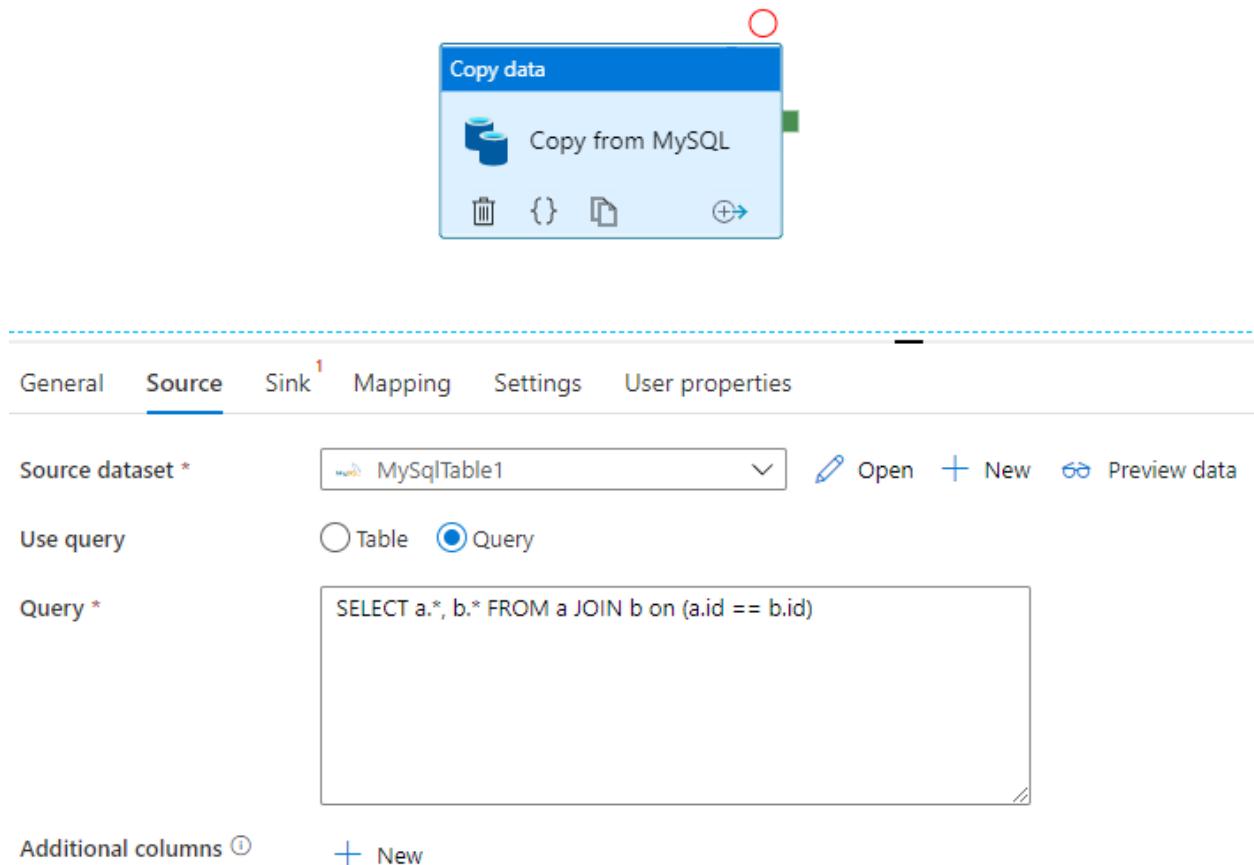
## Apply SQL

Sqoop can import the result set of a SQL query, as shown in this example:

```
sqoop
```

```
$ sqoop import \
--query 'SELECT a.*, b.* FROM a JOIN b on (a.id == b.id) WHERE
$CONDITIONS' \
--split-by a.id --target-dir /user/foo/joinresults
```

Data Factory can also query the database and copy the result set:



General   Source <sup>1</sup>   Sink   Mapping   Settings   User properties

Source dataset \* MySQLTable1  

Use query    Table  Query

Query \*   
SELECT a.\*, b.\* FROM a JOIN b on (a.id == b.id)

Additional columns

See [Copy activity properties](#) for an example that gets the result set of a query on a MySQL database.

## Data transformation

Both Data Factory and HDInsight can perform various data transformation activities.

### Transform data by using Data Factory activities

Data Factory can perform a variety of data transformation activities, such as data flow and data wrangling. For both, you define the transformations by using a visual UI. You can also use the activities of various Hadoop components of HDInsight, Databricks, stored procedures, and other custom activities. Consider using these activities when you migrate Sqoop and want to include data transformations in the process. See [Transform data in Azure Data Factory](#) for more information.

## Transform data by using HDInsight activities

The various HDInsight activities in an Azure Data Factory pipeline, including Hive, Pig, MapReduce, Streaming, and Spark, can run programs and queries on either your own cluster or on an [on-demand HDInsight cluster](#). If you migrate a Soop implementation that uses data transformation logic of the Hadoop ecosystem, it's easy to migrate the transformations to HDInsight activities. For details, see the following articles.

- [Transform data using Hadoop Hive activity in Azure Data Factory or Synapse Analytics](#)
- [Transform data using Hadoop MapReduce activity in Azure Data Factory or Synapse Analytics](#)
- [Transform data using Hadoop Pig activity in Azure Data Factory or Synapse Analytics](#)
- [Transform data using Spark activity in Azure Data Factory and Synapse Analytics](#)
- [Transform data using Hadoop Streaming activity in Azure Data Factory or Synapse Analytics](#)

## File format

Soop supports text, SequenceFile, and Avro as file formats when it imports data into HDFS. Data Factory doesn't support HDFS as a data sink, but it does use Azure Data Lake Storage or Azure Blob Storage as file storage. For more information on HDFS migration, see [Apache HDFS migration](#).

The supported formats for Data Factory to write to file storage are text, binary, Avro, JSON, ORC, and Parquet, but not SequenceFile. You can use an activity such as Spark to convert a file to SequenceFile by using `saveAsSequenceFile`:

Java

```
data.saveAsSequenceFile(<path>)
```

## Scheduling jobs

Soop doesn't provide scheduler functionality. If you're running Soop jobs on a scheduler, you need to migrate that functionality to Data Factory. Data Factory can use triggers to schedule the execution of the data pipeline. Choose a Data Factory trigger according to your existing scheduling configuration. Here are the types of triggers.

- **Schedule trigger:** A schedule trigger runs the pipeline on a wall-clock schedule.

- **Tumbling window trigger:** A tumbling window trigger runs periodically from a specified start time while maintaining its state.
- **Event-based trigger:** An event-based trigger triggers the pipeline in response to the event. There are two types of event-based triggers:
  - **Storage event trigger:** A storage event trigger triggers the pipeline in response to a storage event such as creating, deleting, or writing to a file.
  - **Custom event trigger:** A custom-event trigger triggers the pipeline in response to an event that's sent to a custom topic in an event grid. For information about custom topics, see [Custom topics in Azure Event Grid](#).

For more information about triggers, see [Pipeline execution and triggers in Azure Data Factory](#) or [Azure Synapse Analytics](#).

## Contributors

*This article is maintained by Microsoft. It was originally written by the following contributors.*

Principal authors:

- [Namrata Maheshwary](#) | Senior Cloud Solution Architect
- [Raja N](#) | Director, Customer Success
- [Hideo Takagi](#) | Cloud Solution Architect
- [Ram Yerrabotu](#) | Senior Cloud Solution Architect

Other contributors:

- [Ram Baskaran](#) | Senior Cloud Solution Architect
- [Jason Bouska](#) | Senior Software Engineer
- [Eugene Chung](#) | Senior Cloud Solution Architect
- [Pawan Hosattu](#) | Senior Cloud Solution Architect - Engineering
- [Daman Kaur](#) | Cloud Solution Architect
- [Danny Liu](#) | Senior Cloud Solution Architect - Engineering
- [Jose Mendez](#) | Senior Cloud Solution Architect
- [Ben Sadeghi](#) | Senior Specialist
- [Sunil Sattiraju](#) | Senior Cloud Solution Architect
- [Amanjeet Singh](#) | Principal Program Manager
- [Nagaraj Seeplapudur Venkatesan](#) | Senior Cloud Solution Architect - Engineering

*To see non-public LinkedIn profiles, sign in to LinkedIn.*

## Next steps

## Azure product introductions

- [Introduction to Azure Data Lake Storage Gen2](#)
- [What is Apache Spark in Azure HDInsight?](#)
- [What is Apache Hadoop in Azure HDInsight?](#)
- [What is Apache HBase in Azure HDInsight?](#)
- [What is Apache Kafka in Azure HDInsight?](#)
- [Overview of enterprise security in Azure HDInsight](#)

## Azure product reference

- [Microsoft Entra documentation](#)
- [Azure Cosmos DB documentation](#)
- [Azure Data Factory documentation](#)
- [Azure Databricks documentation](#)
- [Azure Event Hubs documentation](#)
- [Azure Functions documentation](#)
- [Azure HDInsight documentation](#)
- [Microsoft Purview data governance documentation](#)
- [Azure Stream Analytics documentation](#)
- [Azure Synapse Analytics](#)

## Other

- [Enterprise Security Package for Azure HDInsight](#)
- [Develop Java MapReduce programs for Apache Hadoop on HDInsight](#)
- [Use Apache Sqoop with Hadoop in HDInsight](#)
- [Overview of Apache Spark Streaming](#)
- [Structured Streaming tutorial](#)
- [Use Azure Event Hubs from Apache Kafka applications](#)

## Related resources

- [Hadoop migration to Azure](#)
- [Apache HDFS migration to Azure](#)
- [Apache HBase migration to Azure](#)
- [Apache Kafka migration to Azure](#)