



# Azure SQL Database Business Continuity

Module 3



## Learning Units covered in this Module

- Lesson 1: Business Continuity Features in Azure SQL Database
- Lesson 2: Disaster Recovery Features in Azure SQL Database
- Lesson 3: Backing up and Restoring Azure SQL Database

# Lesson 1: Business Continuity Features in Azure SQL Database

# Objectives

After completing this learning, you will be able to:

- Understand the various business continuity options within Azure SQL Database.
- Understand how to copy and export Azure SQL Databases.
- Understand how to perform a point-in-time restore.
- Understand how to perform a restore of a deleted database.



# Business Continuity Problems

Enabling the application to continuously operate during unplanned and planned disruptive events.

Disruption scenarios in general:

- Local hardware or software failures
- Data corruption or deletion typically caused by an application bug or human error.
- Datacenter outage, possibly caused by a natural disaster.
- Upgrade or maintenance errors.

# Basic (DTU), Standard (DTU), General Purpose (vCore) High Availability

Behaves like Failover Cluster Instance

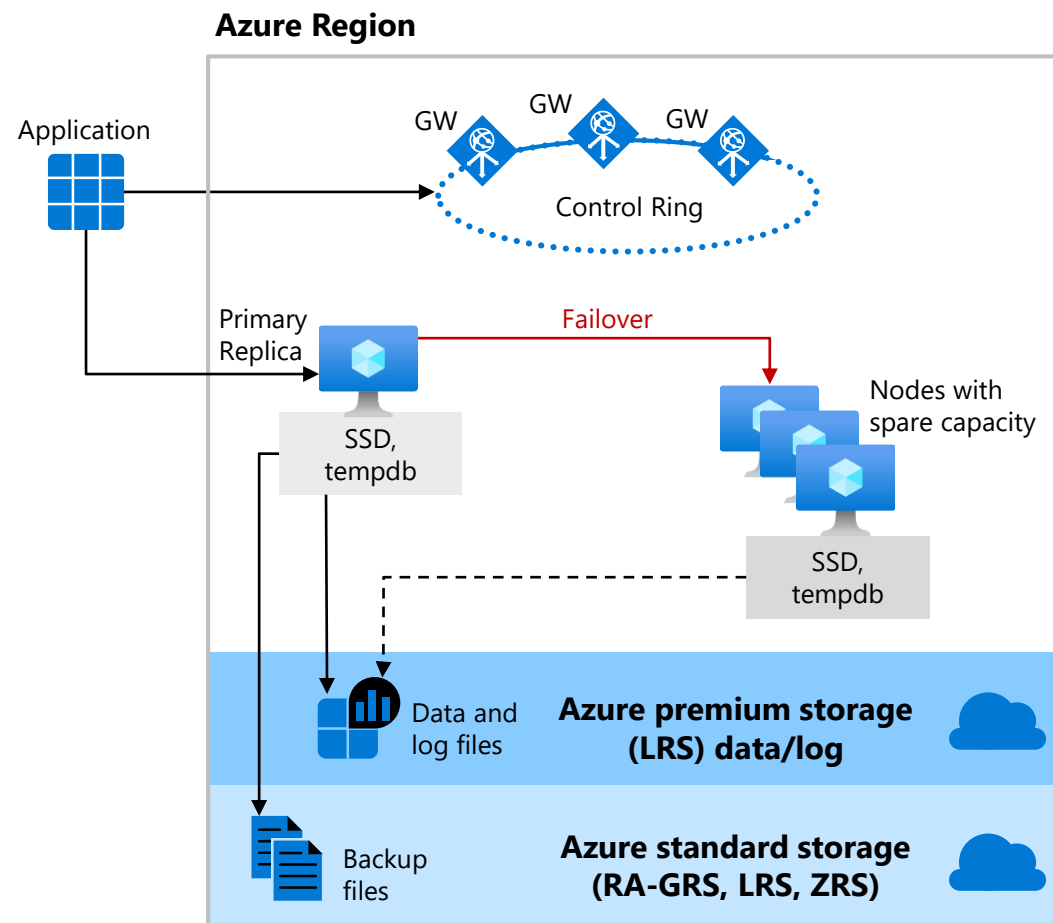
Remote storage provides data redundancy within a datacenter

Backup files are in a different location with geo-redundancy

Failover decisions based on SQL and Service Fabric

Recovery time depends on spare capacity

Connectivity redirection built-in



# Premium (DTU) and Business Critical (vCore) High Availability

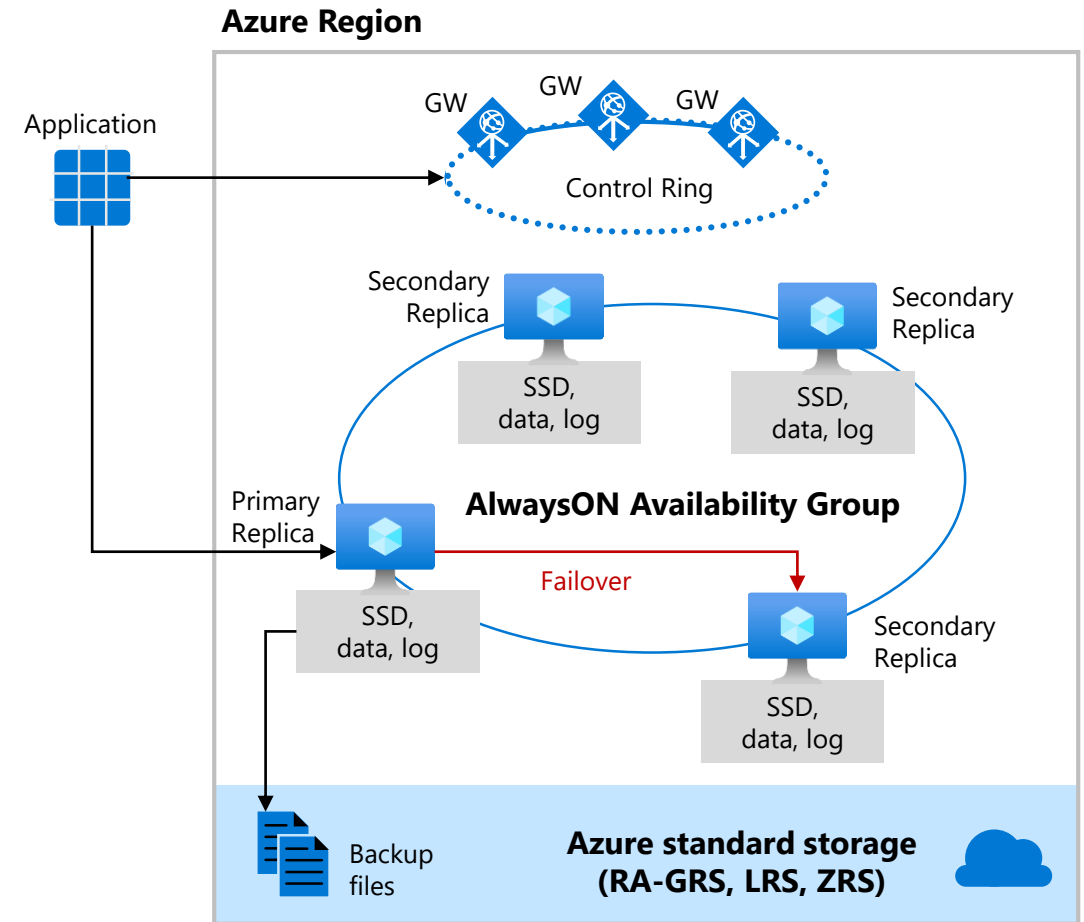
High availability is achieved by replicating both compute and storage to additional nodes.

High availability is implemented using a technology like SQL Server Always On Availability Groups.

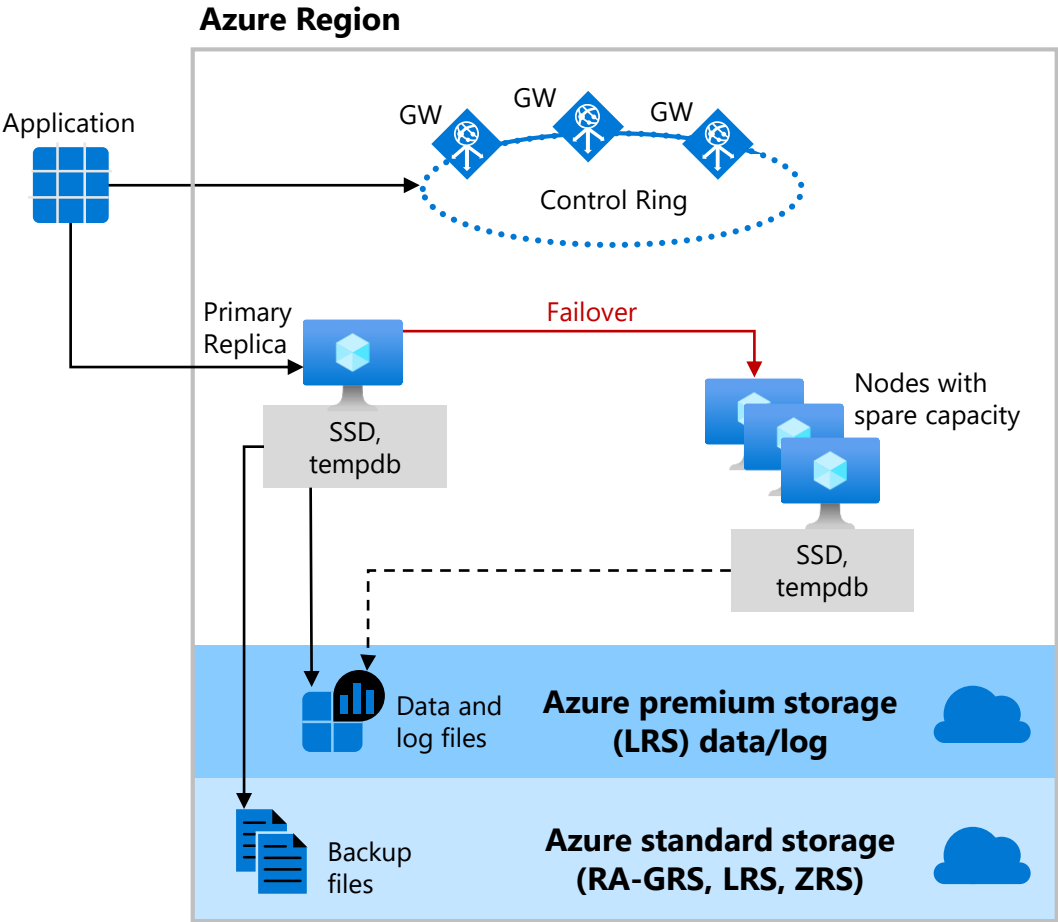
The cluster includes a single primary replica for read-write workloads, and up to three secondary replicas (compute and storage) containing copies of data.

The failover is initiated by the Azure Service Fabric.

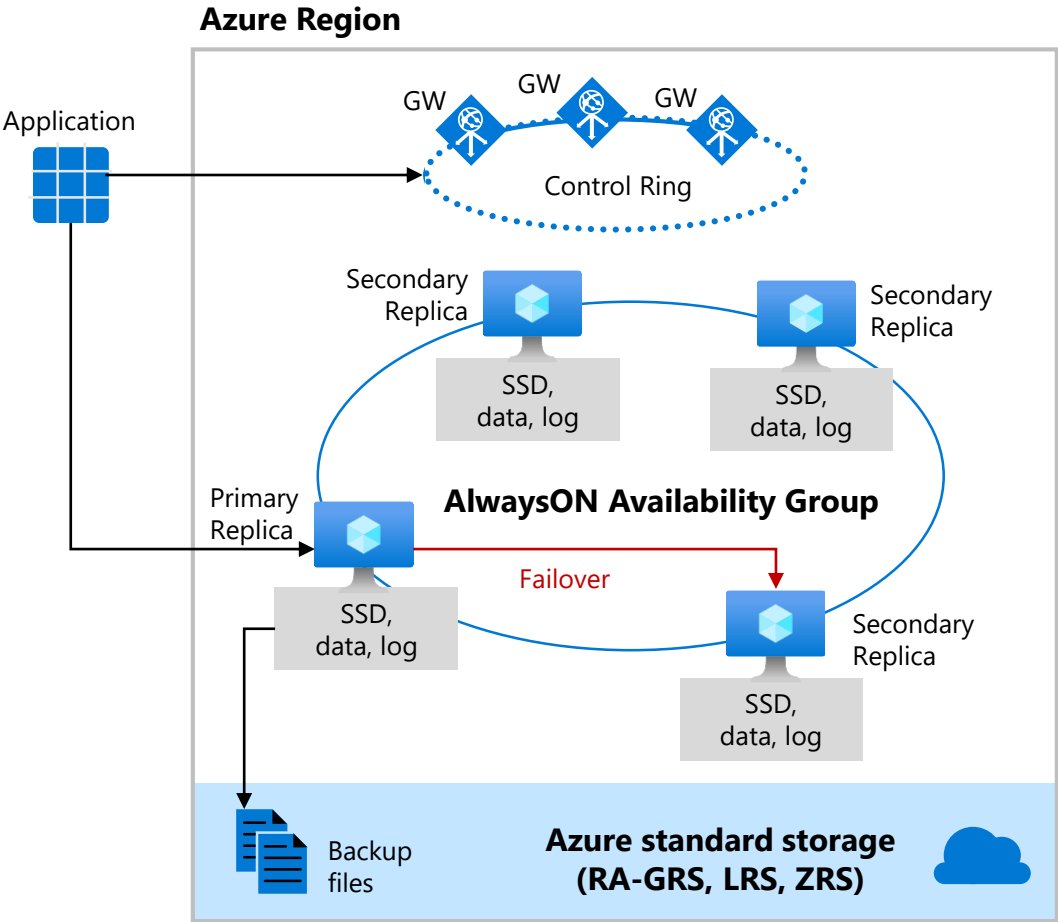
As an extra benefit, the premium availability model includes Read Scale-Out feature.



# Side by Side comparison



**General Purpose (GP) service tier**

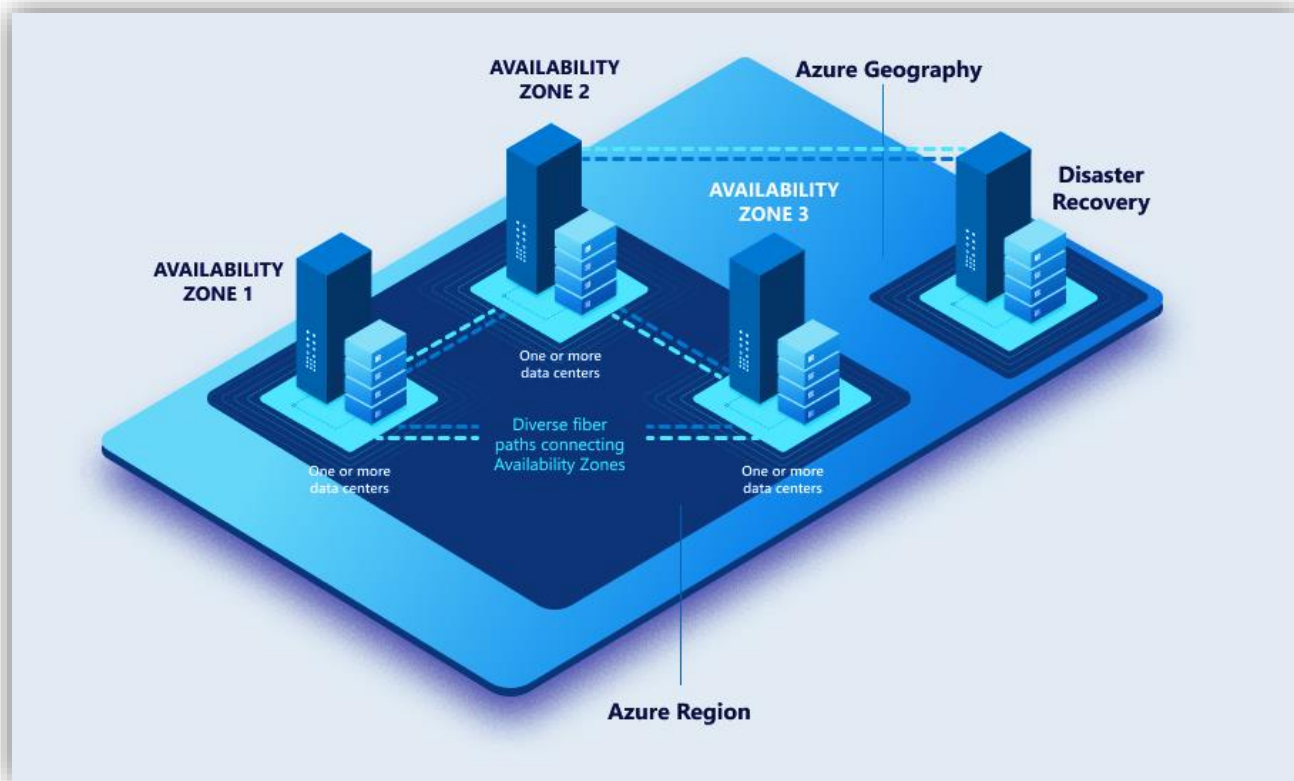


**Business Critical (BC) service tier**



# Backup storage redundancy

To enable high durability of backups several ways of replication are offered on instance creation.



The backups can be all located within

1. LRS: The same building (Local)
2. ZRS: Same region, different buildings (Zone)
3. GRS: Across paired regions (Geo)
4. GZRS: Different buildings AND paired regions (Geo-Zone)

# Zone redundant configuration

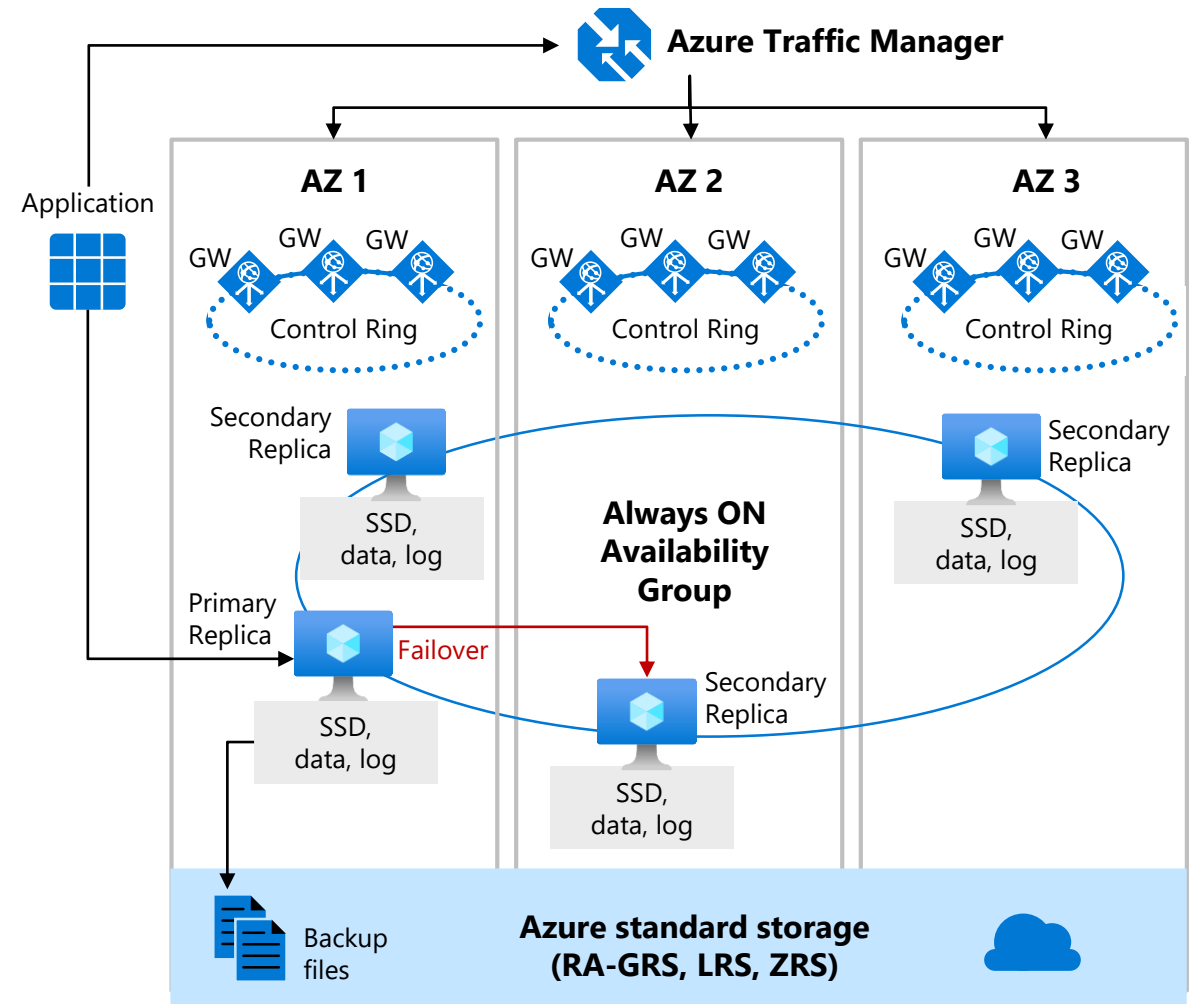
By default, the cluster of nodes for the premium availability model is created in the same datacenter.

SQL Database can place different replicas of the Business-Critical database to different availability zones in the same region.

The routing is controlled by Azure Traffic Manager (ATM).

The zone redundant databases have replicas in different datacenters with some distance between them, the increased network latency may impact the performance.

Zone redundant configurations are currently only supported in the Premium or Business Critical tiers



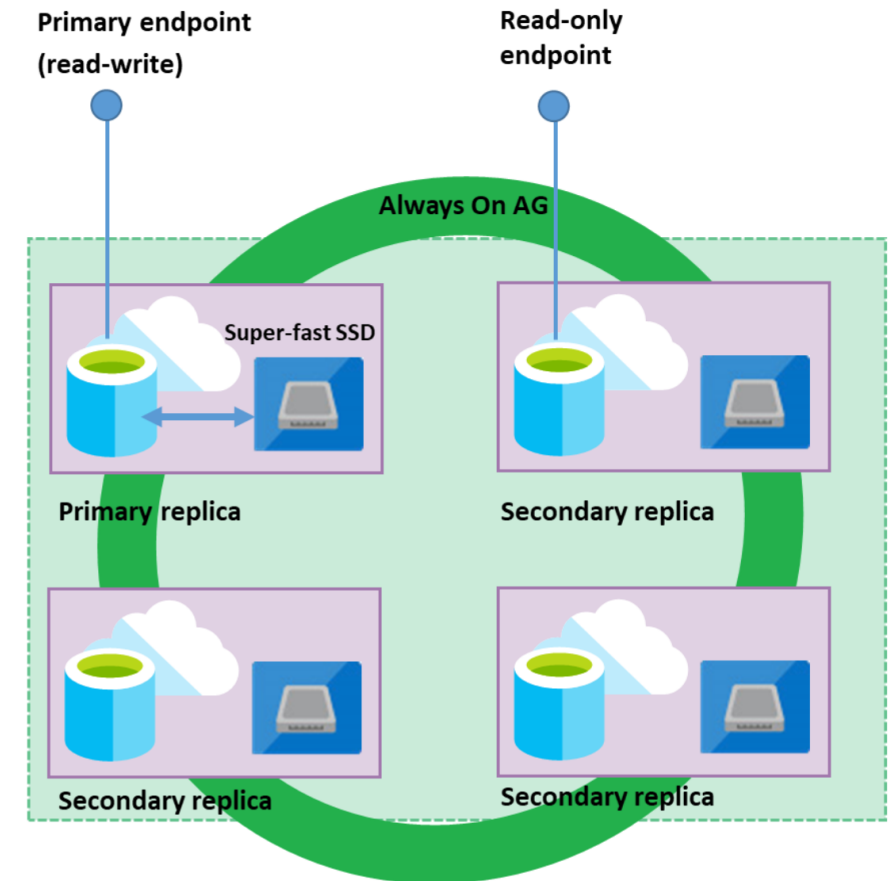
# Read Scale-Out

Each database in the Premium and Business Critical service tiers is automatically provisioned with several AlwaysON replicas to support the availability SLA. These replicas are provisioned with the same compute size as the read-write.

Read Scale-Out redirects the read-only client connections to one of the ready-only replicas available instead of sharing the read-write replica.

Effectively isolate the read-only workload from the main read-write workload and doubles the compute capacity of the database or elastic pool at no additional charge.

This is ideal to scale-out for complex analytical workloads without affecting the primary OLTP workload.



# Service Level Agreement (SLA)

Service tier	Single zone SLA	Multiple zones SLA
Basic, Standard, General Purpose	99.99%	N/A
Premium, Business critical	99.99%	99.995%

Business continuity	Service tier	SLA
Recovery point objective (RPO)	Business critical with Geo-DR	5 sec
Recovery Time Objective (RTO)	Business critical with Geo-DR	30 sec

[SLA for Azure SQL Database](#)

[SLA for Azure SQL Managed Instance](#)

Questions?

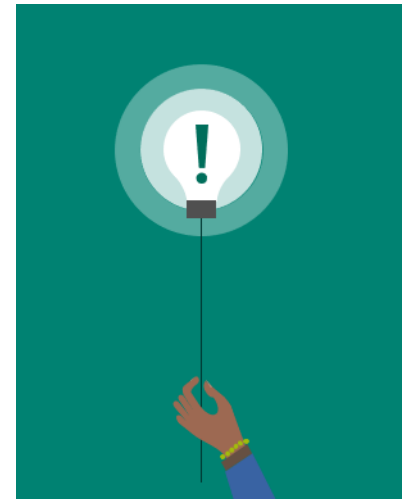


## Lesson 2: Disaster Recovery Features in Azure SQL Database

# Objectives

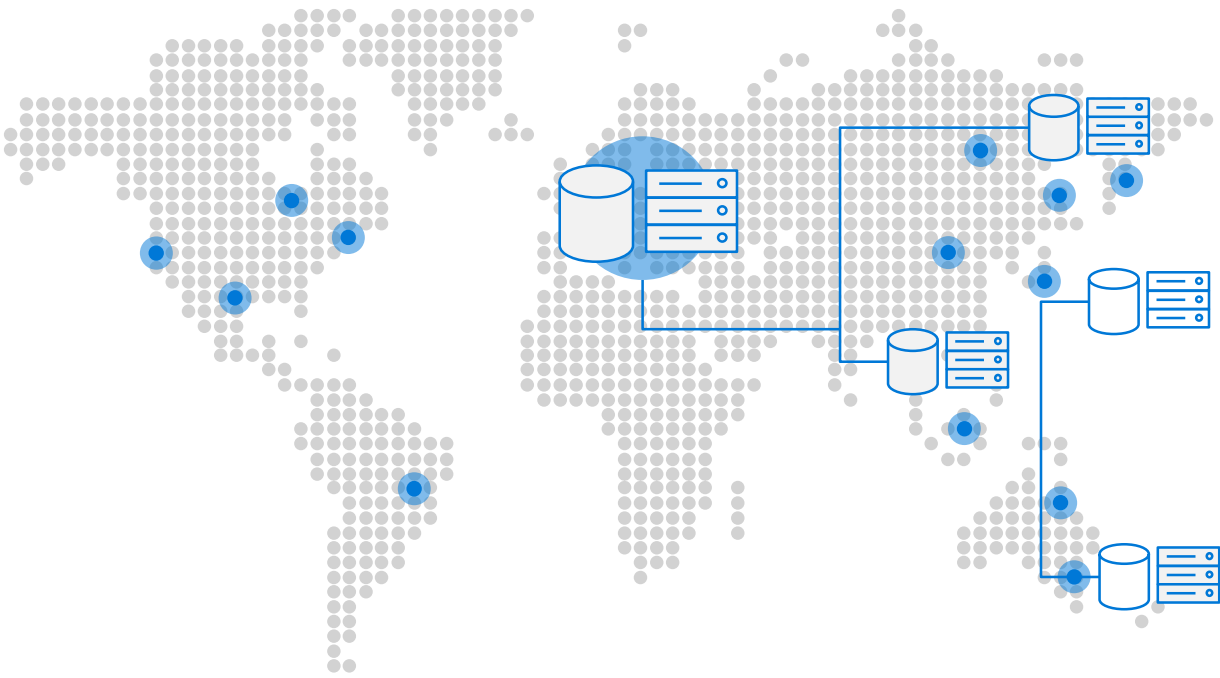
After completing this learning, you will be able to:

- Understand the various disaster recovery options within Azure SQL Database



# Active Geo-replication

Service levels	Basic, standard, premium Self service
Readable secondaries	Up to 4
Regions available	Any Azure region
Replication	Automatic, asynchronous
Manageability tools	REST API, PowerShell, or Azure Portal
Recovery time objective (RTO)	<1 hour
Recovery point objective	<5 minutes
Failover	On demand



Up to 4 secondaries



# Active geo-replication capabilities

Asynchronous  
Replication

Readable secondary  
databases

Multiple Readable  
Secondary Replicas

Configurable  
performance level of  
the secondary database

User-controlled failover  
and failback

Keeping credentials  
and firewall rules in  
sync

# Failover groups extend geo-replication

Enable geo-replication for a group of databases within a server.

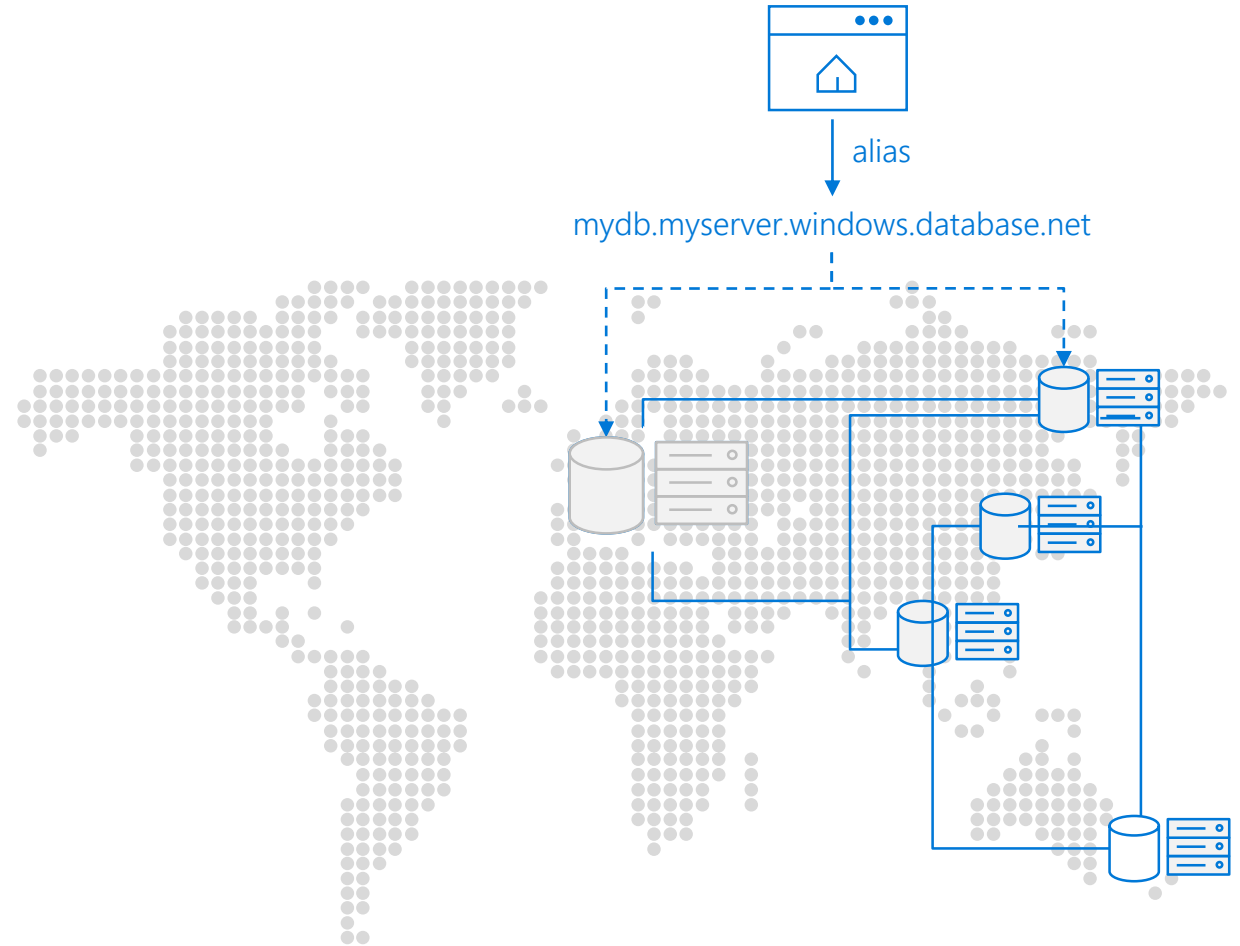
Automatically or manually failover a group of databases.

Available for all service tiers.

Configure the auto-failover policy that best meets your application needs.

Usage of and listener end-points.

DNS record is automatically updated.



\*Currently in private preview - Microsoft Confidential – Shared Under NDA Only

# Auto-failover group capabilities

Failover group

Failover group listener

Automatic Failover Policy

Grace Period with Data  
Loss

# Active geo-replication vs auto-failover groups

	Geo-replication (Database)	Auto-failover groups (Server)
Automatic failover	No	Yes
Fail over multiple databases simultaneously	No	Yes
Update connection string after failover	Yes	No
Managed instance supported	No	Yes
Can be in same region as primary	Yes	No
Multiple replicas	Yes	No
Supports read-scale	Yes	Yes

# Demonstration

## Geo Replication

- Setup Geo Replication for an Azure SQL Database.



# Configure Geo Replication for an Azure SQL Database

- **Exercise 1:** Create a Failover Group.
- **Exercise 2:** Verify the functionality of the secondary.
- **Exercise 3:** Perform a Failover.



# Lesson 3: Backing up and Restoring Azure SQL Database

# Objectives

After completing this learning, you will be able to:

- Understand backing up and restoring Azure SQL Databases
- Setting Point-in-time-Retention (PITR) Backup Policies
- Setting Long-Term Retention (LTR) Backup Policies
- Performing Point-in-time Restore operations.

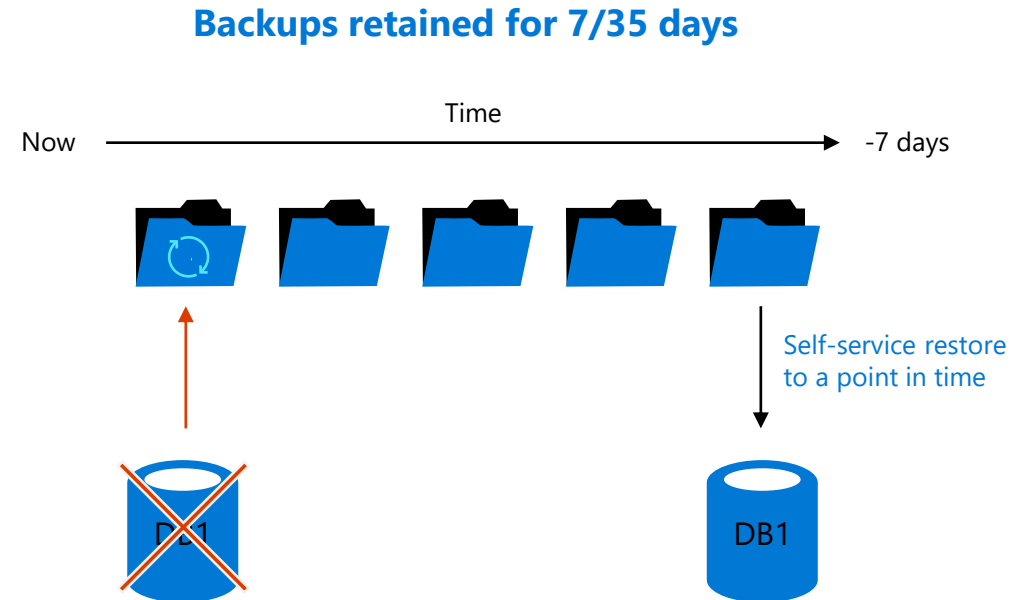




# Backup and restore

## Auto backups and Point in Time Restore (PITR)

- Full Database backup once a week
- Differential Backups every 12-24 hours
- Log Backups every 5-10 minutes
- Backup files on Azure storage with RA-GRS replicated
  - Can optionally select LRS or ZRS
- Backup Integrity checks
- Restore to new database
- Long-term retention (up to 10 years) of backups
- Geo-restore of databases if primary region down
- Restore backups of deleted databases



# Setting Backup Policies

Home > Resource groups > jdSQLRG > jdsqlazure

**jdsqlazure | Backups** ☆ ...

SQL server

Search

**Data management**

- Backups
- Deleted databases
- Failover groups
- Import/Export history

**Security**

- Networking
- Microsoft Defender for Cloud
- Transparent Data Encryption
- Identity
- Auditing

**Intelligent Performance**

Available backups **Retention policies**

Configure and manage your automated backup retention policies. Long-term retention policies enable you to keep full backups for up to 10 years.

Search for a database

**Databases in the Basic tier are limited to a 7 day retention policy.**

Database	PITR	Differential backup frequency
jdsqldb	7 Days	24 Hours

### Configure policies

SQL server

#### Point-in-time-restore

Specify how long you want to keep your point-in-time backups. [Learn more](#)

How many days would you like PITR backups to be kept? 7

#### Differential backup frequency

Specify how often you want differential backups to be taken. [Learn more](#)

Take a differential backup every:

24 Hours

#### Long-term retention


Specify how long you want to keep your long-term retention backups. You may choose to keep yearly backups for up to 10 years. [Learn more](#)

Weekly LTR Backups

Keep weekly backups for:

6 Week(s)

# Automatic Backups

- 
- Uses SQL Server technology to create full, differential, and transaction log backups.
  - Transaction log backups, with full and differential backups, allow you to restore a database to a specific point-in-time to the same server that hosts the database.
  - When you restore a database, the service figures out which full, differential, and transaction log backups need to be restored.

# Azure SQL Database Backup Retention Periods

All Azure SQL databases (single, pooled, and managed instance databases) have a default backup retention period of **seven** days.

You can change backup retention period up to 35 days.

If you delete a database, SQL Database will keep the backups in the same way it would for an online database.

If you need to keep the backups for longer than the maximum retention period, you can modify the backup properties to add one or more long-term retention periods to your database.

The point-in-time backups are geo-redundant and protected by Azure Storage cross-regional replication.  
[How long are backups kept.](#)

# How to change backup retention period

You can change the default PITR backup retention period using the Azure portal, PowerShell, or REST API.

The following examples illustrate how to change PITR retention to 28 days.

## PowerShell

```
Set-AzSqlDatabaseBackupShortTermRetentionPolicy -ResourceGroupName resourceGroup -ServerName testserver -  
DatabaseName testDatabase -RetentionDays 28
```

## REST

```
PUT https://management.azure.com/subscriptions/00000000-1111-2222-3333-  
444444444444/resourceGroups/resourceGroup/providers/Microsoft.Sql/servers/testserver/databases/testDatabase/back  
upShortTermRetentionPolicies/default?api-version=2017-10-01-preview
```

The supported values are: 7, 14, 21, 28 or 35 days.

# Extending the Retention Period

You can configure a single or a pooled database with a long-term backup retention policy (LTR) to automatically retain the database backups in separate Azure Blob storage containers for up to 10 years.

You can then recover a database using these backups using the Azure portal or PowerShell.

Deleting LTR backup is non-reversible. To delete an LTR backup after the server has been deleted you must have Subscription scope permission.

# How SQL Database long-term retention works

Long-term backup retention leverages the automatic SQL Database backups created to enable point-time restore (PITR).

Specify for each SQL database how frequently you need to copy the backups to the long-term storage.

- Weekly backup retention (W)
- Monthly backup retention (M)
- Yearly backup retention (Y)
- Week of year (WeekOfYear)

$W=0, M=0, Y=5, \text{WeekOfYear}=3$

The 3rd full backup of each year will be kept for 5 years.

$W=0, M=3, Y=0$

The first full backup of each month will be kept for 3 months.

$W=12, M=0, Y=0$

Each weekly full backup will be kept for 12 weeks.

# Demonstration

Configure the long-term retention and view backups in long-term retention.





# Copy & Export

Copy

- Copies an Azure SQL Database to another DB on same or different Server.
- Essentially Creates a snapshot.

Export

- Creates BACPAC.
- Use Portal, SSMS, PowerShell or SQLPackage Utility.

Neither of these were really designed for ongoing backup operations.

# Point In Time Restore

The database can be restored to any service tier or performance level

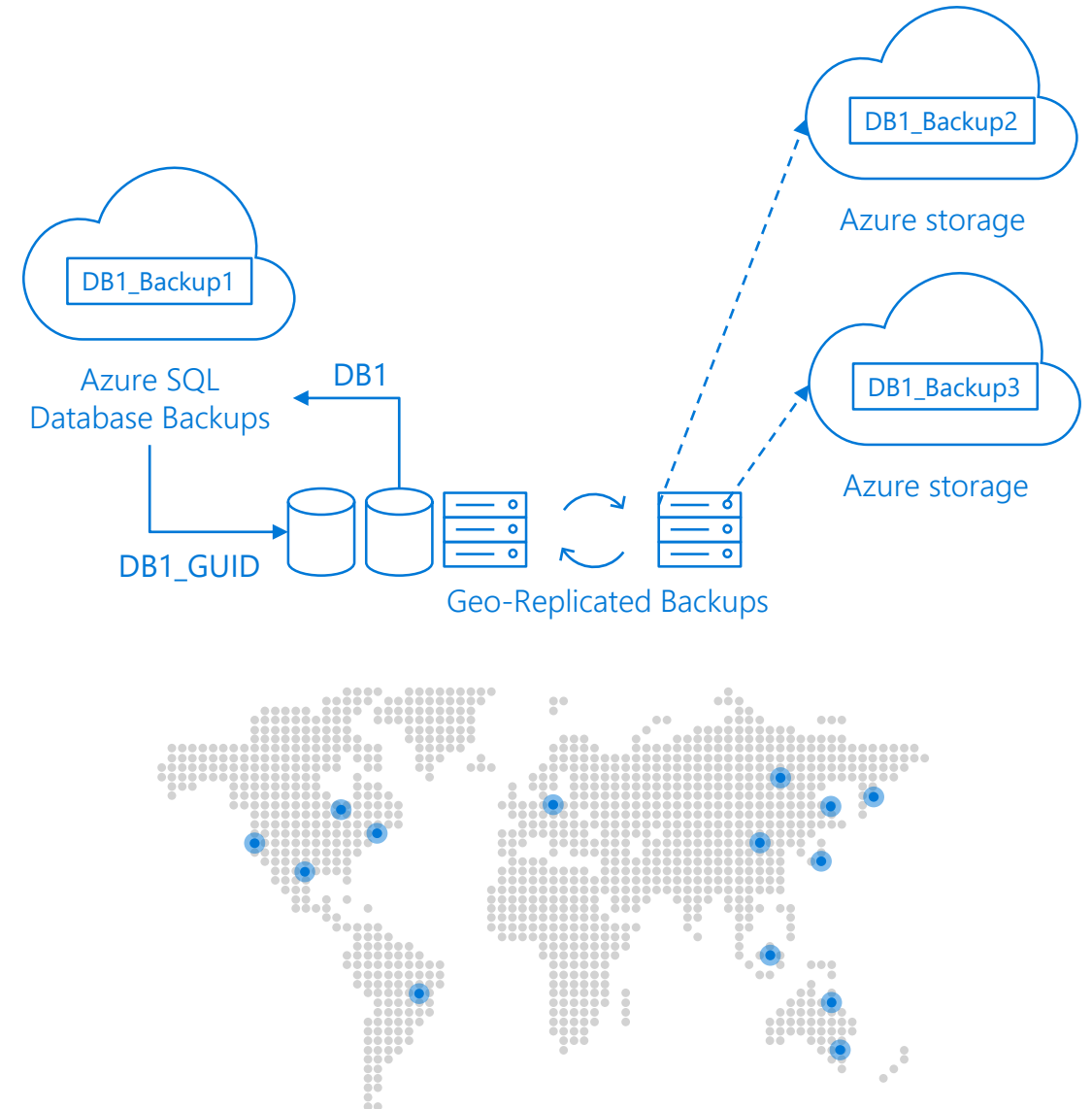
- Creates a new database in the same logical server.

## Database Replacement

- Rename the original database and then give the restored database the original name using the ALTER DATABASE command in T-SQL.

## Data Recovery

- Write and execute the necessary data recovery scripts to extract data from the restored database to the original database.



# Geo-Restore

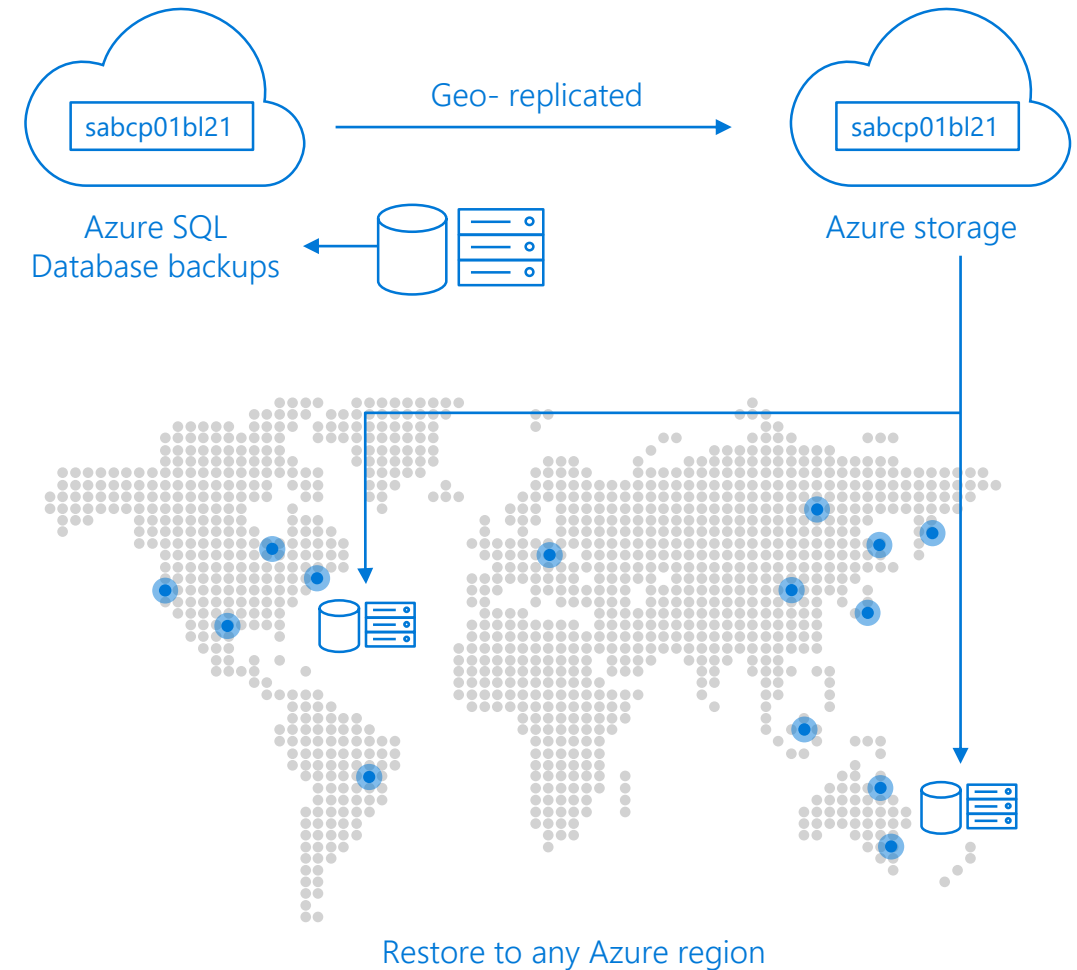
Restores last daily backup to any Azure region.

Built on geo-redundant Azure Storage.

$RTO \geq 24h$ ,  $RPO = 24h$

Database URL will change after restore.

Point-in-time restore on a geo-secondary is not currently supported.



# Recover an Azure SQL database by using automated database backups

By default, Azure SQL Database backups are stored in **geo-replicated blob storage** (RA-GRS storage type).

The following options are available for database recovery by using automated database backups. You can:

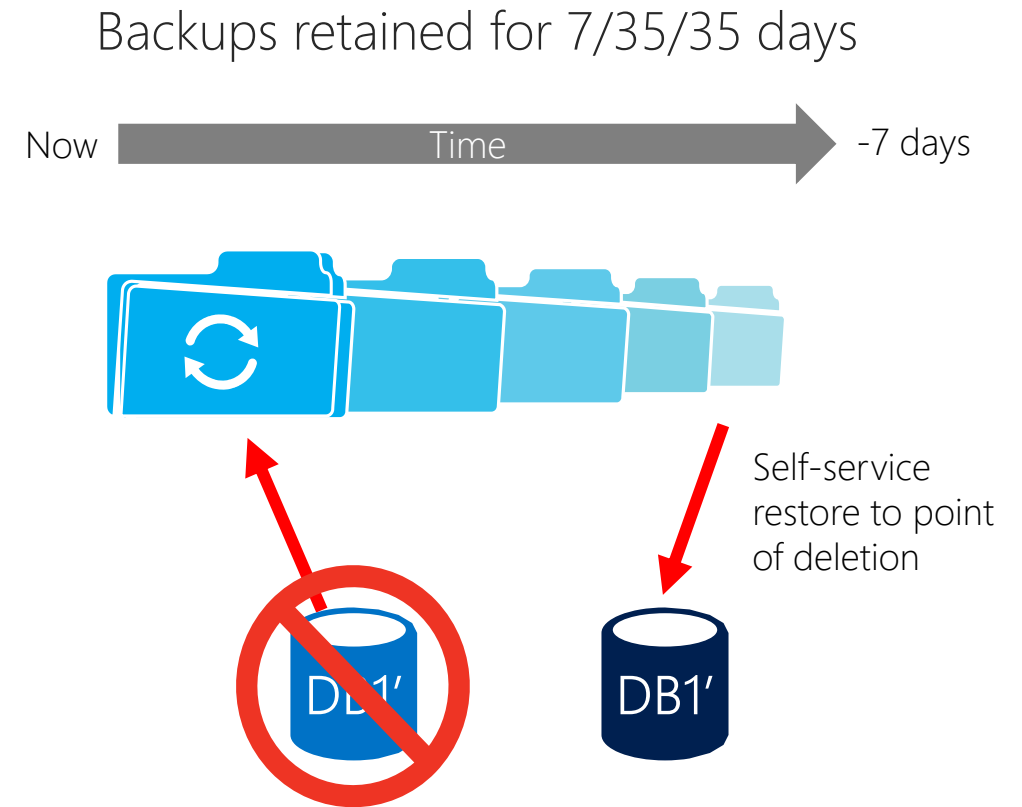
- Create a **new database** on the **same SQL Database server**, recovered to a specified point in time within the retention period.
- Create a **database** on the **same SQL Database server**, recovered to the deletion time for a deleted database.
- Create a **new database** on any SQL Database server **in the same region**, recovered to the point of the most recent backups.
- Create a **new database** on any SQL Database server **in any other region**, recovered to the point of the most recent replicated backups.

If you configured backup long-term retention, you could also create a new database from any long-term retention backup on any SQL Database server.

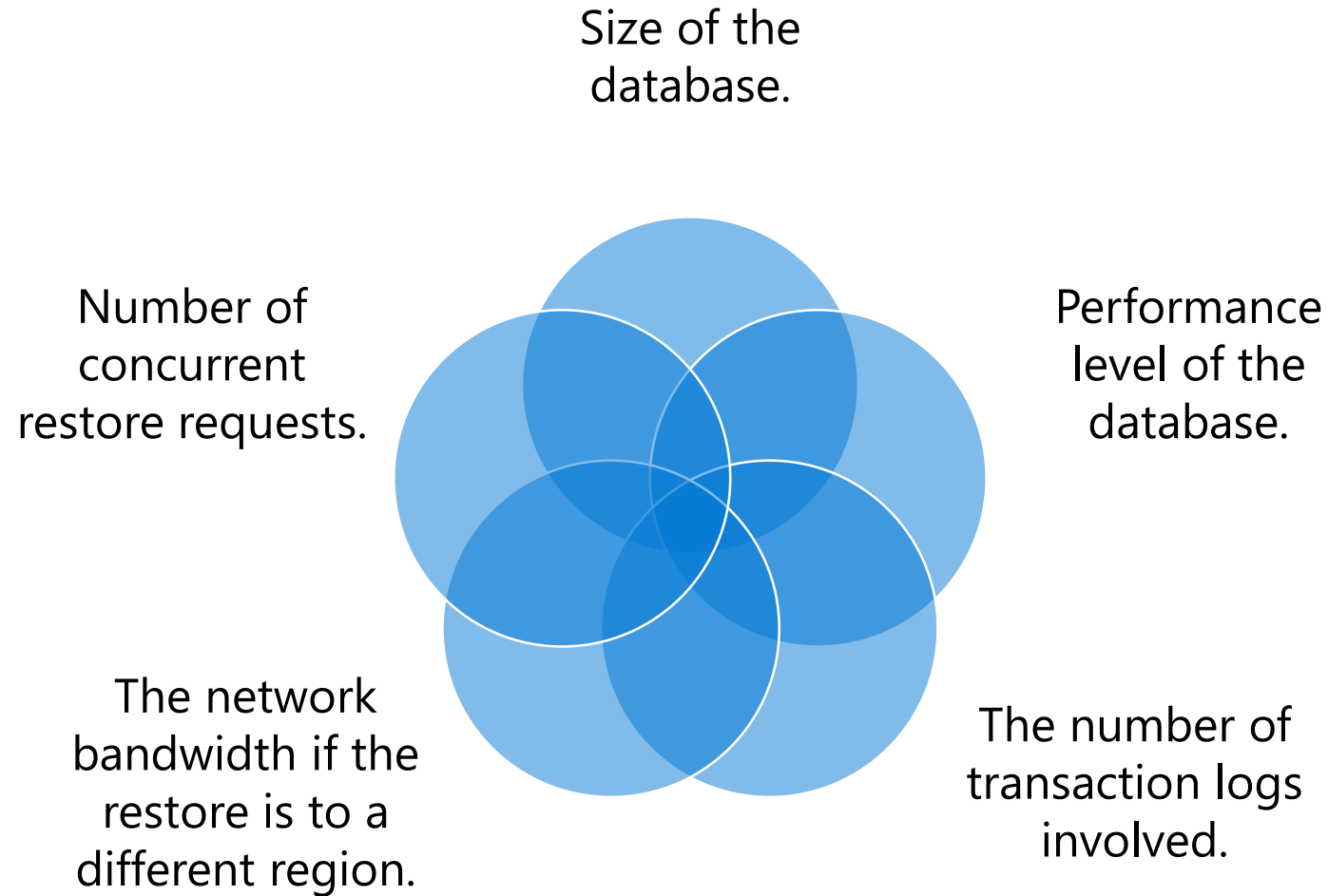
# Restore Deleted Database

Recovery after accidental database deletion:

- Restores the database to the point of deletion. (earlier backups are deleted).
- Creates a new database on the server used by the original database.
- You can choose to failover to the restored database or use scripts to recover data.



# Factors Affecting Recovery Time



# Demonstration

## Point in Time Restore

- Perform a point in time restore of a database.



# Point in time restore of an Azure SQL Database

- **Exercise 1:** Perform a point in time restore over the original database.
- **Exercise 2:** Rename old and new databases.





Questions?



# Knowledge Check

True or false: Daily and weekly backups of Azure SQL databases are automatically uploaded to geo-redundant Azure Storage.

True or false: When performing a point-in-time restore, you can choose to overwrite the source database?

Questions?



# Knowledge Check

True or false: Can you configure both Synchronous and Asynchronous Replication for the replicas in Geo Replication?

True or false: Both primary and secondary databases are required to have the same service tier.

What is the Grace period with data loss on the Failover Group?

# Module Summary

