

# Deploy IaaS solutions with Azure SQL

---

## 1. Introduction

<https://learn.microsoft.com/en-us/training/modules/deploy-iaas-solutions-with-azure-sql/1-introduction>

## Introduction

---

Completed

The most common reason for deploying SQL Server in an Azure Virtual Machine (VM) is because you want an easy, straightforward method to migrate an existing on-premises SQL Server into the cloud.

Understanding the options and methods for deploying SQL Server in an Azure VM is critical to ensure a successful migration. Infrastructure as a Service (IaaS) allows for greater flexibility in configuration. This flexibility means that you, as a database administrator, must plan your configuration carefully. Choosing the proper VM sizing, storage, and networking options are crucial to ensure adequate performance for your workloads.

## Learning objectives

At the end of this module, you'll be able to:

- Explore the basics of SQL Server in an Infrastructure as a Service (IaaS) offering
- Learn how hybrid scenario works
- Explore performance and security options available
- Understand the high availability and disaster recovery options

## 2. Explain IaaS options to deploy SQL Server in Azure

# Explain IaaS options to deploy SQL Server in Azure

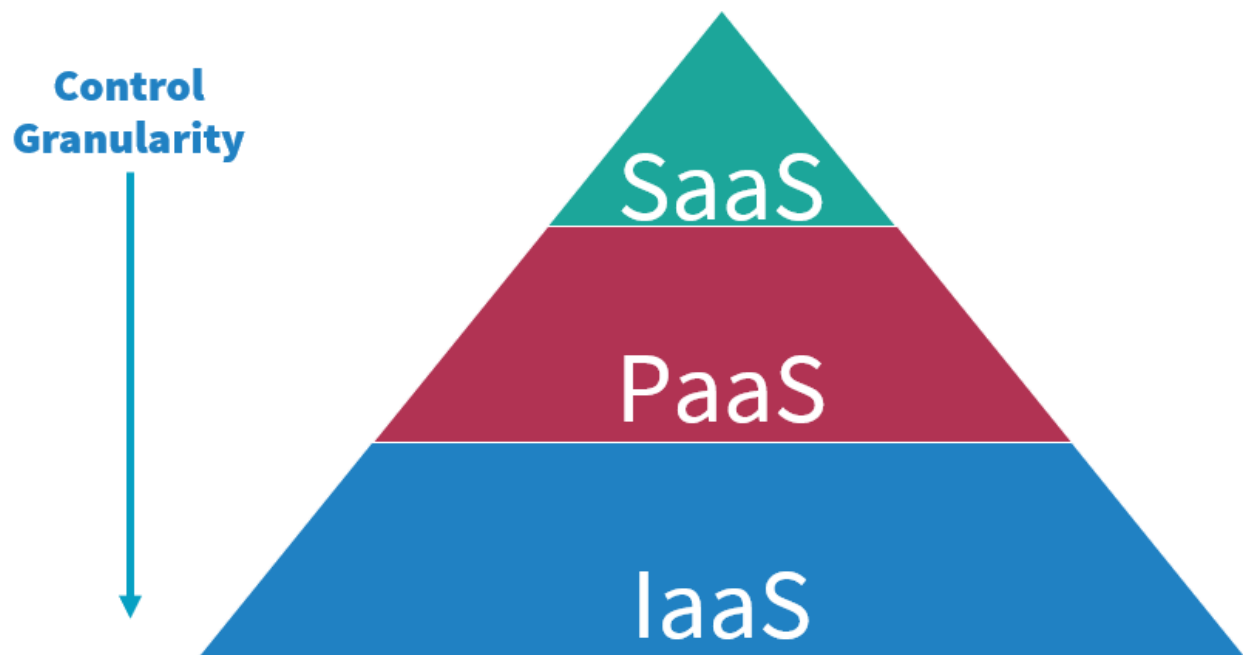
---

Completed

Many applications require a VM running SQL Server. Some reasons for this option include:

- **Older versions of SQL Server**—If an application requires an older version of SQL Server for vendor support, running inside a VM is the best option for those applications, because it allows for the application to be supported by that vendor.
- **Use of other SQL Server services**—While Analysis Services and to an extent Integration Services (by using Azure Data Factory) are available as PaaS offerings, many users maximize their licensing by running SQL Server Analysis Services, Integration Services, or Reporting Services on the same machine as the database engine.
- **General application incompatibility**—This reason is a somewhat of a catch-all. For example, Azure SQL Database doesn't support cross-database querying, while managed instance does. Some applications require other services to coexist with the database instance in a manner that isn't compatible with a PaaS offering.

Infrastructure as a Service (IaaS) allows the administrator to have more granular access over specific settings of the underlying infrastructure than the other Azure offerings. While the Azure platform manages the underlying server and network hardware, you still have access to the virtual storage, virtual networking configuration, and any other software you might install within the virtual machine, including SQL Server.



The image illustrates the increased control you have using IaaS, compared to PaaS offerings. Note that while this diagram shows SaaS as a general cloud service model, Azure SQL offerings are either PaaS (such as Azure SQL Database and Azure SQL Managed Instance) or IaaS (SQL Server on Azure Virtual Machines).

In Azure SQL PaaS offerings, the administrator is responsible for database management, user security, and data management, while Microsoft manages the operating system and SQL Server software. When you use Azure SQL PaaS services, the operating system (OS) and SQL Server software are managed by the cloud provider. A good example of this is Azure SQL Database where the operating system and database engine are installed and configured by Microsoft, allowing you to start building database applications quickly. IaaS solutions are the most open-ended; you're responsible for OS patching, SQL Server installation and configuration, and optimal configuration of your network and storage options. With an IaaS deployment, you're also responsible for software configuration.

Some of your applications may not be suited for other Azure offerings, such as Azure SQL Database, because they require specific operating conditions. These conditions could include a specific combination of SQL Server and Windows versions for vendor support purposes, or other software that needs to be installed alongside of SQL Server. SQL Server paired with the Azure IaaS platform provides the required control options for many organizations, whether it be specific feature like CLR or replication, or the use of Active Directory (as opposed to Microsoft Entra ID) authentication. Another requirement is that some applications install software alongside SQL Server, which requires direct access to the underlying operating system. Direct access to the OS isn't supported in a PaaS model. These organizations and their applications can obtain the advantages of moving to a cloud service without losing critical capabilities that their organization requires.

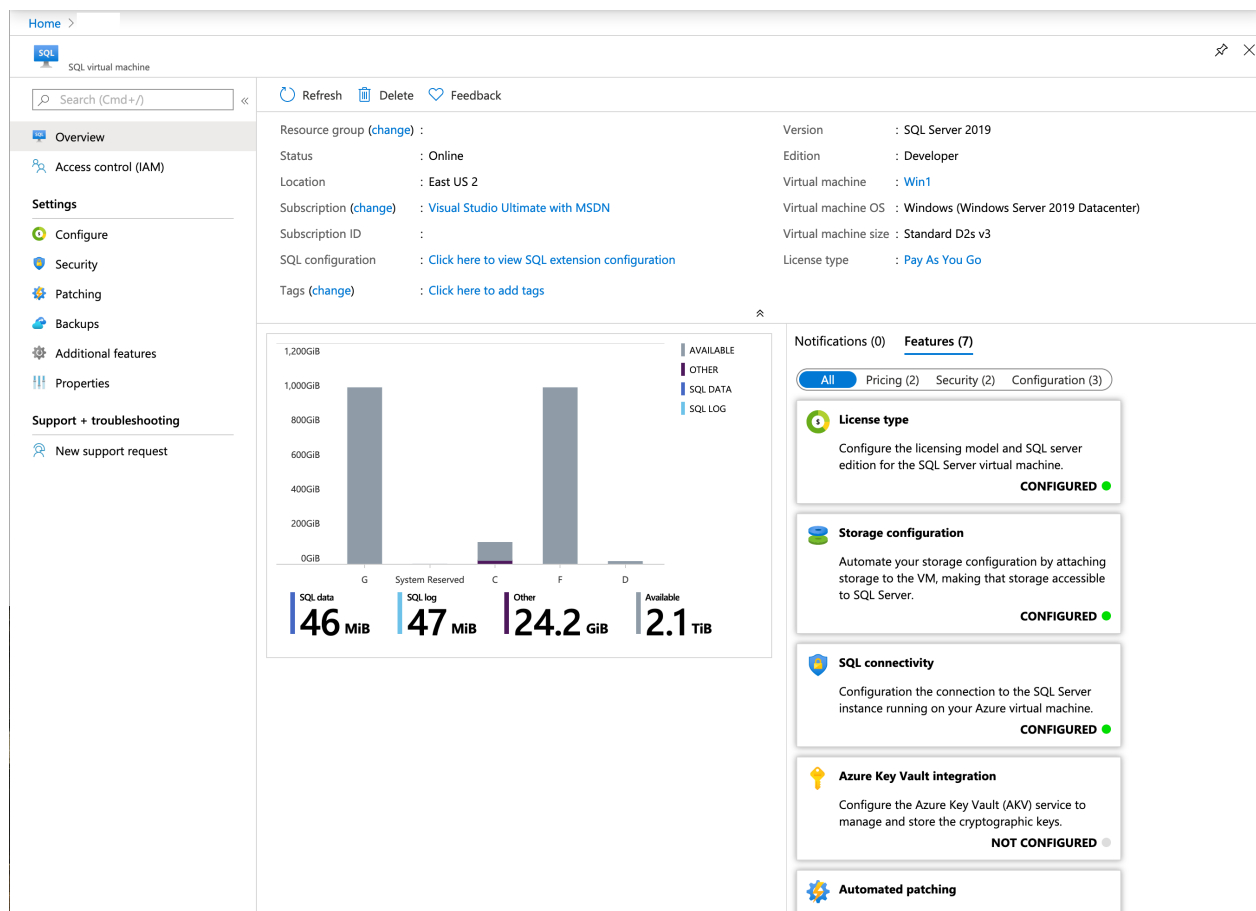
## SQL Server IaaS Agent Extension

When you deploy an SQL Server VM from Azure Marketplace, part of the process installs the IaaS Agent Extension.

Extensions are code that is executed on your VM post-deployment, typically to perform post-deployment configurations. Some examples include installing anti-virus software or enabling a Windows feature. The SQL Server IaaS Agent Extension provides the following main features that can reduce your administrative overhead.

- **Automated backup**
- **Automated patching**
- **Azure Key Vault integration**
- **Microsoft Defender for Cloud integration**
- **View Disk utilization in the portal**
- **Flexible licensing**
- **Flexible version or edition**
- **SQL best practices assessment**

In addition to these features, the extension allows you to view information about your SQL Server's configuration and storage utilization.



## SQL Server licensing models

There are several different options related to how SQL Server is licensed when using the Azure IaaS offering.

If you aren't participating in the Microsoft Software Assurance (SA) program, you can deploy an image from Azure Marketplace containing a preconfigured SQL Server, and pay-per-minute for the use of SQL Server. This option is referred to as the Pay as you Go model and the cost of the SQL Server license is included with the cost of the virtual machine.

If you're participating in the Microsoft Software Assurance (SA) program, you have more flexibility in how you license your SQL Server:

- You can use the previous method and pay-per-minute by deploying a virtual machine image containing a SQL Server from Azure Marketplace
- You can Bring Your Own License (BYOL) when deploying the virtual machine that doesn't contain a preconfigured SQL Server instance. This option is possible when you already have purchased a valid SQL Server license for your on-premises infrastructure. This license can be applied to the virtual machine to ensure that you're properly licensed. You must report the usage of licenses to Microsoft by using the License Mobility verification form within 10 days of implementing the virtual machine.

When choosing this method, you can manually install SQL Server through media you have obtained, or you can choose to upload a virtual machine image to Azure.

In addition to flexible licensing options for SQL Server, there are also Windows Server licensing options that can be taken advantage of. These Windows Server options are known as the Azure Hybrid Benefit (AHB). Similar to applying a SQL Server license you already have purchased, you're able to take advantage of Windows Server licenses you already own.

Reserving a virtual machine for one to three years provides another option for cost savings. This commitment doesn't require an upfront payment and can be billed monthly. Using the reservation option can be beneficial if you know the workloads are going to be persisted. The cost savings can be significant, especially for larger VMs.

## Virtual machine families

There are several series, or 'families,' of virtual machine sizes that you can choose. Each series is a combination of memory, CPU, and storage that meets certain requirements. For example, the series that are compute optimized have a higher CPU-to-memory ratio. Having multiple options allows you to select an appropriate hardware configuration for the expected workload. The following six series each has various sizes available, the details of which are fully described in the Azure portal when you choose the option to select your VM size.

**General purpose** - These VMs provide a balanced ratio of CPU to memory. This VM class is ideal for testing and development, small to medium-sized database servers, and web servers with a low to medium amount of traffic.

**Compute optimized** - Compute optimized VMs have a high CPU-to-memory ratio and are good for web servers with a medium amount of traffic, network appliances, batch processes, and application servers. These VMs can also support machine learning workloads that can't benefit from GPU-based VMs.

**Memory optimized** - These VMs provide a high memory-to-CPU ratio. These VMs cover a broad range of CPU and memory options (all the way up to 4 TB of RAM) and are well suited for most database workloads.

**Storage optimized** - Storage optimized VMs provide fast, local, NVMe storage that is ephemeral. They're good candidates for scale-out data workloads such as Cassandra. It's possible to use them with SQL Server; however, since the storage is ephemeral, you need to ensure you configure data protection using a feature like Always On Availability Groups or Log Shipping.

**GPU** - Azure VMs with GPUs are targeted at two main types of workloads—naturally graphics processing operations like video rendering and processing, but also massively parallel machine learning workloads that can take advantage of GPUs.

**FPGA accelerated** - These sizes are designed for compute-intensive workloads. Storage throughput and network bandwidth are also included for each size in this group.

**High performance compute** - High Performance Compute workloads support applications that can scale horizontally to thousands of CPU cores. This support is provided by high-performance CPU and remote direct memory access (RDMA) networking that provides low latency communications between VMs.

The easiest way to see the sizing options within each series is through the Azure portal. From the blade for creating a VM, you select the option **See all sizes** to see the list.

Select a VM size ...

vCPUs: All RAM (GiB): All Display cost: Monthly Add filter

Showing 791 VM sizes. | Subscription: WWL-JuPadrao | Region: West US 2 | Current size: Standard\_D16as\_v4 | Image: Free SQL Server License: SQL Server 2022 Developer on Windows Server 2022 [Learn more about VM sizes](#)

| SKU                                                                          | Type            | vCPUs | RAM (GiB) | Data disks | Max IOPS | Local storage (GiB) | Premium disk | Cost per month                   |
|------------------------------------------------------------------------------|-----------------|-------|-----------|------------|----------|---------------------|--------------|----------------------------------|
| Most used by Azure users                                                     |                 |       |           |            |          |                     |              |                                  |
| The most used sizes by users in Azure                                        |                 |       |           |            |          |                     |              |                                  |
| B1s ⓘ                                                                        | General purpose | 1     | 1         | 2          | 320      | 4                   | Supported    | \$10.51 <a href="#">Popular</a>  |
| B2ms ⓘ                                                                       | General purpose | 2     | 8         | 4          | 1920     | 16                  | Supported    | \$66.58 <a href="#">Popular</a>  |
| B2s ⓘ                                                                        | General purpose | 2     | 4         | 4          | 1280     | 8                   | Supported    | \$36.21 <a href="#">Popular</a>  |
| B4ms ⓘ                                                                       | General purpose | 4     | 16        | 8          | 2880     | 32                  | Supported    | \$132.86 <a href="#">Popular</a> |
| D2as_v4 ⓘ                                                                    | General purpose | 2     | 8         | 4          | 3200     | 16                  | Supported    | \$137.24 <a href="#">Popular</a> |
| D2s_v3 ⓘ                                                                     | General purpose | 2     | 8         | 4          | 3200     | 16                  | Supported    | \$137.24 <a href="#">Popular</a> |
| D4s_v3 ⓘ                                                                     | General purpose | 4     | 16        | 8          | 6400     | 32                  | Supported    | \$274.48 <a href="#">Popular</a> |
| D8s_v3 ⓘ                                                                     | General purpose | 8     | 32        | 16         | 12800    | 64                  | Supported    | \$548.96 <a href="#">Popular</a> |
| DS1_v2 ⓘ                                                                     | General purpose | 1     | 3.5       | 4          | 3200     | 7                   | Supported    | \$85.41 <a href="#">Popular</a>  |
| DS2_v2 ⓘ                                                                     | General purpose | 2     | 7         | 8          | 6400     | 14                  | Supported    | \$170.82 <a href="#">Popular</a> |
| DS3_v2 ⓘ                                                                     | General purpose | 4     | 14        | 16         | 12800    | 28                  | Supported    | \$341.64 <a href="#">Popular</a> |
| D-Series v5                                                                  |                 |       |           |            |          |                     |              |                                  |
| The 5th generation D family sizes recommended for your general purpose needs |                 |       |           |            |          |                     |              |                                  |
| D-Series v4                                                                  |                 |       |           |            |          |                     |              |                                  |
| The 4th generation D family sizes for your general purpose needs             |                 |       |           |            |          |                     |              |                                  |

The image above shows just a small set of the series and size possibilities. For each option, you can see the number of Virtual CPUs, the amount of RAM, the number of Data disks, the Max IOPS, the

temporary storage provided and whether Premium storage is supported.

For more information about VM size best practices, see [Best practices for SQL Server on Azure VMs](#).

## Azure Marketplace

Azure Marketplace is essentially a centralized location that provides the ability to create Azure resources based on a predesigned template. For example, you can quickly create a SQL Server 2022 instance on Windows Server 2019 with a couple of clicks, along with some basic information, such as the virtual machine name and SQL Server configuration details. Once provided, Azure Resource Manager initiates the creation of the virtual machine, and within minutes it's up and running.

The blade for SQL Server 2019 on Windows Server 2019 in Azure Marketplace is shown below. This blade gives you the option of preset configurations that support OLTP or Data Warehouse workloads and allows you to specify storage, patching, and backup options.

**SQL Server 2022 on Windows Server 2022**  ...

Microsoft

**SQL Server 2022 on Windows Server 2022**  [Add to Favorites](#)

Microsoft | Virtual Machine

Plan

Free SQL Server License: SQL Server 2...  [Create](#) [Start with a pre-set configuration](#)

Overview **Plans** Usage Information + Support Ratings + Reviews

| Software plan                                                             | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
|---------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Free SQL Server License: SQL Server 2022 Developer on Windows Server 2022 | This image contains the Developer edition of SQL Server 2022 on Windows Server 2022. This free edition (no SQL Server licensing cost) includes all the functionality of Enterprise edition, but it is licensed for development and testing only, not production. It provides comprehensive capabilities for mission-critical transactional processing, data warehousing, and real-time business intelligence. It includes the database engine with support for key features from previous versions including in-memory transactional processing, intelligent query processing, Always On availability groups for +99.99% high availability and read scale-out capabilities, as well as new features such as Azure Synapse Link for SQL, Link to Azure SQL Managed Instance, Ledger, and parameter sensitive plan optimization. Includes Integration Services for moving and transforming data, Analysis Services for data mining, and Reporting Services for web and mobile reports. We recommend that you use a virtual machine size of D8ds-v5 or higher for development and functional testing, E16bds-v5 or higher for performance testing.                                                      |
| SQL Server 2022 Web on Windows Server 2022                                | This image contains the Web edition of SQL Server 2022 on Windows Server 2022. This provides a low-cost database solution for medium-size web applications. It includes the core database engine and Management Studio for integrated administration and development. Also includes Integration Services for moving and transforming data, and Analysis Services for data mining. we recommend that you use a virtual machine size of E16ds-v4 or higher.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| SQL Server 2022 Enterprise on Windows Server 2022                         | This image contains the full version of SQL Server 2022 Enterprise edition on Windows Server 2022. SQL Server 2022 represents a major step towards making SQL Server a platform that gives you choices of development languages, data types, on-premises or cloud, and operating systems by bringing the power of SQL Server to Linux, Linux-based Docker containers, and Windows. It provides comprehensive capabilities for mission-critical transactional processing, data warehousing, and real-time business intelligence. It includes the database engine with support for In-Memory transactional processing and analytics, automatic database tuning and new graph capabilities for modeling many-to-many relationships. This edition also includes Always On for +99.99% high availability and read scale-out capabilities. Various layers of protection, including innovative features like Always Encrypted and Row-Level Security for the highest data protection. Includes Integration Services for moving and transforming data, Analysis Services for data mining and Master Data Services for data modeling. we recommend that you use a virtual machine size of E16ds-v4 or higher. |
| SQL Server 2022 Standard on Windows Server 2022                           | Breakthrough performance and faster insights delivered by a hybrid data platform. This image contains the Standard edition of SQL Server 2022 on Windows Server 2022. Standard edition provides core data management capabilities for medium-size transactional processing and data warehousing. It includes the database engine with a basic version of Always On high availability and Row-Level Security. Includes Management Studio for integrated administration and development. we recommend that you use a virtual machine size of E16as-v4 or higher.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |

The disadvantage of using the portal to create Azure resources is that it isn't an easily repeatable process. However, it's easy to get started with the portal, where you can quickly get up and running

with resources.

## SQL Server configuration

When you provision an Azure virtual machine running SQL Server, you can also configure specific SQL Server settings, such as Security and Networking, SQL Authentication preferences, SQL instance settings, and a few other options. These options are located on the **SQL Server settings** tab.

## Create a virtual machine ...

 [Help me create a low cost VM](#) [Help me create a VM optimized for high availability](#) [Help me choose the right VM size for my workload](#)

[Basics](#) [Disks](#) [Networking](#) [Management](#) [Monitoring](#) [Advanced](#) [SQL Server settings](#) [Tags](#) [Review + create](#)

### Security & Networking

SQL connectivity \*

Port \*

### SQL Authentication

SQL Authentication ⓘ ☒ Disable ☐ Enable

Azure Key Vault integration ⓘ ☒ Disable ☐ Enable

### Storage configuration

Customize performance, size, and workload type to optimize storage for this virtual machine. For optimal performance, separate drives will be created for data and log storage by default. [Learn more about SQL Server best performance practices.](#)

Storage 

SQL Data: 1024 GiB, 5000 IOPS, 200 MB/s, Premium SSD  
SQL Log: 1024 GiB, 5000 IOPS, 200 MB/s, Premium SSD  
SQL tempdb: Use local SSD drive  
[Change configuration](#)

### SQL instance settings

Customize additional SQL instance settings including collation, MAXDOP, server memory limit and optimize for ad-hoc workload.

Instance settings 

**Default configuration**  
MAXDOP: 0  
SQL Server memory limits: 0 - 2147483647 MB  
Collation: SQL\_Latin1\_General\_CP1\_CI\_AS  
[Change SQL instance settings](#)

### SQL Server License

Save up to 43% with licenses you already own. Already have a SQL Server license? [Learn more](#)

SQL Server License ⓘ ☒ No ☐ Yes

### Automated patching

Set a patching window during which all Windows and SQL patches will be applied.

Automated patching ⓘ 

**Disabled**  
[Change configuration](#)

### Automated backup

Automated backup ⓘ ☒ Disable ☐ Enable

[< Previous](#) [Next : Tags >](#) [Review + create](#)

For more information about the SQL Server settings available when creating a virtual machine, see [Provision SQL Server on Azure VM \(Azure portal\)](#).

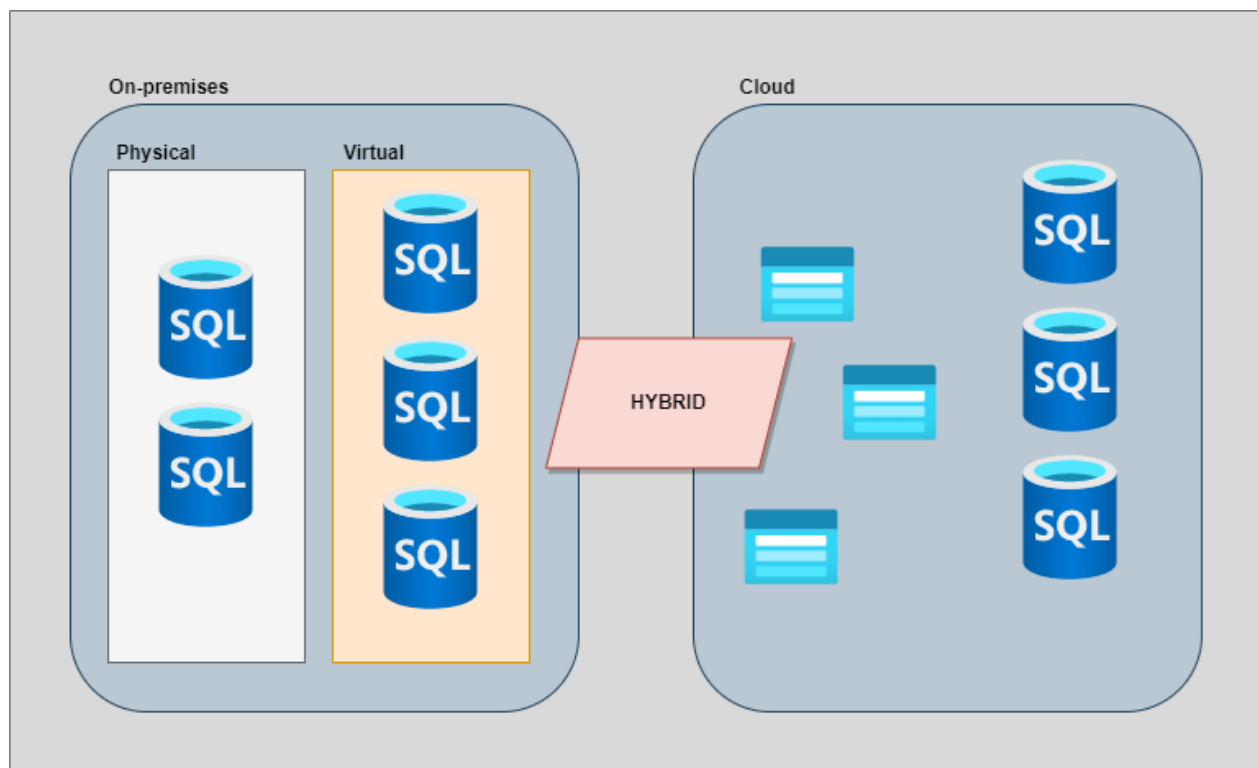
## 3. Understand hybrid scenarios

## Understand hybrid scenarios

Completed

Suppose your organization has a significant investment in SQL Server infrastructure on-premises and in local data centers. In that case, it's crucial to be aware that using the cloud isn't an all-or-nothing proposition. There are ways to use existing on-premises infrastructure in a hybrid capacity with Azure to improve operational resiliency and reduce costs.

Adopting a hybrid infrastructure is an excellent initial step for organizations that rely on on-premises solutions and are cautious about transitioning to the cloud. Many modern organizations use a combination of physical and virtualized SQL Server deployments on-premises, which can be extended to the cloud as part of a hybrid solution. The hybrid SQL Server platform combines the advantages of both on-premises and cloud services, serving as a complementary middle ground. Typically, the cloud component uses IaaS services such as storage or SQL Server virtual machines.



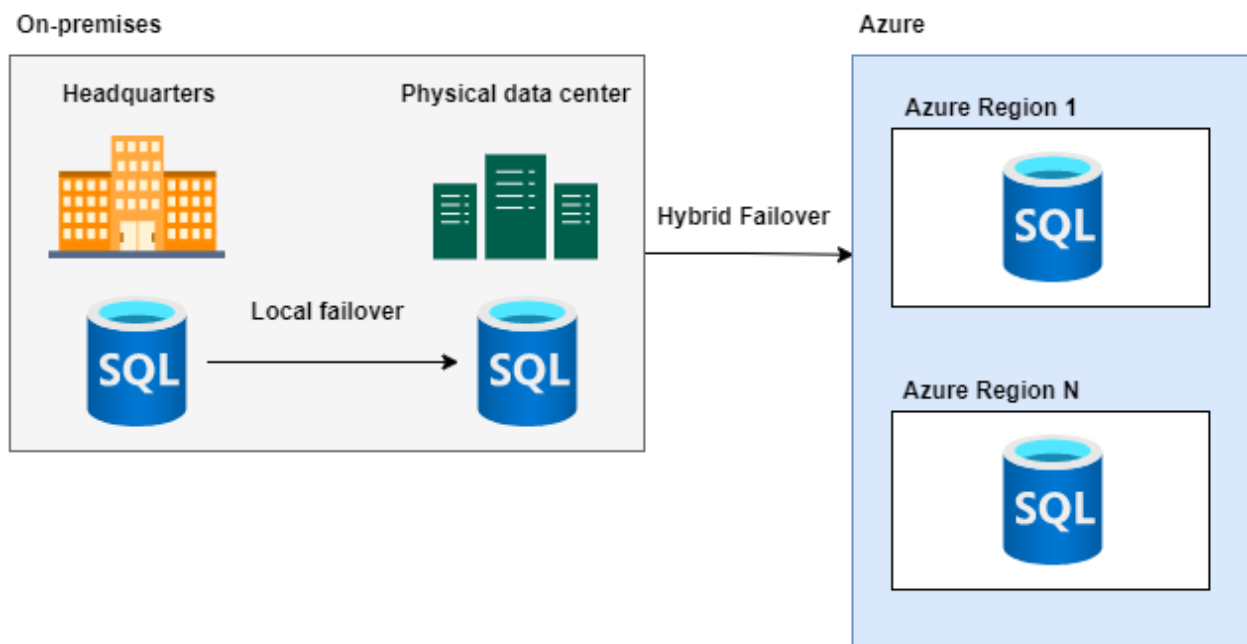
In addition to extending on-premises solutions, the same patterns apply to existing alternative cloud solutions, enabling cloud-to-cloud hybrid implementations. Let's review some of the most common hybrid scenarios for SQL Server.

### Hybrid scenarios for SQL Server

Let's review a few strategies for when deploying a hybrid solution for SQL Server.

## Disaster Recovery

Disaster Recovery is the most common scenario for a hybrid deployment of SQL Server. It ensures business continuity if catastrophic events occur. Organizations can distribute deployments across multiple data centers for failover in an on-premises approach. These data centers typically reside within the same geographic region as the organization, making them susceptible to significant regional disasters. Physical data centers are also costly to deploy, monitor, and maintain. Arguably, the cost of spinning up Azure virtual machines running SQL Server in various geographical regions is much less than establishing a new physical data center in another geography.



In this hybrid approach, Azure is used for [DR failover](#) (to one or more regions) while the regular day-to-day processing continues to use on-premises servers for local high availability.

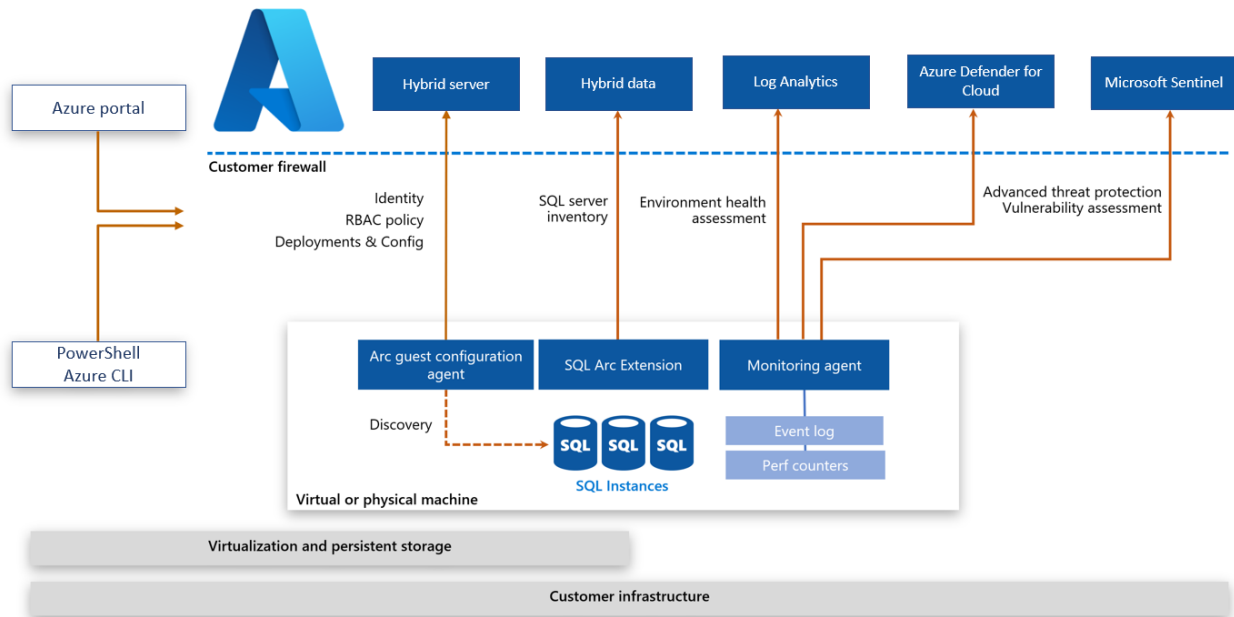
## SQL Server Backups

SQL Server Backups is another common hybrid scenario. Backups may be done directly into Azure Storage via URL or Azure file share (SMB). This scenario protects against data loss when on-site backup storage fails. In addition, these backups may also be restored to virtual machines in Azure and tested as part of Disaster Recovery procedures.

Another scenario uses Azure Storage to store on-premises **SQL Server data files for user databases**. These are user files and not system databases. If the local storage fails, the user files are safely stored in the cloud, preventing data loss. In addition, by using Azure storage, there are [built-in reliability guarantees](#) so storing these files in the cloud is more resilient. For this hybrid scenario, it's essential to keep the network communication secure, evaluate the network latency of the solution, and ensure the storage account is locked down using ACLs and Microsoft Entra ID.

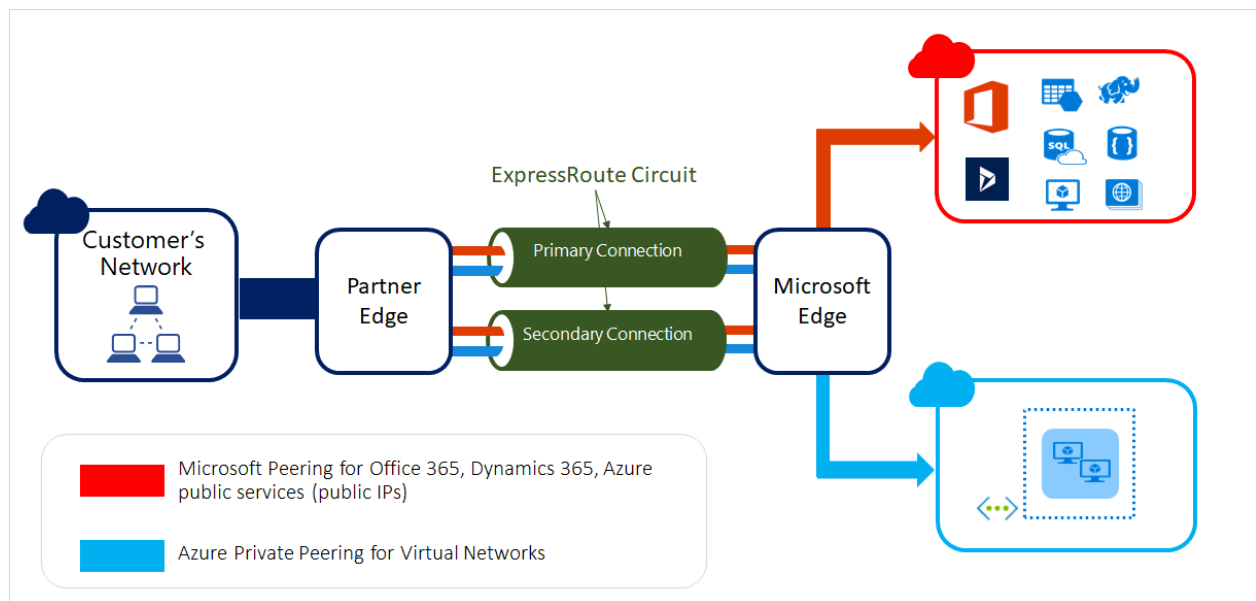
## Azure Arc-enabled SQL Servers

Having **Azure Arc-enabled SQL Servers** extends and centralizes Azure management services to SQL Server instances hosted on-premises, in your data centers, on the edge, and in multicloud environments. In this hybrid scenario, Azure Arc enables the inventory of all registered SQL Server deployments and assesses their configurations, usage patterns, and security to provide actions and recommendations based on best practices. By using **SQL Server enabled by Azure Arc**, you gain the benefits of centralized server management. You also get real-time security alerts and vulnerability reporting on both on-premises SQL Servers and their host operating systems.



## Security considerations

When deploying a hybrid SQL solution, all core infrastructure, such as Active Directory and DNS, must exist on-premises and in Azure. In addition, secure two-way communication must exist between the on-premises network and Azure. This secured communication can take the form of a **site-to-site (S2S) VPN** or a dedicated **ExpressRoute** tunnel. When evaluating different connectivity methods, it's vital to determine the amount of latency acceptable for your organization. Regardless of the solution chosen, network security must also be at the forefront of the implementation.



The ExpressRoute connections tend to be more costly, but they provide the best security and lowest latency as all communication flows over a direct secured channel independent of the public internet. However, common drawbacks for this solution include overall cost and the inability to apply ExpressRoute between cloud providers in a multicloud solution.

## 4. Explore performance and security

<https://learn.microsoft.com/en-us/training/modules/deploy-iaas-solutions-with-azure-sql/4-explore-performance-and-security>

# Explore performance and security

Completed

The Azure ecosystem offers several performance and security options for SQL Server instances on Azure virtual machines. Each option provides various capabilities, such as different disk types that meet the capacity and performance requirements of your workload.

## Storage considerations

SQL Server requires good storage performance to deliver robust application performance, whether it be an on-premises instance or installed in an Azure VM. Azure provides a wide variety of storage solutions to meet the needs of your workload. While Azure offers various types of storage (blob, file, queue, table), in most cases SQL Server workloads will use Azure managed disks. The exceptions are that a Failover Cluster Instance can be built on file storage and backups will use blob storage. Azure-

managed disks act as a block-level storage device that is presented to your Azure VM. Managed disks offer a number of benefits including 99.999% availability, scalable deployment (you can have up to 50,000 VM disks per subscription per region), and integration with availability sets and zones to offer higher levels of resiliency in case of failure.

Azure-managed disks all offer two types of encryption. Azure Server-side encryption is provided by the storage service and acts as encryption-at-rest provided by the storage service. Azure Disk Encryption uses BitLocker on Windows, and DM-Crypt on Linux to provide OS and Data disk encryption inside of the VM. Both technologies integrate with Azure Key Vault and allow you to bring your own encryption key.

Each VM will have at least two disks associated with it:

- **Operating System disk** – Each virtual machine will require an operating system disk that contains the boot volume. This disk would be the C: drive in the case of a Windows platform virtual machine, or /dev/sda1 on Linux. The operating system will be automatically installed on the operating system disk.
- **Temporary disk** – Each virtual machine will include one disk used for temporary storage. This storage is intended to be used for data that doesn't need to be durable, such as page files or swap files. Because the disk is temporary, you shouldn't use it for storing any critical information like database or transaction log files as they'll be lost during maintenance or a reboot of the virtual machine. This drive will be mounted as D:\ on Windows, and /dev/sdb1 on Linux.

Additionally, you can and should add additional data disks to your Azure VMs running SQL Server.

- **Data disks** – The term data disk is used in the Azure portal, but in practice these are just additional managed disks added to a VM. These disks can be pooled to increase the available IOPS and storage capacity, using Storage Spaces on Windows or Logical Volume Management on Linux.

Furthermore, each disk can be one of several types:

| Feature        | Ultra Disk             | Premium SSD v2                  | Premium SSD                     | Standard SSD          | Standard HDD                    |
|----------------|------------------------|---------------------------------|---------------------------------|-----------------------|---------------------------------|
| Disk type      | SSD                    | SSD                             | SSD                             | SSD                   | HDD                             |
| Best for       | IO-intensive workloads | Performance-sensitive workloads | Performance-sensitive workloads | Lightweight workloads | Backups, non-critical workloads |
| Max disk size  | 65,536 GiB             | 64,000 GiB                      | 32,767 GiB                      | 32,767 GiB            | 32,767 GiB                      |
| Max throughput | 10,000 MB/s            | 1,200 MB/s                      | 900 MB/s                        | 750 MB/s              | 500 MB/s                        |
| Max IOPS       | 160,000                | 80,000                          | 20,000                          | 6,000                 | 2,000                           |

The best practices for SQL Server on Azure recommend using Premium Disks pooled for increased IOPs and storage capacity. Data files should be stored in their own pool with read-caching on the Azure disks.

Transaction log files won't benefit from this caching, so those files should go into their own pool without caching. TempDB can optionally go into its own pool, or using the VM's temporary disk, which offers low latency since it's physically attached to the physical server where the VMs are running. Properly configured Premium SSD will see latency in single digit milliseconds. For mission critical workloads that require latency lower than that, you should consider Ultra SSD.

## Security considerations

There are several regulations and standards that Azure complies with that makes it possible to build a compliant solution with SQL Server running in a virtual machine.

### Microsoft Defender for SQL

[Microsoft Defender for SQL](#) provides Microsoft Defender for Cloud security features such as vulnerability assessments and security alerts.

Azure Defender for SQL can be used to identify and mitigate potential vulnerabilities in your SQL Server instance and database. The vulnerability assessment feature can detect potential risks in your SQL Server environment and help you remediate them. It also provides insight into your security state and actionable steps to resolve security issues.

### Microsoft Defender for Cloud

Microsoft Defender for Cloud is a unified security management system that evaluates and offers opportunities for improving several security aspects of your data environment. Microsoft Defender for Cloud is a security management tool that allows you to gain insight into your security state across hybrid cloud workloads, reduce your exposure to attacks, and respond to detected threats quickly.

## Performance considerations

Most of the existing on-premises SQL Server performance features are also available on Azure virtual machines (VMs). Among the options offered is data compression, which can improve the performance of I/O-intensive workloads while decreasing the size of the database. Similarly, table and index partitioning can improve query performance of large tables, while improving performance and scalability.

### Table partitioning

[Table partitioning](#) provides many benefits, but often this strategy is only considered when the table becomes large enough that starts compromising query performance. Identifying which tables are candidates for table partitioning is a good practice that could lead to fewer disruptions and

interventions. When you filter your data using your partition column, only a subset of the data is accessed, not the entire table. Similarly, maintenance operations on a partitioned table will reduce maintenance duration, for example, by compressing specific data in a particular partition or rebuilding specific partitions of an index.

There are four main steps required when defining a table partition:

- The filegroups creation, which defines the files involved when the partitions are created.
- The partition function creation, which defines the partition rules based on the specified column.
- The partition scheme creation, which defines the filegroup of each partition.
- The table to be partitioned.

The example below illustrates how to create a partition function for January 1, 2021 through December 1, 2021, and distribute the partitions across different filegroups.

```
-- Partition function
CREATE PARTITION FUNCTION PartitionByMonth (datetime2)
AS RANGE RIGHT
-- The boundary values defined is the first day of each month, where the table
FOR VALUES ('20210101', '20210201', '20210301',
            '20210401', '20210501', '20210601', '20210701',
            '20210801', '20210901', '20211001', '20211101',
            '20211201');

-- The partition scheme below will use the partition function created above, and as
CREATE PARTITION SCHEME PartitionByMonthSch
AS PARTITION PartitionByMonth
TO (FILEGROUP1, FILEGROUP2, FILEGROUP3, FILEGROUP4,
    FILEGROUP5, FILEGROUP6, FILEGROUP7, FILEGROUP8,
    FILEGROUP9, FILEGROUP10, FILEGROUP11, FILEGROUP12);

-- Creates a partitioned table called Order that applies PartitionByMonthSch partiti
CREATE TABLE Order ([Id] int PRIMARY KEY, OrderDate datetime2)
ON PartitionByMonthSch (OrderDate) ;
GO
```

## Data compression

SQL Server offers different options of [data compression](#). While SQL Server still stores compressed data on 8 KB pages, when the data is compressed, more rows of data can be stored on a given page, which allows the query to read fewer pages. Reading fewer pages has a twofold benefit: it reduces the amount of physical IO performed and it allows more rows to be stored in the buffer pool, making more efficient use of memory. We recommend enabling database page compression where appropriate.

The tradeoffs to compression are that it does require a small amount of CPU overhead; however, in most cases, the storage IO benefits far outweigh any additional processor usage.

```
SQLQuery6.sql - dca...14 (jdantoni (107))* X SQLQuery4.sql - VM...VM1\jdantoni (78)) SQ

SELECT COUNT(*)
FROM Production.TransactionHistory
WHERE TransactionDate > '2008-01-01'

SELECT COUNT(*)
FROM Production.TransactionHistory_Page
WHERE TransactionDate > '2008-01-01'

100 %
Results Messages

(1 row affected)
Table 'TransactionHistory'. Scan count 1, logical reads 992, physical read

(1 row affected)
Table 'TransactionHistory_Page'. Scan count 1, logical reads 273, physical

Completion time: 2020-04-22T14:55:13.3744311+00:00
```

The image above shows this performance benefit. These tables have the same underlying indexes; the only difference is that the clustered and nonclustered indexes on the *Production.TransactionHistory\_Page* table are page compressed. The query against the page compressed object performs 72% fewer logical reads than the query that uses the uncompressed objects.

Compression is implemented in SQL Server at the object level. Each index or table can be compressed individually, and you have the option of compressing partitions within a partitioned table or index. You can evaluate how much space you'll save by using the `sp_estimate_data_compression_savings` system stored procedure. Prior to SQL Server 2019, this procedure didn't support columnstore indexes or columnstore archival compression.

- **Row compression** - Row compression is fairly basic and doesn't incur much overhead; however, it doesn't offer the same amount of compression (measured by the percentage reduction in storage space required) that page compression may offer. Row compression basically stores each value in each column in a row in the minimum amount of space needed to store that value. It uses a variable-length storage format for numeric data types like integer, float, and decimal, and it stores fixed-length character strings using variable length format.
- **Page compression** - Page compression is a superset of row compression, as all pages will initially be row compressed prior to applying the page compression. Then a combination of techniques called prefix and dictionary compression are applied to the data. Prefix

compression eliminates redundant data in a single column, storing pointers back to the page header. After that step, dictionary compression searches for repeated values on a page and replaces them with pointers, further reducing storage. The more redundancy in your data, the greater the space savings when you compress your data.

- **Columnstore archival compression** - Columnstore objects are always compressed; however, they can be further compressed using archival compression, which uses the Microsoft *XPRESS* compression algorithm on the data. This type of compression is best used for data that is infrequently read but needs to be retained for regulatory or business reasons. While this data is further compressed, the CPU cost of decompression tends to outweigh any performance gains from IO reduction.

## Additional options

Below is a list of additional SQL Server features and actions to consider for production workloads:

- Enable backup compression
- Enable instant file initialization for data files
- Limit autogrowth of the database
- Disable autoshrink/autoclose for the databases
- Move all databases to data disks, including system databases
- Move SQL Server error log and trace file directories to data disks
- Set max SQL Server memory limit
- Enable lock pages in memory
- Enable optimize for adhoc workloads for OLTP heavy environments
- Enable Query Store.
- Schedule SQL Server Agent jobs to run DBCC CHECKDB, index reorganize, index rebuild, and update statistics jobs
- Monitor and manage the health and size of the transaction log files

For more information about performance best practices, see [Best practices for SQL Server on Azure VMs](#).

## 5. Explain high availability and disaster recovery options

<https://learn.microsoft.com/en-us/training/modules/deploy-iaas-solutions-with-azure-sql/5-explain-high-availability-and-disaster-recovery-options>

# Explain high availability and disaster recovery options

---

Completed

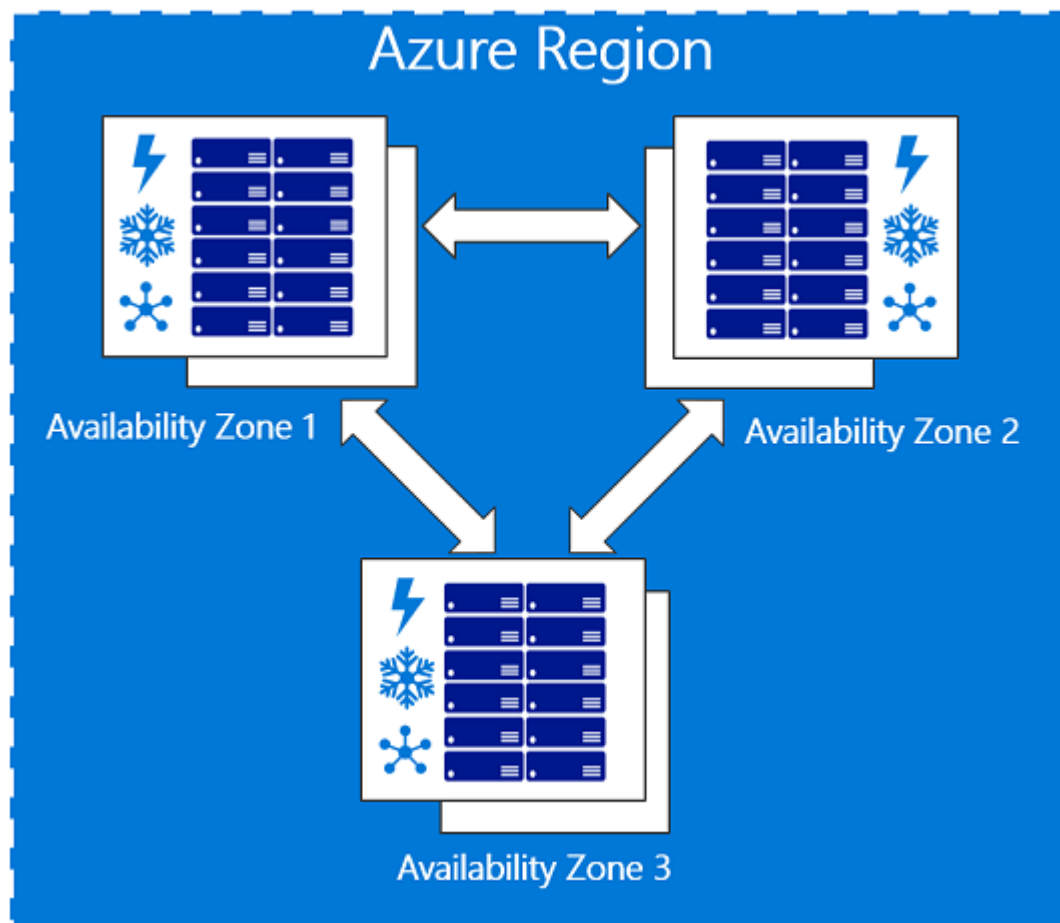
Beyond its built-in high availability, the Azure platform offers two options for providing higher levels of availability for VM and some PaaS workloads. Availability Zones and Availability Sets protect your workloads from planned maintenance activity and potential hardware failures.

## High availability options

Most SQL Server high availability solutions are available on Azure virtual machines (VMs). In an Azure-only solution, the entire HADR system runs in Azure. In a hybrid configuration, part of the solution runs in Azure and the other part runs on-premises in your organization. The flexibility of the Azure environment enables you to move partially or completely to Azure to satisfy the budget and HADR requirements of your SQL Server database systems.

### Availability Zones

Availability Zones are unique physical locations within a region. Each zone is made up of one or more data centers equipped with independent power, cooling, and networking. Within Azure regions that support Availability Zones, you can specify in which zone you want the virtual machine to reside when you choose to use Available Zones during VM creation. There are three Availability Zones within each supported Azure region. Availability Zones provide high availability against data center failures when you deploy multiple VMs into different zones. In addition, they also provide a means for Microsoft to perform maintenance (using a grouping called an update domain) within each region by only updating one zone at any given time. You can spread out your virtual machine ecosystem across three zones in the region. Utilizing Availability Zones in conjunction with your Azure virtual machines raises your uptime to four nines (99.99%) which equates to a maximum of 52.60 minutes of downtime per year. You can identify which Azure regions support Availability Zones at [docs.microsoft.com](https://docs.microsoft.com). If Availability Zones are available in your region, and your application can support the minimal cross-zone latency, Availability Zones will provide the highest level of availability for your application.



When you deploy a VM into a region with an availability zone you have the option to deploy in Zone 1, 2, and 3. These zones are logical representations of physical data centers, which means a deployment to Zone 1 in one subscription, does not mean that Zone 1 represents the same data center in another subscription.

## Availability Sets

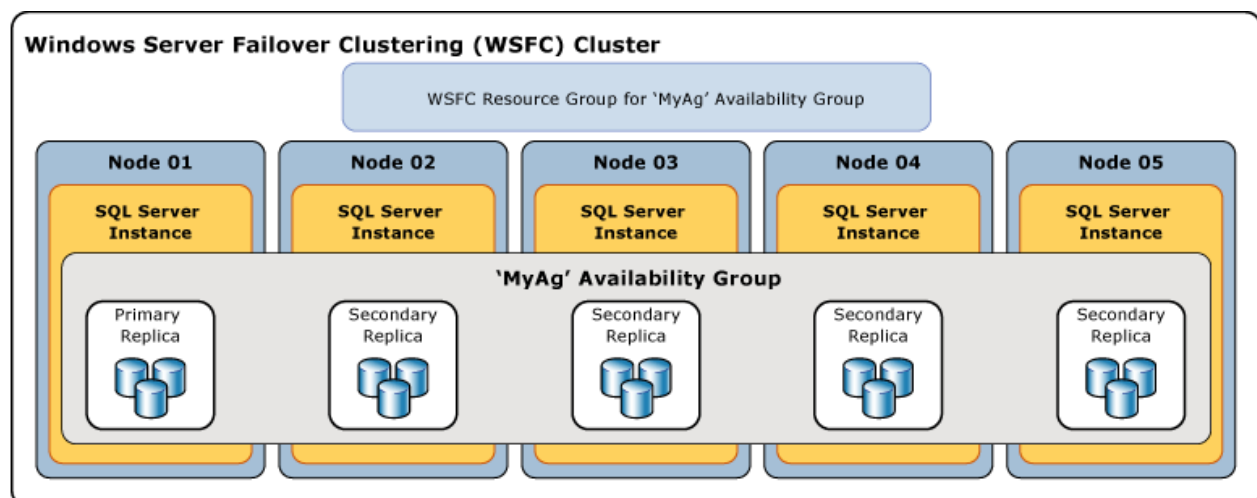
Availability sets are similar to Availability Zones, except instead of spreading workloads across data centers in a region, they spread workloads across servers and racks in a data center. Since nearly all workloads in Azure are virtual, you can use availability sets in order to guarantee that the two VMs containing your Always On Availability Group members are not running on the same physical host. Availability sets can provide up to 99.95% availability, and should be used when Availability Zones are unavailable in a region, or an application cannot tolerate intra-zone latency.

## Always On availability groups (AG)

Always On availability groups can be implemented between two or more (up to a maximum of nine) SQL Server instances running on Azure virtual machines or across an on-premises data center and Azure. In an availability group, database transactions are committed to the primary replica, and then the transactions are sent either synchronously or asynchronously to all secondary replicas. The physical distance between the servers (that is, whether or not they are in the same Azure region) dictates which availability mode you should choose. Generally, if the workload requires the lowest

possible latency or the secondary replicas are geographically spread apart, asynchronous availability mode is recommended. If the replicas are within the same Azure region and the applications can withstand some level of latency, synchronous commit mode should be considered. Synchronous mode will help to ensure that each transaction is committed to one or more secondaries before allowing the application to continue. Always On availability groups provide both high availability and disaster recovery, because a single availability group can support both synchronous and asynchronous availability modes. The unit of failover for an availability group is a group of databases, and not the entire instance.

Always On Availability Groups can also be used for disaster recovery purposes. You can implement up to nine replicas of a database across Azure regions, and stretch this architecture even further using Distributed Availability Groups. Availability Groups ensure that a viable copy of your database(s) is in another location beyond the primary region. By doing so, you help to ensure that your data ecosystem is protected against natural disasters as well as some human made ones.



The image above shows a logical diagram of an Always On Availability Group, running on a Windows Server Failover Cluster. There are one primary and four secondary replicas. In this scenario all five replicas could be synchronous, or some combination of synchronous and asynchronous replicas. Keep in mind that the unit of failover is the group of databases and not the instance. While a failover cluster instance provides HA at an instance level, it doesn't provide disaster recovery.

## SQL Server Failover Cluster instances

If you need to protect the entire instance, you could use a SQL Server Failover Cluster Instance (FCI), which provides high availability for an entire instance, in a single region. An FCI doesn't provide disaster recovery without being combined with another feature like availability groups or log shipping. FCIs also require shared storage that can be provided on Azure by using shared file storage or using Storage Spaces Direct on Windows Server.

For Azure workloads, availability groups are the preferred solution for newer deployments, because the shared storage required of FCIs increases the complexity of deployments. However, for migrations from on-premises solutions, an FCI may be required for application support.

## Disaster Recovery options

While the Azure platform offers 99.9% up time by default, disasters can still occur and affect application uptime. It's important that you have a proper disaster recovery plan in place when you are performing any type of migration. Azure offers us several methods to ensure that your SQL Server on a virtual machine is protected in case of a disaster. There are two components to this protection. First, there are Azure platform options like geo-replicated storage for backups and Azure Site Recovery, which is an all-encompassing disaster recovery solution for all of your workloads. Second, there are SQL Server specific offerings like Availability Groups and backups.

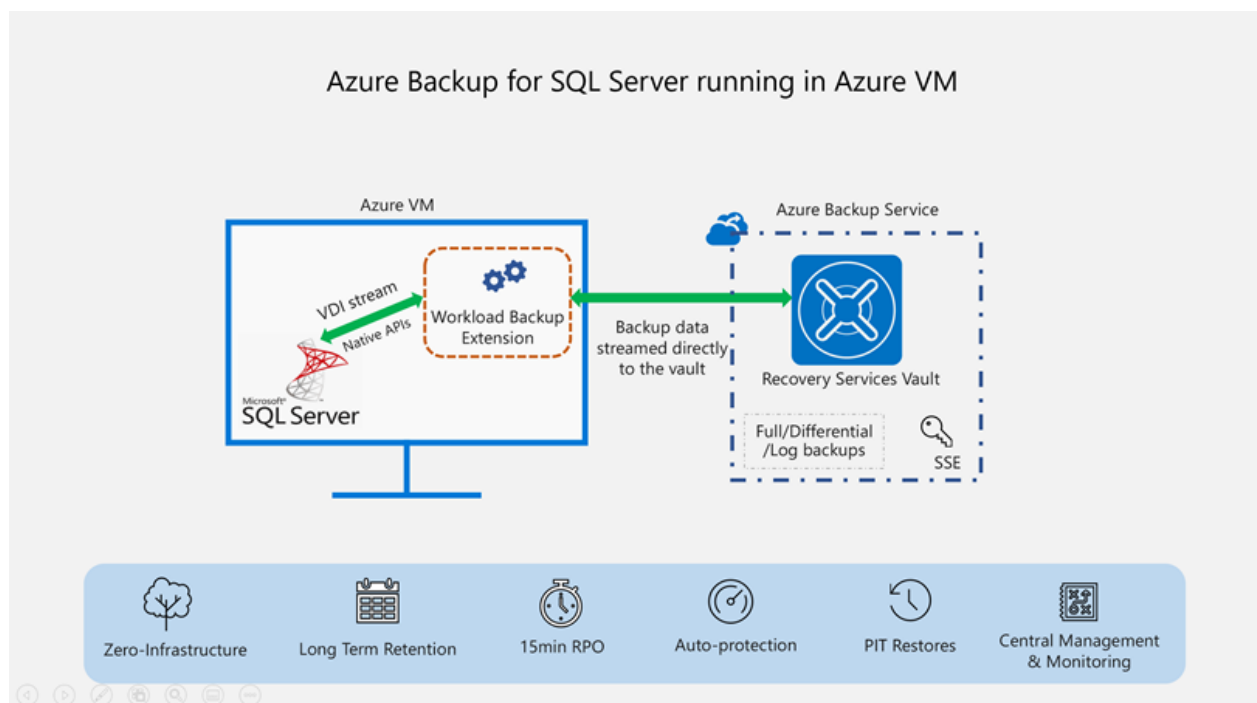
### Native SQL Server backups

Backups are considered the life blood of any database administrator and it's not any different when working with a cloud solution. With SQL Server on an Azure virtual machine, you have granular control of when backups occur and where they're stored. You can use SQL agent jobs to back up directly to a URL linked to Azure blob storage. Azure provides the option to use geo-redundant storage (GRS) or read-access geo-redundant storage (RA-GRS) to ensure that your backup files are stored safely across the geographic landscape.

Additionally as part of the Azure SQL VM service provider, you can have your backups automatically managed by the platform.

### Azure Backup for SQL Server

The Azure Backup solution requires an agent to be installed on the virtual machine. The agent then communicates with an Azure service that manages automatic backups of your SQL Server databases. Azure Backup also provides a central location that you can use to manage and monitor the backups to ensure meeting any specified RPO/RTO metrics.

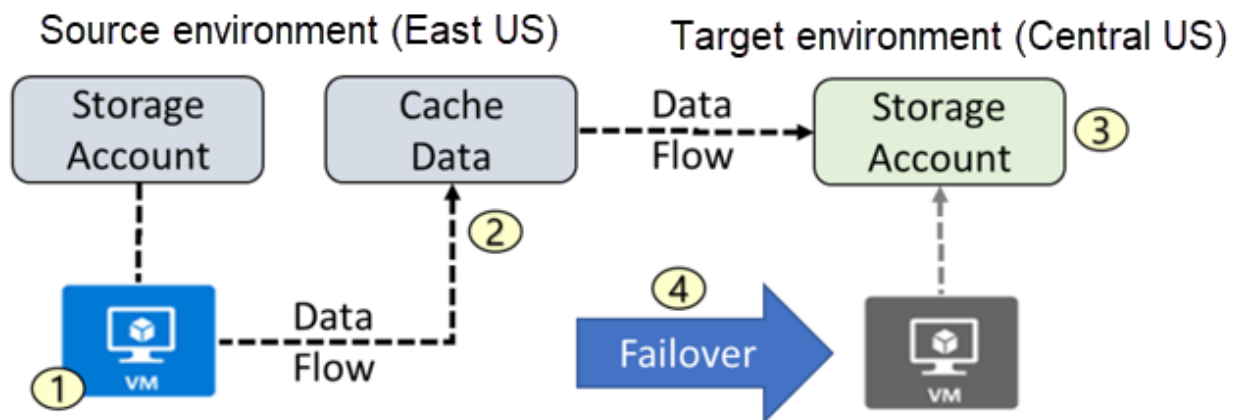


As shown above, the Azure Backup solution is a comprehensive, enterprise backup solution that provides long-term data retention, automated management, and additional data protection. This option costs more than simply performing your own backups, or using the Azure resource provider for SQL Server, but does offer a more complete backup feature set.

## Azure Site Recovery

Azure Site Recovery is a low-cost solution that will perform block level replication of your Azure virtual machine. This service offers various options, including the ability to test and verify your disaster recovery strategy. This solution is best used for stateless environments (for example, web servers) versus transactional database virtual machines.

Azure Site Recovery is supported for use with SQL Server, but keep in mind that you will need to set a higher recovery point which means potential loss. In this case, your RTO will essentially be your RPO.



1. VM is registered with Azure Site Recovery
2. Data is continuously replicated to cache
3. Cache is replicated to the target storage account
4. During failover the virtual machine is added to the target environment

## 6. Exercise: Provision a SQL Server on an Azure Virtual Machine

<https://learn.microsoft.com/en-us/training/modules/deploy-iaas-solutions-with-azure-sql/6-exercise-provision-sql-server-azure-virtual-machine>

## Exercise: Provision a SQL Server on an Azure Virtual Machine

Completed

Now it's your chance to deploy a SQL Server on an Azure Virtual Machine.

In this exercise, you'll explore the Azure portal and use it to create an Azure VM with SQL Server installed. Then they'll connect to the virtual machine through Remote Desktop Protocol.

#### Note

To complete this exercise, you'll need a Microsoft Azure subscription. If you don't already have one, you can sign up for a free trial at <https://azure.com/free>.

Launch the exercise and follow the instructions.

**Launch Exercise**

## 7. Module assessment

<https://learn.microsoft.com/en-us/training/modules/deploy-iaas-solutions-with-azure-sql/7-knowledge-check>

# Module assessment

---

Completed

## 8. Summary

<https://learn.microsoft.com/en-us/training/modules/deploy-iaas-solutions-with-azure-sql/8-summary>

# Summary

---

Completed

Azure provides a great deal of flexibility for deploying SQL Server to an Azure virtual machine. The hybrid licensing options reduce your cost. It also allows for additional protection for high availability and disaster recovery as part of your software assurance benefits. Azure Resource Manager offers many ways to deploy your resources in both a programmatic and graphical fashion.

Now that you've reviewed this module, you should be able to:

- Explore the basics of SQL Server in an Infrastructure as a Service (IaaS) offering
- Learn how hybrid scenario works
- Explore performance and security options available
- Understand the high availability and disaster recovery options