1. **Azure SQL Database**
2. **Azure Database for PostgreSQL**
3. **Azure Database for MySQL**
4. **Azure Database for MariaDB**

**Introduction**

1. **A database is a collection of data**.
2. A database can be as simple as a desktop spreadsheet, or as complex as a global system holding petabytes of highly structured information.
3. The data can be structured in many different ways.
4. A common approach is to store data in a **tabular format, with rows and columns**.
5. **You can define relationships between tables. These databases are called *relational* databases**.
6. Databases can also be **semi-structured or unstructured**, comprising of semi-processed or unprocessed data.
7. These databases are typically referred to as ***non-relational*.**
8. **Databases are managed using a database management system (DBMS). The DBMS handles the physical aspects of a database, such as where and how it's stored, who can access it, and how to ensure that it's available when required.**
9. Many organizations depend on the information stored in their databases to help make critical business decisions. In the past, these organizations ran their DBMSs on-premises. However, this approach requires the organization to maintain its own hardware infrastructure. Therefore, an increasing number of businesses are migrating their databases to the cloud, where the costs of configuring and maintaining the infrastructure are highly reduced.
10. Suppose you're a database administrator at Wide World Importers. You're responsible for database design and maintenance, as well as providing information for leadership and creating customer lists for the marketing department. You have an existing SQL Server database that relies heavily on stored procedures and other advanced database features such as linked servers. The database is situated on your internal network. You've been asked to make it globally available to your partners worldwide.

**Note**

1. **A *stored procedure* is a block of code that runs inside your database**. Applications often use stored procedures because they are optimized to run in the database environment, and can access data very quickly.
2. A ***linked server*** is a connection from one database server to another.
3. SQL Server can use linked servers to run queries on one server that can include data retrieved from other servers; these are known **as *distributed queries*.**

**Explore relational Azure data services**

Azure offers a range of options for running a database management system in the cloud. For example, you can migrate your on-premises systems to a collection of Azure virtual machines. This approach still requires that you manage your DBMS carefully. Alternatively, you can take advantage of the various Azure relational data services available. These data services manage the DBMS for you, leaving you free to concentrate on the data they contain and the applications that use them.

**Understand IaaS, PaaS, and SaaS**

Before delving into Azure Data Services, you need to understand some common terms used to describe the different ways in which you can host a database in Azure.

**IaaS** is an acronym for *Infrastructure-as-a-Service*. Azure enables you to create a virtual infrastructure in the cloud that mirrors the way an on-premises data center might work. You can create a set of virtual machines, connect them together using a virtual network, and add a range of virtual devices. You take responsibility for installing and configuring the software, such as the DBMS, on these virtual machines. In many ways, this approach is similar to the way in which you run your systems inside an organization, except that you don't have to concern yourself with buying or maintaining the hardware.

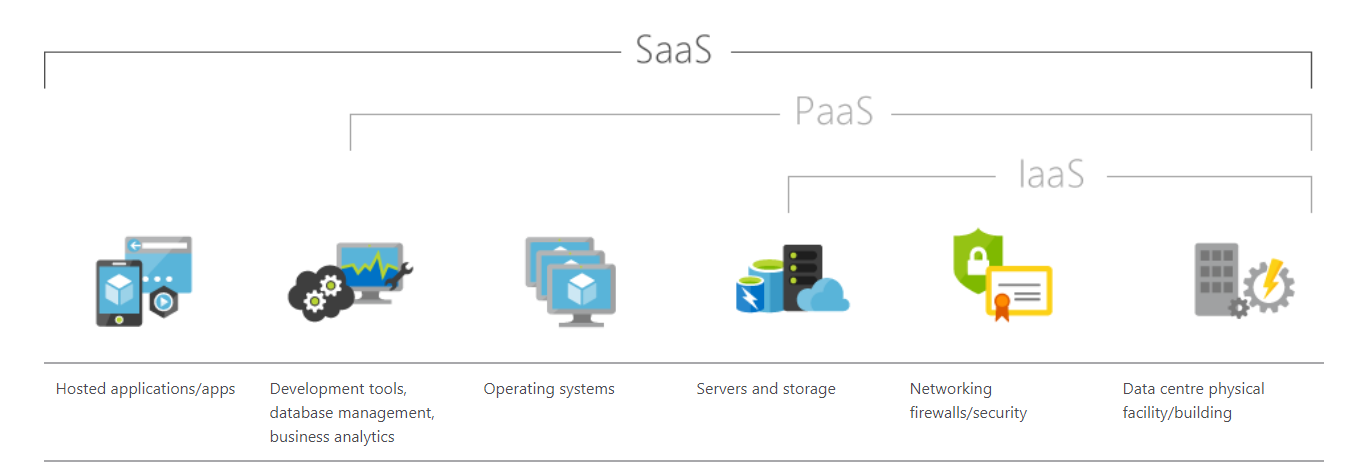
**Note**

An Azure Virtual Network is a representation of your own network in the cloud. A virtual network enables you to connect virtual machines and Azure services together, in much the same way that you might use a physical network on-premises. Azure ensures that each virtual network is isolated from other virtual networks created by other users, and from the Internet. Azure enables you to specify which machines (real and virtual), and services, are allowed to access resources on the virtual network, and which ports they can use.

**PaaS** stands for *Platform-as-a-service*. Rather than creating a virtual infrastructure, and installing and managing the database software yourself, a PaaS solution does this for you. You specify the resources that you require (based on how large you think your databases will be, the number of users, and the performance you require), and Azure automatically creates the necessary virtual machines, networks, and other devices for you. You can usually scale up or down (increase or decrease the size and number of resources) quickly, as the volume of data and the amount of work being done varies; Azure handles this scaling for you, and you don't have to manually add or remove virtual machines, or perform any other form of configuration.

**SaaS** is short for *Software-as-a-Service*. SaaS services are typically specific software packages that are installed and run on virtual hardware in the cloud. SaaS packages are typically hosted applications rather than more generalized software such as a DBMS. Common SaaS packages available on Azure include Microsoft 365 (formerly Office 365).

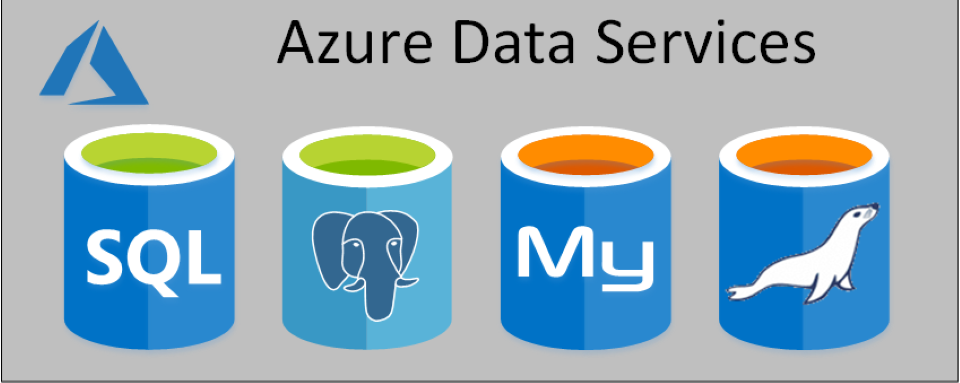
| **TABLE 1** | | |
| --- | --- | --- |
|  | **Example** | **Includes** |
| IaaS | Azure virtual network | Servers, storage, networking, and physical data center. |
| PaaS | Azure SQL Databases | IaaS plus database management (or other server systems), and operating systems. |
| SaaS | Office 365 | PaaS plus apps. |



**What are Azure Data Services?**

**Azure Data Services fall into the PaaS category.** These services are a series of DBMSs managed by Microsoft in the cloud. Each data service takes care of the configuration, day-to-day management, software updates, and security of the databases that it hosts. All you do is create your databases under the control of the data service.

Azure Data Services are available for several common relational database management systems. The most well-known service is **Azure SQL Database.** The others currently available are Azure Database for MySQL servers, Azure Database for MariaDB servers, and Azure Database for PostgreSQL servers. The remaining units in this module describe the features provided by these services.



**Note**

Microsoft also provides data services for non-relational database management systems, such as Cosmos DB.

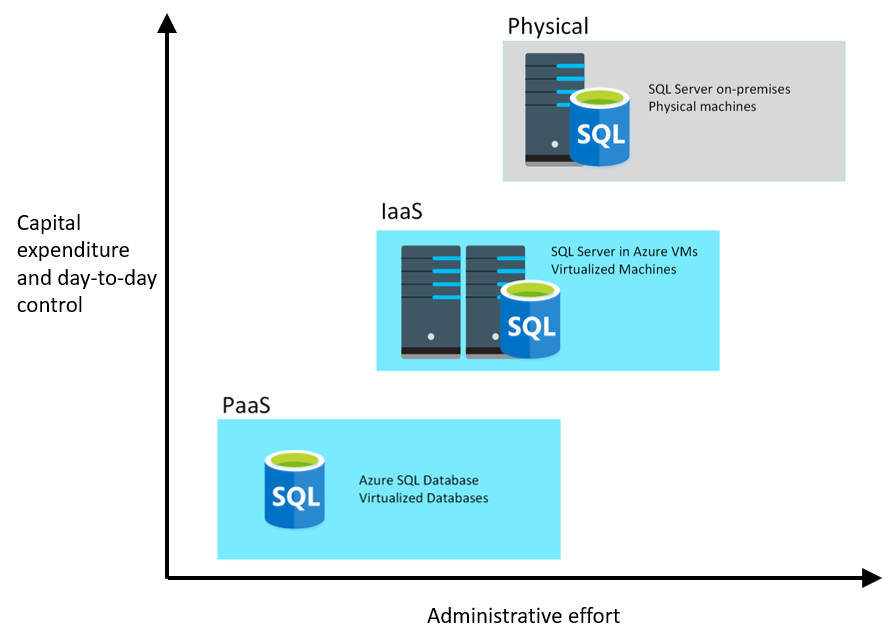
Using Azure Data Services reduces the amount of time that you need to invest to administer a DBMS. However, these services can also limit the range of custom administration tasks that you can perform, because manually performing some tasks might risk compromising the way in which the service runs. For example, some DBMSs enable you to install custom software into a database, or run scripts as part of a database operation. This software might not be supported by the data service, and allowing an application to run a script from a database could affect the security of the service. You must be prepared to work with these restrictions in mind.

**Apart from reducing the administrative workload, Azure Data Services ensure that your databases are available for at least 99.99% of the time.**

There are costs associated with running a database in Azure Data Services. The base price of each service covers underlying infrastructure and licensing, together with the administration charges. Additionally, these services are designed to be *always on*. This means that you can't shut down a database and restart it later.

Not all features of a database management system are available in Azure Data Services. This is because Azure Data Services takes on the task of managing the system and keeping it running using hardware situated in an Azure datacenter. Exposing some administrative functions might make the underlying platform vulnerable to misuse, and even open up some security concerns. Therefore, you have no direct control over the platform on which the services run. If you need more control than Azure Data Services allow, you can install your database management system on a virtual machine that runs in Azure. The next unit examines this approach in more detail for SQL Server, although the same issues apply for the other database management systems supported by Azure Data Services.

The image below highlights the different ways in which you could run a DBMS such as SQL Server, starting with an on-premises system in the bottom left-hand corner, to PaaS in the upper right. The diagram illustrates the benefits of moving to the PaaS approach.



# SQL Server on Azure virtual machines

**Microsoft SQL Server is a popular relational DBMS.**

* It has a long history, and has features that provide database management to organizations of all sizes.
* In the past, organizations have run SQL Server on-premises.
* However, many organizations are now looking to shift operations on-line to take advantage of services available in the cloud.
* SQL Server offers several ways to run a database in Azure. In this unit, you'll look at moving SQL Server to an Azure Virtual Machine.

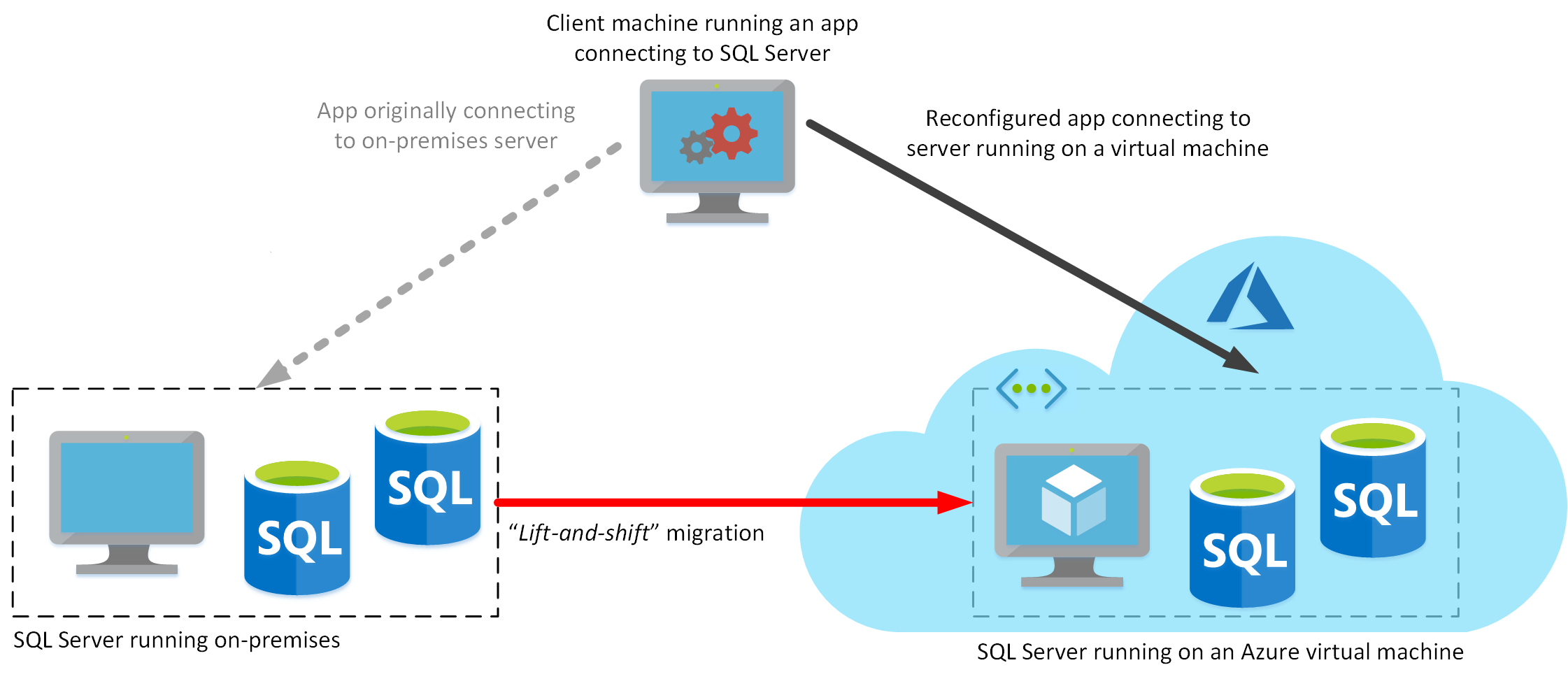
## What is SQL Server on Azure Virtual Machines?

SQL Server on Virtual Machines enables you to use full versions of SQL Server in the Cloud without having to manage any on-premises hardware. This is an example of the **IaaS** approach.

SQL Server running on an Azure virtual machine effectively replicates the database running on real on-premises hardware. Migrating from the system running on-premises to an Azure virtual machine is no different than moving the databases from one on-premises server to another.

In the example scenario described in the introduction, the database runs stored procedures and scripts as part of the database workload. If these stored procedures and scripts depend on features that are restricted by following a PaaS approach, then running SQL Server on your own virtual machines might be a good option. However, you remain responsible for maintaining the SQL Server software and performing the various administrative tasks to keep the database running from day-to-day.

This approach is suitable for migrations and applications requiring access to operating system features that might be unsupported at the PaaS level. SQL virtual machines are lift-and-shift ready for existing applications that require fast migration to the cloud with minimal changes.



**Note**

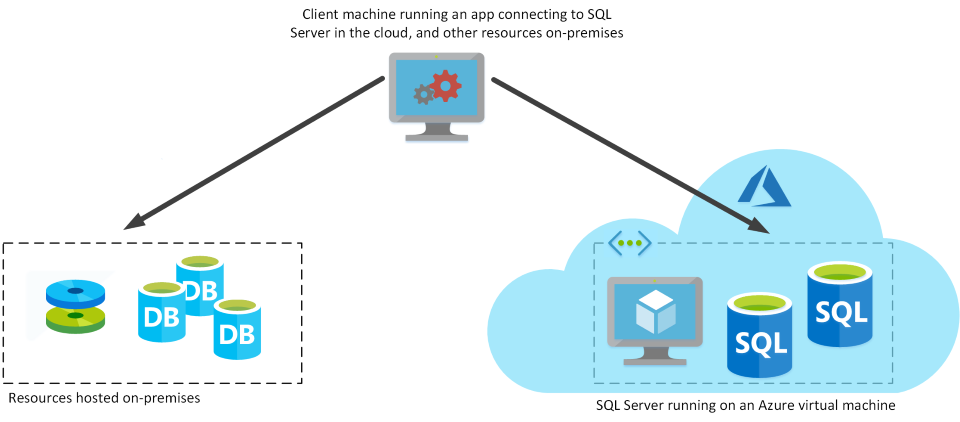
The **term lift-and-shift refers** to the way in which you can move a database directly from an on-premises server to an Azure virtual machine without requiring that you make any changes to it. Applications that previously connected to the on-premises database can be quickly reconfigured to connect to the database running on the virtual machine, but should otherwise remain unchanged.

### Use cases

This approach is optimized for migrating existing applications to Azure, or extending existing on-premises applications to the cloud in hybrid deployments.

**Note**

A **hybrid** **deployment** is a system where part of the operation runs on-premises, and part in the cloud. Your database might be part of a larger system that runs on-premises, although the database elements might be hosted in the cloud.



You can use SQL Server in a virtual machine to develop and test traditional SQL Server applications. With a virtual machine, you have the full administrative rights over the DBMS and operating system. It's a perfect choice when an organization already has IT resources available to maintain the virtual machines.

These capabilities enable you to:

* Create rapid development and test scenarios when you do not want to buy on-premises non-production SQL Server hardware.
* Become lift-and-shift ready for existing applications that require fast migration to the cloud with minimal changes or no changes.
* Scale up the platform on which SQL Server is running, by allocating more memory, CPU power, and disk space to the virtual machine. You can quickly resize an Azure virtual machine without the requirement that you reinstall the software that is running on it.

### Business benefits

Running SQL Server on virtual machines allows you to meet unique and diverse business needs through a combination of on-premises and cloud-hosted deployments, while using the same set of server products, development tools, and expertise across these environments.

It's not always easy for businesses to switch their DBMS to a fully managed service. There may be specific requirements that must be satisfied in order to migrate to a managed service that requires making changes to the database and the applications that use it. For this reason, using virtual machines can offer a solution, but using them does not eliminate the need to administer your DBMS as carefully as you would on-premises.

# Azure SQL Database

If you don't want to incur the management overhead associated with running SQL Server on a virtual machine, you can use Azure SQL Database.

## What is Azure SQL Database?

Azure SQL Database is a **PaaS** offering from Microsoft.

You create a managed database server in the cloud, and then deploy your databases on this server.

**Note**

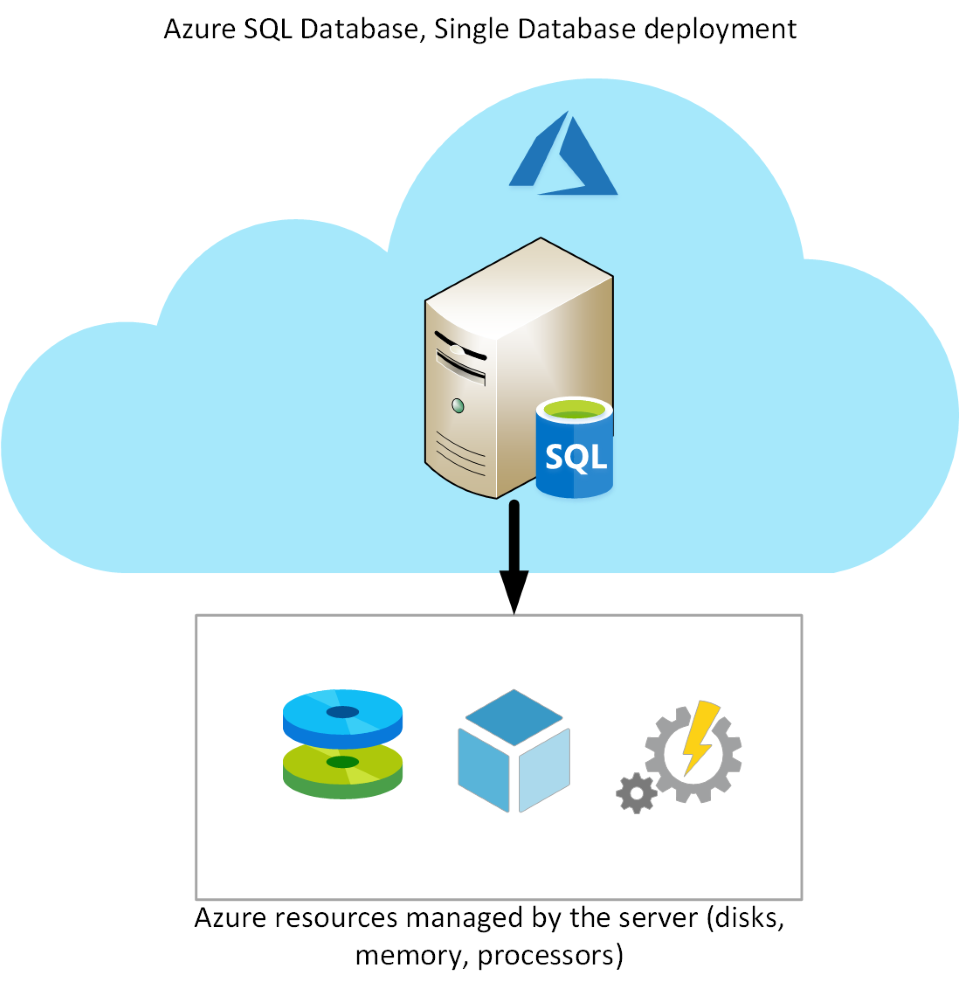
A SQL Database server is a logical construct that acts as a central administrative point for multiple single or pooled databases, **logins, firewall rules, auditing rules, threat detection policies, and failover groups**.

Azure SQL Database is available with several options:

* Single Database,
* Elastic Pool,
* Managed Instance.

### **Single Database**

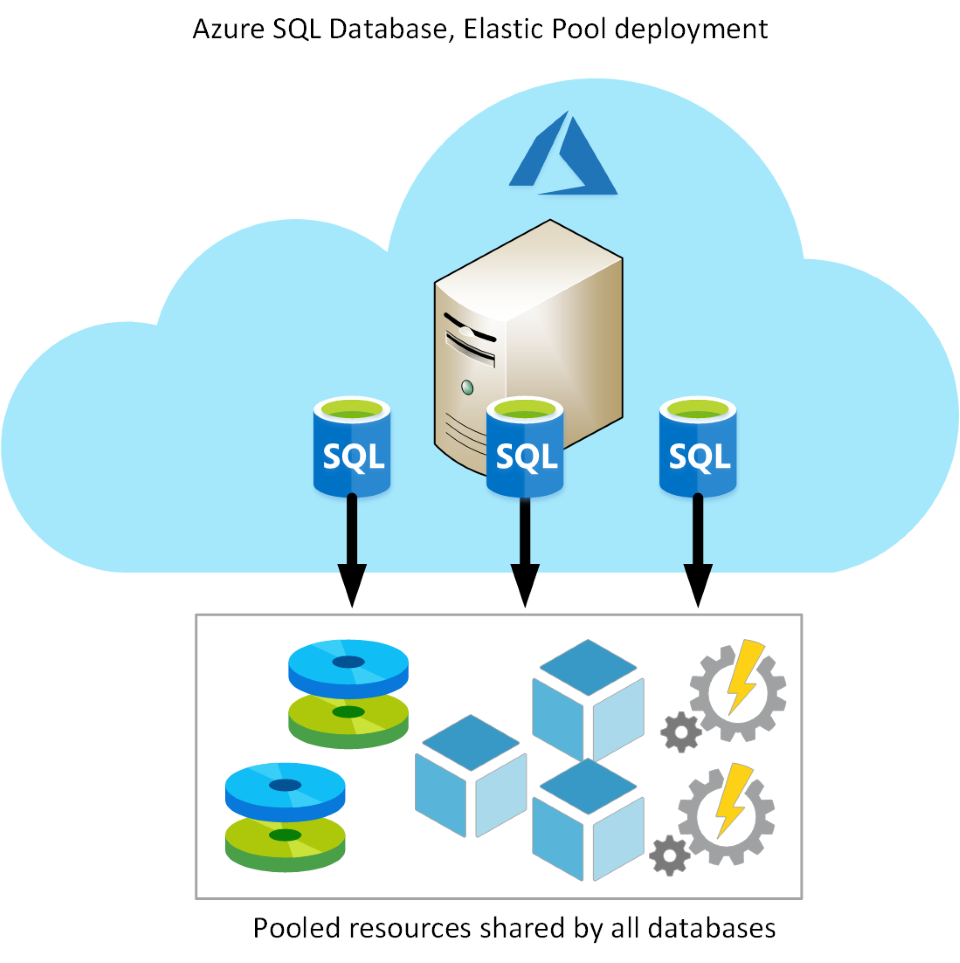
This option enables you to quickly set up and run a single SQL Server database. You create and run a database server in the cloud, and you access your database through this server. Microsoft manages the server, so all you have to do is configure the database, create your tables, and populate them with your data. You can scale the database if you need additional storage space, memory, or processing power. By default, resources are pre-allocated, and you're charged per hour for the resources you've requested. You can also specify a serverless configuration. In this configuration, Microsoft creates its own server, which might be shared by a number of databases belonging to other Azure subscribers. Microsoft ensures the privacy of your database. Your database automatically scales and resources are allocated or deallocated as required. For more information, read [What is a single database in Azure SQL Database](https://docs.microsoft.com/en-us/azure/sql-database/sql-database-single-database).



### **Elastic Pool**

This option is similar to Single Database, except that by default multiple databases can share the same resources, such as memory, data storage space, and processing power through **multiple-tenancy**.

The resources are referred to as a pool. You create the pool, and only your databases can use the pool. This model is useful if you have databases with resource requirements that vary over time, and can help you to reduce costs. For example, your payroll database might require plenty of CPU power at the end of each month as you handle payroll processing, but at other times the database might become much less active. You might have another database that is used for running reports. This database might become active for several days in the middle of the month as management reports are generated, but with a lighter load at other times. Elastic Pool enables you to use the resources available in the pool, and then release the resources once processing has completed.



## Use cases

Azure SQL Database gives you the best option for low cost with minimal administration. It is not fully compatible with on-premises SQL Server installations. It is often used in new cloud projects where the application design can accommodate any required changes to your applications.

**Note**

You can use the Data Migration Assistant to detect compatibility issues with your databases that can impact database functionality in Azure SQL Database. For more information, see [**Overview of Data Migration Assistant**](https://docs.microsoft.com/en-us/sql/dma/dma-overview).

Azure SQL Database is often used for:

* Modern cloud applications that need to use the latest stable SQL Server features.
* Applications that require high availability.
* Systems with a variable load, that need the database server to scale up and down quickly.

## Business benefits

Azure SQL Database automatically updates and patches the SQL Server software to ensure that you are always running the latest and most secure version of the service.

The scalability features of Azure SQL Database ensure that you can increase the resources available to store and process data without having to perform a costly manual upgrade.

The service provides high availability guarantees, to ensure that your databases are available at least 99.99% of the time. Azure SQL Database supports point-in-time restore, enabling you to recover a database to the state it was in at any point in the past. Databases can be replicated to different regions to provide additional assurance and disaster recovery

Advanced threat protection provides advanced security capabilities, such as vulnerability assessments, to help detect and remediate potential security problems with your databases. Threat protection also detects anomalous activities that indicate unusual and potentially harmful attempts to access or exploit your database. It continuously monitors your database for suspicious activities, and provides immediate security alerts on potential vulnerabilities, SQL injection attacks, and anomalous database access patterns. Threat detection alerts provide details of the suspicious activity, and recommend action on how to investigate and mitigate the threat.

Auditing tracks database events and writes them to an audit log in your Azure storage account. Auditing can help you maintain regulatory compliance, understand database activity, and gain insight into discrepancies and anomalies that might indicate business concerns or suspected security violations.

SQL Database helps secure your data by providing encryption. For data in motion, it uses Transport Layer Security. For data at rest, it uses Transparent Data Encryption. For data in use, it uses Always Encrypted. For more information on Transport Layer Security, Transparent Data Encryption, and Always Encrypted, see the links in the Summary unit.

In the Wide World Importers scenario, linked servers are used to perform distributed queries. However, neither Single Database nor Elastic Pool support linked servers. If you want to use Single Database or Elastic Pool, you may need to modify the queries that use linked servers and rework the operations that depend on these features.

**Azure SQL Database Managed Instance**

A business may want to eliminate as much management overhead as possible from administering databases and servers, but the limitations of the Single Database and Elastic Pool options may mean that those options aren't suitable. In these situations. Azure SQL Database managed instance may be a good choice to consider.

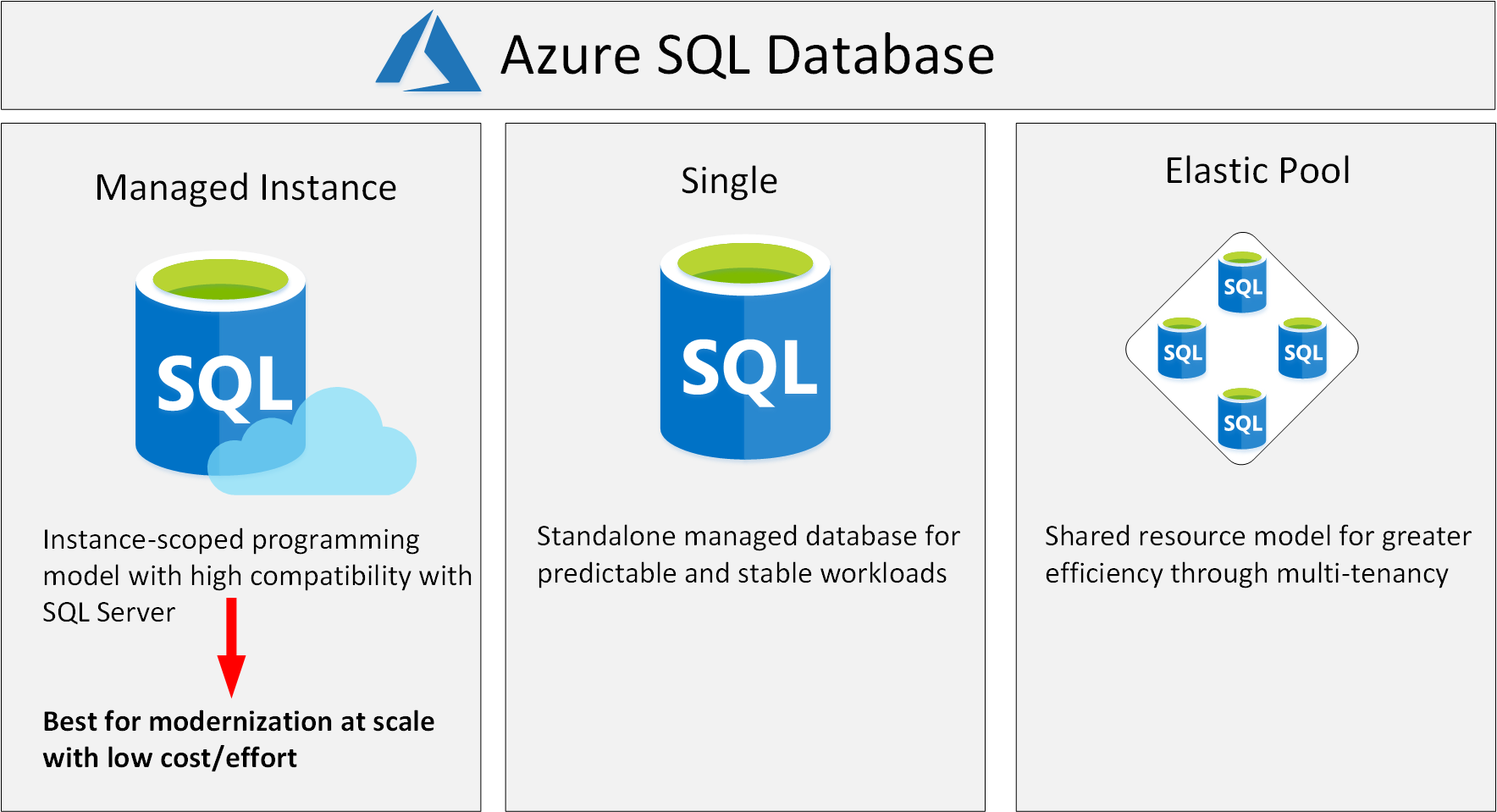
**What is Azure SQL Database managed instance?**

The Single Database and Elastic Pool options restrict some of the administrative features available to SQL Server. Managed instance effectively runs a fully controllable instance of SQL Server in the cloud. You can install multiple databases on the same instance. You have complete control over this instance, much as you would for an on-premises server. The Managed instance service automates backups, software patching, database monitoring, and other general tasks, but you have full control over security and resource allocation for your databases. You can find detailed information at [What is Azure SQL Database managed instance?](https://docs.microsoft.com/en-us/azure/sql-database/sql-database-managed-instance).

Managed instances depend on other Azure services such as Azure Storage for backups, Azure Event Hubs for telemetry, Azure Active Directory for authentication, Azure Key Vault for Transparent Data Encryption (TDE) and a couple of Azure platform services that provide security and supportability features. The managed instances make connections to these services.

All communications are encrypted and signed using certificates. To check the trustworthiness of communicating parties, managed instances constantly verify these certificates through certificate revocation lists. If the certificates are revoked, the managed instance closes the connections to protect the data.

The following image summarizes the differences between SQL Database managed instance, Single Database, and Elastic Pool



**Use cases**

Consider Azure SQL Database managed instance if you want to *lift-and-shift* an on-premises SQL Server instance and all its databases to the cloud, without incurring the management overhead of running SQL Server on a virtual machine.

SQL Database managed instance provides features not available with the Single Database or Elastic Pool options. If your system uses features such as linked servers, Service Broker (a message processing system that can be used to distribute work across servers), or Database Mail (which enables your database to send email messages to users), then you should use managed instance. To check compatibility with an existing on-premises system, you can install [Data Migration Assistant (DMA)](https://www.microsoft.com/download/details.aspx?id=53595). This tool analyzes your databases on SQL Server and reports any issues that could block migration to a managed instance.

**Business benefits**

SQL Database managed instance provides all the management and security benefits available when using Single Database and Elastic Pool. managed instance deployment enables a system administrator to spend less time on administrative tasks because the SQL Database service either performs them for you or greatly simplifies those tasks. Automated tasks include operating system and database management system software installation and patching, dynamic instance resizing and configuration, backups, database replication (including system databases), high availability configuration, and configuration of health and performance monitoring data streams.

Managed instance has near 100% compatibility with SQL Server Enterprise Edition, running on-premises.

The SQL Database managed instance deployment option supports traditional SQL Server Database engine logins and logins integrated with Azure Active Directory (AD). Traditional SQL Server Database engine logins include a username and a password. You must enter your credentials each time you connect to the server. Azure AD logins use the credentials associated with your current computer sign-in, and you don't need to provide them each time you connect to the server.

In the Wide World Importers scenario, SQL Database managed instance may be a more suitable choice than Single Database or Elastic Pool. SQL Database managed instance supports linked servers, although some of the other the advanced features required by the database might not be available. If you want a complete match, then running SQL Server on a virtual machine may be your only option, but you need to balance the benefits of complete functionality against the administrative and maintenance overhead required.

# PostgreSQL, MariaDB, and MySQL

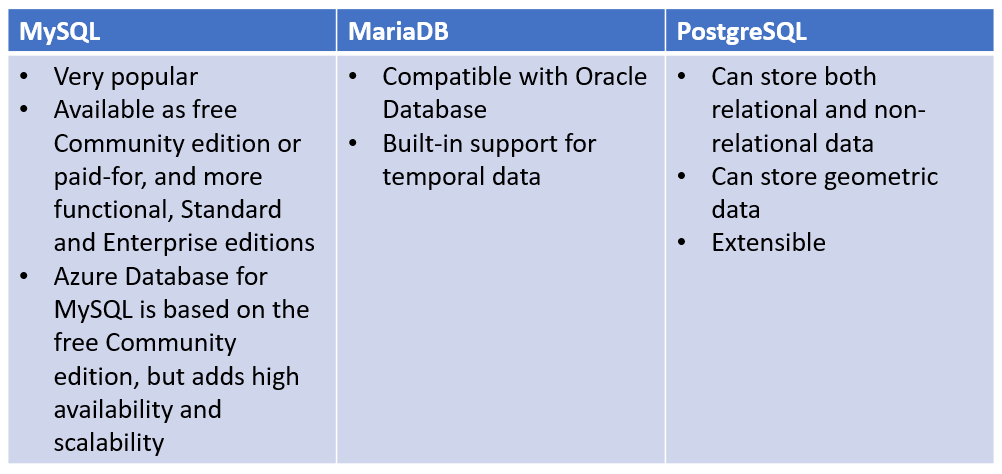
As well as Azure SQL Database, Azure Data Services are available for other popular SQL-based database solutions. Currently, data services are available for PostgreSQL, MySQL, and MariaDB. The primary reason for these services is to enable organizations running PostgreSQL, MySQL, or MariaDB to move to Azure quickly, without making wholesale changes to their applications.

## What are MySQL, MariaDB, and PostgreSQL

PostgreSQL, MariaDB, and MySQL are relational database management systems that are tailored for different specializations.

**MySQL started life as a simple-to-use open-source database management system.**

* It is the leading open source relational database for Linux, Apache, MySQL, and PHP (LAMP) stack apps.
* It's available in several editions; Community, Standard, and Enterprise.
* The Community edition is available free-of-charge, and has historically been popular as a database management system for web applications, running under Linux.
* Versions are also available for Windows.
* Standard edition offers higher performance, and uses a different technology for storing data.
* Enterprise edition provides a comprehensive set of tools and features, including enhanced security, availability, and scalability.
* The Standard and Enterprise editions are the versions most frequently used by commercial organizations, although these versions of the software aren't free.
* MariaDB is a newer database management system, created by the original developers of MySQL.
* The database engine has since been rewritten and optimized to improve performance.
* MariaDB offers compatibility with Oracle Database (another popular commercial database management system).
* One notable feature of MariaDB is its built-in support for temporal data. A table can hold several versions of data, enabling an application to query the data as it appeared at some point in the past.
* PostgreSQL is a hybrid relational-object database.
* You can store data in relational tables, but a PostgreSQL database also enables you to store custom data types, with their own non-relational properties.
* The database management system is extensible; you can add code modules to the database, which can be run by queries.
* Another key feature is the ability to store and manipulate geometric data, such as lines, circles, and polygons.
* PostgreSQL has its own query language called pgsql.
* This language is a variant of the standard relational query language, SQL, with features that enable you to write stored procedures that run inside the database.



## What is Azure Database for MySQL?

Azure Database for MySQL is a PaaS implementation of MySQL in the Azure cloud, based on the MySQL Community Edition.

The Azure Database for MySQL service includes high availability at no additional cost and scalability as required. You only pay for what you use. Automatic backups are provided, with point-in-time restore.

The server provides connection security to enforce firewall rules and, optionally, require SSL connections. Many server parameters enable you to configure server settings such as lock modes, maximum number of connections, and timeouts.

Azure Database for MySQL provides a global database system that scales up to large databases without the need to manage hardware, network components, virtual servers, software patches, and other underlying components.

Certain operations aren't available with Azure Database for MySQL. These functions are primarily concerned with security and administration. Azure manages these aspects of the database server itself.

### Benefits of Azure Database for MySQL

You get the following features with Azure Database for MySQL:

* High availability features built-in.
* Predictable performance.
* Easy scaling that responds quickly to demand.
* Secure data, both at rest and in motion.
* Automatic backups and point-in-time restore for the last 35 days.
* Enterprise-level security and compliance with legislation.

The system uses pay-as-you-go pricing so you only pay for what you use.

Azure Database for MySQL servers provides monitoring functionality to add alerts, and to view metrics and logs.

## What is Azure Database for MariaDB?

Azure Database for MariaDB is an implementation of the MariaDB database management system adapted to run in Azure. It's based on the MariaDB Community Edition.

The database is fully managed and controlled by Azure. Once you've provisioned the service and transferred your data, the system requires almost no additional administration.

### Benefits of Azure Database for MariaDB

Azure Database for MariaDB delivers:

* Built-in high availability with no additional cost.
* Predictable performance, using inclusive pay-as-you-go pricing.
* Scaling as needed within seconds.
* Secured protection of sensitive data at rest and in motion.
* Automatic backups and point-in-time-restore for up to 35 days.
* Enterprise-grade security and compliance.

## What is Azure Database for PostgreSQL?

If you prefer PostgreSQL, you can choose Azure Database for PostgreSQL to run a PaaS implementation of PostgreSQL in the Azure Cloud. This service provides the same availability, performance, scaling, security, and administrative benefits as the MySQL service.

Some features of on-premises PostgreSQL databases are not available in Azure Database for PostgreSQL. These features are mainly concerned with the extensions that users can add to a database to perform specialized tasks, such as writing stored procedures in various programming languages (other than pgsql, which is available), and interacting directly with the operating system. A core set of the most frequently used extensions is supported, and the list of available extensions is under continuous review.

Azure Database for PostgreSQL has two deployment options: Single-server and Hyperscale.

### **Azure Database for PostgreSQL single-server**

The single-server deployment option for PostgreSQL provides similar benefits as Azure Database for MySQL. You choose from three pricing tiers: Basic, General Purpose, and Memory Optimized. Each tier supports different numbers of CPUs, memory, and storage sizes—you select one based on the load you expect to support.

### **Azure Database for PostgreSQL Hyperscale (Citus)**

Hyperscale (Citus) is a deployment option that scales queries across multiple server nodes to support large database loads. Your database is split across nodes. Data is split into chunks based on the value of a partition key or sharding key. Consider using this deployment option for the largest database PostgreSQL deployments in the Azure Cloud.

### **Benefits of Azure Database for PostgreSQL**

Azure Database for PostgreSQL is a highly available service. It contains built-in failure detection and failover mechanisms.

Users of PostgreSQL will be familiar with the **pgAdmin** tool, which you can use to manage and monitor a PostgreSQL database. You can continue to use this tool to connect to Azure Database for PostgreSQL. However, some server-focused functionality, such as performing server backup and restore, are not available because the server is managed and maintained by Microsoft.

Azure Database for PostgreSQL servers records information about the queries run against databases on the server, and saves them in a database named azure\_sys. You query the query\_store.qs\_view view to see this information, and use it to monitor the queries that users are running. This information can prove invaluable if you need to fine-tune the queries performed by your applications.

## Migrate data to Azure

If you have existing MySQL, MariaDB, or PostgreSQL databases running on premises, and you want to move the data to a database running the corresponding data services in Azure, you can use the [Azure Database Migration Service (DMS)](https://docs.microsoft.com/en-us/azure/dms/tutorial-postgresql-azure-postgresql-online).

The Database Migration Service enables you to restore a backup of your on-premises databases directly to databases running in Azure Data Services. You can also configure replication from an on-premises database, so that any changes made to data in that database are copied to the database running in Azure Data Services. This strategy enables you to reconfigure users and applications to connect to the database in the cloud while the on-premises system is still active; you don't have to shut down the on-premises system while you transfer users to the cloud.

## Additional resources

* [Choose the right deployment option](https://docs.microsoft.com/en-us/azure/sql-database/sql-database-paas-vs-sql-server-iaas)
* [What is a single database in Azure SQL Database](https://docs.microsoft.com/en-us/azure/sql-database/sql-database-single-database)
* [What is Azure SQL Database managed instance?](https://docs.microsoft.com/en-us/azure/sql-database/sql-database-managed-instance)
* [Data Migration Assistant (DMA)](https://www.microsoft.com/download/details.aspx?id=53595)
* [Azure Database Migration Service (DMS)](https://docs.microsoft.com/en-us/azure/dms/tutorial-postgresql-azure-postgresql-online)
* [Choose the right data store](https://docs.microsoft.com/en-us/azure/architecture/guide/technology-choices/data-store-overview)
* [Transport Layer Security](https://docs.microsoft.com/en-us/windows-server/security/tls/transport-layer-security-protocol)
* [Transparent Data Encryption](https://docs.microsoft.com/en-us/sql/relational-databases/security/encryption/transparent-data-encryption)
* [Always Encrypted](https://docs.microsoft.com/en-us/sql/relational-databases/security/encryption/always-encrypted-database-engine)