

Prepare to maintain SQL Server-based databases on Azure

1. Introduction

<https://learn.microsoft.com/en-us/training/modules/prepare-to-maintain-sql-databases-azure/1-introduction>

Introduction

Completed

Understanding Azure Database Administrator responsibilities, and how they relate to other Microsoft Intelligent Data Platform roles, is important to performing the tasks and duties expected for this function.

In general, cloud services can be broken down into two sets of services: Infrastructure as a Service (IaaS) and Platform as a Service (PaaS).

IaaS services typically consist of virtual machines, storage, and virtual networking components, and are largely managed by the user in terms of patching and software. Alternatively, PaaS services have a larger percentage of management tasks, which are handled by the cloud provider.

Learning objectives

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

- Understand the role of Azure Database Administrator, and other data platform roles
- Describe the key differences between the SQL Server-based database options in Azure
- Describe other features for Azure SQL platforms available

2. Describe Microsoft Intelligent Data Platform roles

Describe Microsoft Intelligent Data Platform roles

Completed

The role of Azure Database Administrator is just one of the roles that can be filled by professionals working with the Microsoft Intelligent Data Platform services. There are five different role-based certifications offered by Microsoft for people whose main job responsibilities deal with cloud-focussed data.

Role	Definition
Azure Data Engineer	Design and implement the management, monitoring, security, and privacy of data using the full stack of Azure data services to satisfy business needs.
Azure Data Analyst	Enable businesses to maximize the value of their data assets by using Microsoft Power BI. As a subject matter expert, Data Analysts are responsible for designing and building scalable data models, cleaning and transforming data, and enabling advanced analytic capabilities that provide meaningful business value through easy-to-comprehend data visualizations.
Azure Data Scientist	Applies their knowledge of data science and machine learning to implement and run machine learning workloads on Azure; in particular, using Azure Machine Learning Service.
Azure Artificial Intelligence Engineer	Use Cognitive Services, Machine Learning, and Knowledge Mining to architect and implement Microsoft AI solutions involving natural language processing, speech, computer vision, bots, and agents.
Azure Database Administrator	Implements and manages the operational aspects of cloud-native and hybrid data platform solutions built on Microsoft Azure data services and Microsoft SQL Server. The Azure Database Administrator uses a variety of methods and tools to perform day-to-day operations, including applying knowledge of using T-SQL for administrative management purposes.

3. Understand SQL Server in an Azure virtual machine

Understand SQL Server in an Azure virtual machine

Completed

A SQL Server running in an Azure virtual machine (IaaS) is equivalent to an on-premises SQL Server. You'll notice that several features described for SQL Server on Azure virtual machine are applicable to all your on-premises SQL Servers.

Many applications require SQL Server running on a virtual machine. The reasons include:

- **General application support and incompatibility** - For applications requiring an older version of SQL Server for vendor support. In addition, some application services may have a requirement to be installed with the database instance in a manner that isn't compatible with a PaaS offering.
- **Use of other SQL Server Services** - In order to maximize licensing, many users choose to run SQL Server Analysis Services (SSAS), SQL Server Integration Services (SSIS), and/or SQL Server Reporting Services (SSRS) on the same machine as the database engine.

Versions of SQL Server available

Microsoft keeps images of all supported versions of SQL Server available in the Azure Marketplace. If you require an older version that is covered by an extended support contract, you'll need to install your own SQL Server binaries.

Backup solutions

In recent releases of SQL Server, Microsoft has introduced several features to support running SQL Server in an Azure virtual machine. We're going to focus on two key backup features:

- Back up to URL
- Azure Backup

The back up to URL option enables you to back up your databases to Azure Blob Storage service. Azure Backup for SQL Server Virtual Machines provides a comprehensive enterprise backup solution that automatically manages your backups across your entire infrastructure.

Deployment options

All resources in Azure are managed and deployed through a common provider known as Azure Resource Manager. While there are various methods to deploy Azure resources, they ultimately converge into JSON documents called Azure Resource Manager templates, which serve as one of the deployment options for Azure resources.

The key distinction between these methods is that Azure Resource Manager templates use a declarative deployment approach, which defines the desired structure and state of the resources to be deployed. In contrast, other methods are imperative, using procedural models to explicitly specify the steps to be executed. For large-scale deployments, the declarative approach is preferable and should be adopted.

Overview of Azure storage

Azure offers a fully redundant object-based storage model, and there are a few things to be aware of in designing and deploying Virtual Machines architecture. In terms of virtual machines, there are four types of storage you can use:

- Standard
- Standard SSD
- Premium SSD
- Ultra Disk

For production SQL Server data and transaction log files, you should only use Premium SSD storage and Ultra Disk. With premium storage, you see latencies in the range of 5-10 ms on a properly configured system. Alternatively, with Ultra Disk you may have sub millisecond latency but will likely see 1-2 ms workloads in the real world. You can use Standard storage for your database backups, as the performance is adequate for most backup and restore workloads.

High availability in Azure

The Azure platform is designed to be fault tolerant and provides quick recovery from service disruptions and transient errors. In fact, many organizations see higher levels of availability in single virtual machines deployments than they previously experienced in their on-premises environments. Microsoft guarantees uptime of at least 99.9% for single instance Azure virtual machine, when using Premium SSD or Ultra Disk for all disks.

Azure offers several features to support high availability including availability sets, availability zones, and load-balancing techniques that provide high availability by distributing incoming traffic among Virtual Machines.

SQL Server enabled by Azure Arc

Azure Arc extends Azure management capabilities to SQL Server instances running outside of Azure, whether they're on-premises, in other clouds, or at the edge. By enabling SQL Server with Azure Arc, you can bring the benefits of Azure's cloud management and governance to your existing SQL Server

deployments without needing to move them to Azure. This includes applying consistent policies, ensuring compliance, and using Azure services such as Azure Monitor and Azure Security Center for enhanced security and performance monitoring.

With Azure Arc, you can centrally manage and monitor your SQL Server instances through the Azure portal just like you would with native Azure services. This unified management experience simplifies operations and reduces the complexity of managing disparate environments. Additionally, Azure Arc enables advanced features like automated updates, backup and restore, and disaster recovery for your SQL Server instances, ensuring they're always up-to-date, secure, and resilient against failures. By connecting your SQL Server instances to Azure Arc, you can also take advantage of Azure's machine learning and artificial intelligence capabilities, enabling you to build and deploy intelligent applications that use your existing data.

To learn more about enabling SQL Server with Azure Arc, see [SQL Server enabled by Azure Arc](#).

4. Design Azure SQL Database for cloud-native applications

<https://learn.microsoft.com/en-us/training/modules/prepare-to-maintain-sql-databases-azure/4-design-azure-sql-database-for-cloud-native-applications>

Design Azure SQL Database for cloud-native applications

Completed

Azure SQL Database is a Platform as a Service (PaaS) that provides high scalability capabilities, and it can be a great solution for certain workloads, and requires minimal maintenance efforts.

Azure SQL Database is aimed at new application development as it gives developers a great deal of flexibility in building new application services, and granular deployment options at scale. SQL Database offers a low maintenance solution that can be a great option for certain workloads.

Purchasing model

SQL Database comes in two main purchasing models: vCore-based model and DTU-based model. Each purchasing model offers the following service tiers:

vCore-based

In this purchasing model, compute and storage resources are decoupled. It means you can scale storage and compute resources independently from one another. Here are listed the service tiers

available:

Service tier	Capability
General Purpose	This service tier is designed for less intensive operations, and offers budget oriented balanced compute and storage options. It provides both provisioned compute tier and serverless compute tier.
Business Critical	This service tier supports In-Memory OLTP, and built-in read-only replica. It also includes more memory per core, and uses local SSD storage, which is designed for performance-sensitive workloads.
Hyperscale	Hyperscale introduces horizontal scaling features that use advanced techniques to add compute nodes as the data sizes grow. It's only supported on single SQL database. Hyperscale allows you to scale storage and compute resources significantly over the limits available for the General Purpose and Business Critical service tiers.

DTU-based

In the DTU model, there are three service tiers available: Basic, Standard, and Premium. Compute and storage resources are dependent on the DTU level, and they provide a range of performance capabilities at a fixed storage limit, backup retention, and cost.

For example, if your database grows to the point it reaches the maximum storage limit, you would need to increase your DTU capacity, even if the compute utilization is low.

The scaling operation on SQL Database may incur in a brief connection interruption at the end of the scaling operation. There are two main changes that trigger this behavior:

- Once you initiate a scaling operation that requires an internal failover.
- When adding or removing databases to the elastic pool.

You can use a proper [retry logic](#) in your application to handle connection errors.

Note

Azure SQL Managed Instance doesn't support DTU-based purchasing model.

Serverless compute tier

Despite its name, the serverless compute tier does require you to have a server with your database. The serverless option can best be thought of as an autoscaling and autopause solution for SQL Database. It's effective for lowering the costs in development and testing environments. For example, you can set up a minimum and a maximum vCores configuration for your database, where it scales dynamically based on your workload.

The autopause delay feature allows you to define the period of time the database will be inactive before it's automatically paused. The autopause delay feature can be set up from 1 hour to seven

days. Alternatively, autopause delay feature can be disabled.

The resume operation is triggered when the next attempt to access the database occurs, and only storage charges are applicable when the database is paused.

The screenshot displays the 'Configure' page for an Azure SQL Database. The 'Service and compute tier' section is active, showing the 'General Purpose (Scalable compute and storage options)' service tier. Under 'Compute tier', the 'Serverless' option is selected and highlighted with a red box. The 'Compute Hardware' section shows 'Gen5' hardware configuration with a red box around the 'Max vCores' and 'Min vCores' sliders, which are set to 4 and 0.5 respectively. Below this, the 'Auto-pause delay' section is also highlighted with a red box, showing 'Enable auto-pause' checked and a delay of 0 days, 1 hour, and 0 minutes. The 'Data max size (GB)' is set to 32, and '9.6 GB LOG SPACE ALLOCATED' is shown. On the right, a 'Cost summary' box shows 'Gen5 - General Purpose (GP S Gen5 4)' with a cost of \$41.6 per month. A 'NOTES' section at the bottom right explains that serverless databases are billed in vCores based on CPU and memory utilization.

Configure

Feedback

Service and compute tier

Select from the available tiers based on the needs of your workload. The vCore model provides a wide range of configuration controls and offers Hyperscale and Serverless to automatically scale your database based on your workload needs. Alternately, the DTU model provides set price/performance packages to choose from for easy configuration. [Learn more](#)

Service tier: General Purpose (Scalable compute and storage options) [Compare service tiers](#)

Compute tier:

- ☐ Provisioned - Compute resources are pre-allocated. Billed per hour based on vCores configured.
- ☒ **Serverless** - Compute resources are auto-scaled. Billed per second based on vCores used.

Compute Hardware

Select the hardware configuration based on your workload requirements. Availability of compute optimized, memory optimized, and confidential computing hardware depends on the region, service tier, and compute tier.

Hardware Configuration: Gen5 up to 40 vCores, up to 120 GB memory [Change configuration](#)

Max vCores: 4 vCores

Min vCores: 0.5 vCores

2.1 GB MIN MEMORY 12 GB MAX MEMORY

Auto-pause delay

The database automatically pauses if it is inactive for the time period specified here, and automatically resumes when database activity recurs. Alternatively, auto-pausing can be disabled.

☒ Enable auto-pause

Days: 0 Hours: 1 Minutes: 0

Data max size (GB): 32

9.6 GB LOG SPACE ALLOCATED

Apply

Cost summary

Gen5 - General Purpose (GP S Gen5 4)
Cost per GB (in USD)

Max storage selected (in GB) x 41.6

ESTIMATED STORAGE COST / MONTH USD

COMPUTE COST / VCORE / SECOND ¹ USD

NOTES

¹ Serverless databases are billed in vCores based on a combination of CPU and memory utilization. [Learn more about serverless billing](#)

The image above shows where you can change the autoscaling and autopause properties for serverless compute tier.

Deployment model

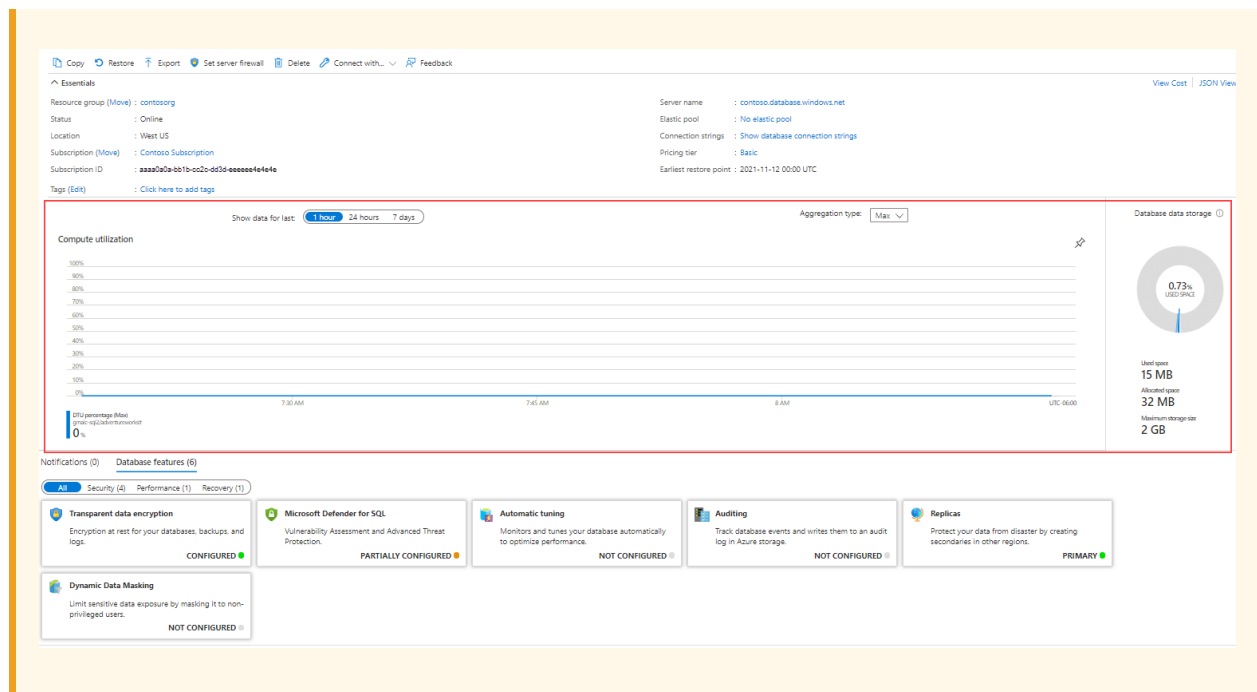
There are two main deployment models when deploying SQL Database on Azure: **single database** and **elastic pool**. Elastic pools share resources with other databases part of the same pool, while single databases resources are managed independently.

SQL Database, like virtual machines, can be deployed using Azure Resource Manager templates, PowerShell, Azure CLI, or the Azure portal.

Single database

Single database is the simplest deployment model of Azure SQL Database. You manage each of your databases individually from scale and data size perspectives. Each database deployed in this model has its own dedicated resources, even if deployed to the same logical server.

You can monitor database resources utilization through Azure portal. This feature allows you to easily identify how the database is performing, as shown in the image below:



Elastic pool

Elastic pools allow you to allocate storage and compute resources to a group of databases, rather than having to manage resources for each database individually. Additionally, elastic pools are easier to scale than single databases, where scaling individual databases is no longer needed due to changes made to the elastic pool.

Elastic pools provide a cost-effective solution for software as a service application model, since resources are shared between all databases. You can configure resources based either on the DTU-based purchasing model or the vCore-based purchasing model.

Due to the nature of this feature, it's recommended to monitor your resources continually to identify concurrent performance spikes that could affect other databases part of the same elastic pool. Often, you may need to revisit your allocation strategy to make sure there's enough resource available for all databases sharing the same elastic pool.

Elastic pool is a good fit for multitenant architecture with low average utilization, where each tenant has its own copy of the database.

Network options

Azure SQL Database by default has a public internet endpoint. Access to this endpoint can be controlled via firewall rules, or limited to specific Azure networks, using features like Virtual Network endpoints or Private Link.

Backup and restore

Azure provides seamless backup and restore capabilities for SQL Database and SQL Managed Instance. Let's learn about some of the most important features.

Continuous backup

With SQL Database, you can increase administration efficiency by knowing that databases are backed up regularly, and that they're continuously copied to a read-access geo-redundant storage (RA-GRS).

Full backups are performed every week, differential backups every 12 to 24 hours, and transaction log backups every 5 to 10 minutes.

Geo-restore

As backups are geo-redundant by default for SQL Database and SQL Managed Instance, you can easily restore databases to a different geographical region, which is especially useful for less strict disaster recovery scenarios.

Backup storage is billed apart from regular database files storage. However, when provisioning a SQL Database, the backup storage is created with the maximum size of the data tier selected for your database at no extra cost.

The duration of a geo-restore operation can be affected by several underlying components including the size of the database, the number of transaction logs involved in a restore operation, and the amount of simultaneous restore requests being processed in the target region.

Note

Geo-restore is available when the backup storage redundancy property is set to ***geo-redundant backup storage***.

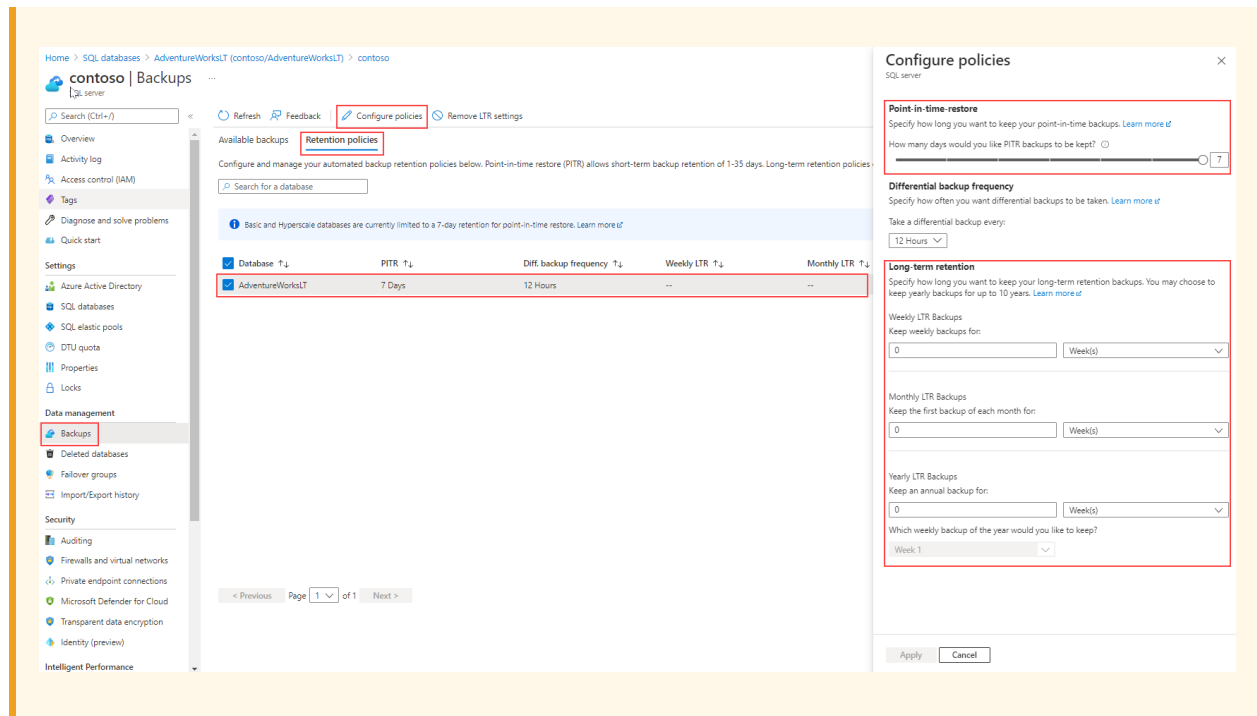
Point-in-time restore (PITR)

You can configure a specific point in time retention policy for each database running on a SQL Database offering. SQL Database retention period can be set from 1 up to 35 days. In fact, if not specified, the default configuration is seven days.

You can restore your databases to a specific point in time according to the retention defined, but PITR is only supported if you're restoring a database in the same server the backup was originated. You can use either Azure portal, Azure PowerShell, Azure CLI, or REST API to restore a SQL Database.

Long-term retention (LTR)

Long-term retention is useful for scenarios that require you to set the retention policy beyond what is offered by Azure. You can set a retention policy for up to 10 years, and this option is disabled by default.



In the image above, you can configure long-term retention policies through Azure portal. Once the database is selected, a new panel opens on the right side of the screen, where you can override the default properties.

For more information about automated backups, see [Automated backups - Azure SQL Database & Azure SQL Managed Instance](#).

Automatic Tuning

Automatic tuning is a built-in feature that relies on machine learning regression capabilities, and automatically identify tuning opportunities based on your query performance.

Automatic tuning currently includes the following features:

- Identify Expensive Queries
- Forcing Last Good Execution Plan
- Adding Indexes
- Removing Indexes

The Azure services use a combination of built-in advanced features to determine the best indexes for your query pattern. Initially, these indexes are tested on a copy of your database, and finally applied to your database.

All databases inherit configuration from their parent server, and you can easily disable this feature at any time.

Elastic query (preview)

Elastic query allows you to run T-SQL queries that bridge multiple databases in SQL Database. This feature is useful for applications using three- and four-part names that can't be changed. It also increases portability as it allows for migration.

Elastic queries support the following partitioning strategies:

Service tier	Capability
Vertical partitioning	Also called cross-database queries. The data is partitioned vertically between many databases. Schemas are different for each database. For example, you may have a database used for customer data, and a different one used for payment information. With the help of vertical partitioning, you can now run a cross-database query between both databases.
Horizontal Partitioning	Also called sharding. The data is partitioned horizontally to distribute rows across several scaled-out databases. In this topology, the schema remains the same among all sharding databases. It supports either single-tenant model or multiple tenant models.

Elastic jobs

The elastic job feature is the SQL Server Agent replacement for Azure SQL Database. To some extent, elastic job is equivalent to the Multi Server Administration feature available on an on-premises SQL Server instance.

You can execute T-SQL commands across several target deployments like SQL Databases, SQL Database elastic pools, and SQL Databases in shard maps. Database resources can run on different Azure subscriptions, and/or regions. The execution runs in parallel, which is useful when automating database maintenance tasks.

Note

Azure SQL Managed Instance doesn't support elastic jobs.

SQL Database in Fabric

SQL Database in Microsoft Fabric is a powerful and versatile solution designed to integrate seamlessly with the broader Microsoft ecosystem. It allows users to use the capabilities of SQL databases within the Microsoft Fabric environment, providing a unified platform for managing and analyzing data. This integration enables organizations to streamline their data workflows, enhance collaboration, and drive more informed decision-making processes.

Moreover, SQL Database in Microsoft Fabric supports a wide range of use cases, from operational and transactional workloads to advanced analytics. This flexibility makes it an ideal choice for organizations looking to consolidate their data infrastructure and use the full potential of their data assets. With features like seamless integration with other Microsoft services, enhanced security, and scalability, SQL Database in Microsoft Fabric empowers users to build and deploy data-driven applications with ease.

The **Databases** workload in Fabric is designed to address many aspects of operational SQL databases, enabling data engineers, analysts, and data scientists to work together to create and query data that is optimized for their specific needs.

5. Explore Azure SQL Managed Instance

<https://learn.microsoft.com/en-us/training/modules/prepare-to-maintain-sql-databases-azure/5-explore-azure-sql-database-managed-instance>

Explore Azure SQL Managed Instance

Completed

Most of the features available on Azure SQL Database will also work for Azure SQL Managed Instance as they share the same base code. The fully managed platform as a service (PaaS) provides some of the following benefits:

- Automatic backups
- Automatic patching
- Built-in high availability
- Security and performance tools
- Embedded auditing capabilities

Another key benefit when migrating to one of the PaaS offerings on Azure is that you no longer have to install or patch SQL Server, which can increase application uptime, and reduce maintenance efforts.

Unlike Azure SQL Database, which is designed around single database structures, SQL Managed Instance provides several other features including cross-database queries, common language runtime (CLR), access to the system databases, and use of the SQL Agent features.

For a complete list of the features available on Azure SQL Managed Instance, see [Features of SQL Database and SQL Managed Instance](#).

Hybrid licensing options

Microsoft offers several benefits to SQL Server licenses. For both SQL Database and SQL Managed Instance, taking advantage of your existing licenses can reduce the cost of running the PaaS offering.

- For each core of Enterprise Edition with Active Software Assurance, you're eligible for one vCore of SQL Database or SQL Managed Instance Business Critical, and eight vCores of General Purpose.
- For each core of Standard Edition with Active Software Assurance, you're eligible for one vCore of General Purpose.

This model can reduce the total license costs by up to 40%. Effectively, you'll only be paying for the compute and storage costs, and not the software licensing costs.

For more information on bring-your-own licensing model, see [License Mobility through Software Assurance on Azure](#).

Connectivity architecture

Connections to SQL Managed Instance are made through TDS endpoints. While the routing and security on these connections differ, there's a gateway component that handles and routes connections to the database service. This gateway component is also deployed in a highly available fashion.

Backup and restore

Automated database backup provides a fully managed backup service that takes full, differential, and log backups regularly for both SQL Managed Instance and SQL Database offerings. Automated backups are geo-redundant, and replicated to a paired-region automatically, which protects your data against localized outages in the primary region.

Similarly, SQL Managed Instance allows easy migration of existing applications, enabling restores from on-premises backups.

There are some important considerations when running backup and restore operations on SQL Managed Instance databases:

- It isn't possible to overwrite an existing database during the restore process. Before restoring a database, you must ensure that it doesn't exist.
- For SQL Managed Instance, backups can only be restored to another managed instance. It isn't possible to restore a managed instance database backup to a SQL Server running on a virtual machine or SQL Database.
- Copy-only backup to Azure blob storage is available for SQL Managed Instance. SQL Database doesn't support this feature.

For more information about automated backups, see [Automated backups - Azure SQL Database & Azure SQL Managed Instance](#).

High availability architecture

SQL Database and SQL Managed Instance have similar high availability architectures, which guarantee 99.99% uptime. Windows and SQL Server updates are handled by the backend infrastructure, generally without any effect to your application, though it's important to place a [retry logic](#) into your application.

The auto-failover groups feature allows you to fail over a group of replicated databases on a server to another region. This feature is designed on top of the existing active geo-replication capability, which simplifies deployment and management of geo-replicated databases.

A failover group can include one or multiple databases, often used by the same application. Additionally, you can use the readable secondary databases to offload read-only query workloads.

Note

The auto-failover groups feature is supported on both SQL Managed Instance and SQL Database.

For more information about auto-failover groups, see [Use auto-failover groups to enable transparent and coordinated geo-failover of multiple databases](#).

Migration options

In general, migrating to SQL Managed Instance is often simple given the large set of available features. There are a couple of ways to migrate on-premises databases:

- **Log Replay Service.** It's an online migration option, and used when you need more control of your database migration project.
- **Managed Instance link.** The Managed Instance link, using distributed availability groups, securely extends your data estate by replicating data almost instantly (online) between any hosted SQL Server and Azure SQL Managed Instance, and vice versa.
- **Native backup and restore.** Backup and restore are a simple migration method favored by many SQL Server professionals. It's the easiest migration option for customers who can provide full database backups to Azure Storage.
- **Transactional replication.** Transactional replication is a way to move data between continuously connected database servers. It's best to be used for online or offline migration of large and complex databases.

Machine Learning Services

Machine Learning Services provides machine learning operations within your relational database structure. This feature supports Python and R packages, ideal for high-intensive predictive capabilities. This option is available on SQL Managed Instance, SQL Server on Azure virtual machine, and on-premises SQL Server.

Applications can use relational database on Azure combined with machine learning high-performance capabilities, where you can:

- Train machine learning models based on either sampled dataset or population dataset.
- Reduce complexity in security and compliance, where you don't need to relocate your data to build and train your machine learning models.
- Deploy machine learning models using T-SQL stored procedures that support Python or R programming language.
- Use of open-source libraries like scikit-learn, PyTorch, and TensorFlow.

For busy environments, you can use the [T-SQL PREDICT function](#), which allows you to accelerate predictions based on your stored model.

The Machine Learning Services feature can be enabled by running the following command:

```
EXEC sp_configure 'external scripts enabled', 1;  
RECONFIGURE WITH OVERRIDE;
```

The command above allows the execution of external scripts in your managed instance, and it should be enabled before you attempt to use **sp_execute_external_script** to execute Python or R scripts in your database.

Note

SQL Database doesn't support Machine Learning Services feature.

For more information about Machine Learning Services, see [Machine Learning Services in Azure SQL Managed Instance](#).

6. Module assessment

<https://learn.microsoft.com/en-us/training/modules/prepare-to-maintain-sql-databases-azure/6-knowledge-check>

Module assessment

Completed

7. Summary

<https://learn.microsoft.com/en-us/training/modules/prepare-to-maintain-sql-databases-azure/7-summary>

Summary

Completed

Microsoft provides a great deal of options for deploying SQL Server to Azure. Migrating on-premises databases to Azure SQL Server on Virtual Machine offers portability, while Azure SQL Database and Azure SQL Managed Instance provide scalability and flexibility.

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

- Understand the role of Azure Database Administrator, and other data platform roles
- Describe the key differences between the SQL Server-based database options in Azure
- Describe other features for Azure SQL platforms available

Note

If you're ready to get started with Azure SQL Database, [try Azure SQL Database free of charge](#) for the life of your subscription.