Recover an Azure SQL database using automated database backups

SQL Database provides these options for database recovery using automated database backups and backups in long-term retention. You can restore from a database backup to:

- A new database on the same logical server recovered to a specified point in time within the retention period.
- A database on the same logical server recovered to the deletion time for a deleted database.
- A new database on any logical server in any region recovered to the point of the most recent daily backups in geo-replicated blob storage (RA-GRS).

Important

You cannot overwrite an existing database during restore.

A restored database incurs an extra storage cost under the following conditions:

- Restore of P11–P15 to S4-S12 or P1–P6 if the database max size is greater than 500 GB.
- Restore of P1–P6 or PRS1–PRS6 to S4-S12 if the database max size is greater than 250 GB.

The extra cost is because the max size of the restored database is greater than the amount of storage included for the performance level, and any extra storage provisioned above the included amount is charged extra. For pricing details of extra storage, see the SQL Database pricing page. If the actual amount of space used is less than the amount of storage included, then this extra cost can be avoided by reducing the database max size to the included amount. For more information about database storage sizes and changing the database maximum size, see single database resource limits.

Note

Automated database backups are used when you create a database copy.

Recovery time

The recovery time to restore a database using automated database backups is impacted by several factors:

- The size of the database

- The performance level of the database
- The number of transaction logs involved
- The amount of activity that needs to be replayed to recover to the restore point
- The network bandwidth if the restore is to a different region
- The number of concurrent restore requests being processed in the target region.

  For a very large and/or active database, the restore may take several hours. If there is prolonged outage in a region, it is possible that there are large numbers of geo-restore requests being processed by other regions. When there are many requests, the recovery time may increase for databases in that region. Most database restores complete within 12 hours.

There is no built-in functionality to do bulk restore. The Azure SQL Database: Full Server Recovery script is an example of one way of accomplishing this task.

Important

To recover using automated backups, you must be a member of the SQL Server Contributor role in the subscription or be the subscription owner. You can recover using the Azure portal, PowerShell, or the REST API. You cannot use Transact-SQL.

Point-in-time restore

You can restore an existing database to an earlier point in time as a new database on the same logical server using the Azure portal, PowerShell, or the REST API.

Tip

For a sample PowerShell script showing how to perform a point-in-time restore of a database, see **Restore a SQL database using PowerShell**.

The database can be restored to any service tier or performance level, and as a single database or into an elastic pool. Ensure you have sufficient resources on the logical server or in the elastic pool to which you are restoring the database. Once complete, the restored database is a normal, fully accessible, online database. The restored database is charged at normal rates based on its service tier and performance level. You do not incur charges until the database restore is complete.

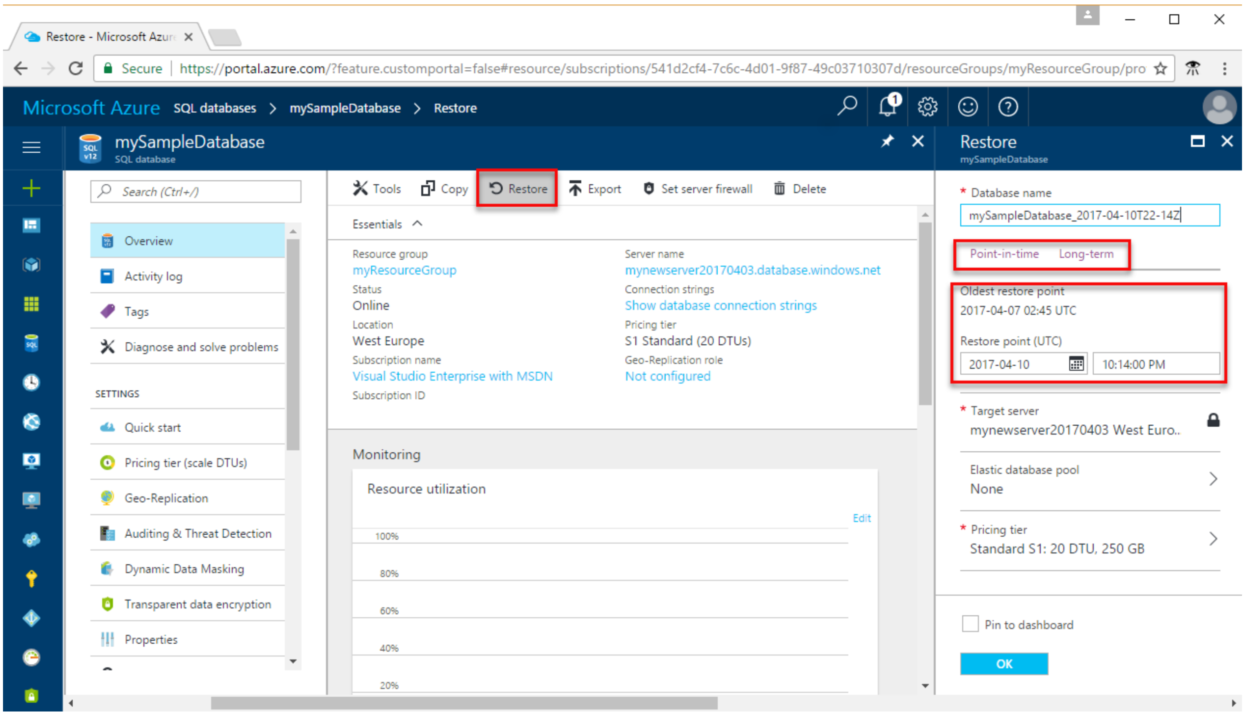
You generally restore a database to an earlier point for recovery purposes. When doing so, you can treat the restored database as a replacement for the original database or use it to retrieve data from and then update the original database.

- Database replacement: If the restored database is intended as a replacement for the original database, you should verify the performance level and/or service tier are appropriate and scale the database if necessary. You can rename the original database and then give the restored database the original name using the ALTER DATABASE command in T-SQL.

- Data recovery: If you plan to retrieve data from the restored database to recover from a user or application error, you need to write and execute the necessary data recovery scripts to extract data from the restored database to the original database. Although the restore operation may take a long time to complete, the restoring database is visible in the database list throughout the restore process. If you delete the database during the restore, the restore operation is canceled and you are not charged for the database that did not complete the restore.

Azure portal

To recover to a point in time using the Azure portal, open the page for your database and click Restore on the toolbar.


Restore - Microsoft Azure
Secure
https://portal.azure.com/?feature.customportal=false#resource/subscriptions/541d2cf4-7c6c-4d01-9f87-49c03710307d/resourceGroups/myResourceGroup/pro
Microsoft Azure
SQL databases > mySampleDatabase > Restore
mySampleDatabase
SQL database
Search (Ctrl+/)
Overview
Activity log
Tags
Diagnose and solve problems
SETTINGS
Quick start
Pricing tier (scale DTUs)
Geo-Replication
Auditing & Threat Detection
Dynamic Data Masking
Transparent data encryption
Properties
Tools
Copy
Restore
Export
Set server firewall
Delete
Essentials
Monitoring
Resource utilization
Edit
100%
80%
60%
40%
20%
Restore
mySampleDatabase
Point-in-time
Long-term
OK

Resource group: myResourceGroup
Status: Online
Location: West Europe
Subscription name: Visual Studio Enterprise with MSDN
Subscription ID
Server name: mynewserver20170403.database.windows.net
Connection strings: Show database connection strings
Pricing tier: S1 Standard (20 DTUs)
Geo-Replication role: Not configured

* Database name: mySampleDatabase_2017-04-10T22-14Z
Oldest restore point: 2017-04-07 02:45 UTC
Restore point (UTC): 2017-04-10 10:14:00 PM
* Target server: mynewserver20170403 West Euro..
Elastic database pool: None
* Pricing tier: Standard S1: 20 DTU, 250 GB
□ Pin to dashboard

Deleted database restore

You can restore a deleted database to the deletion time for a deleted database on the same logical server using the Azure portal, PowerShell, or the REST (createMode=Restore). You can restore a deleted database to an earlier point in time during the retention using PowerShell.

Tip

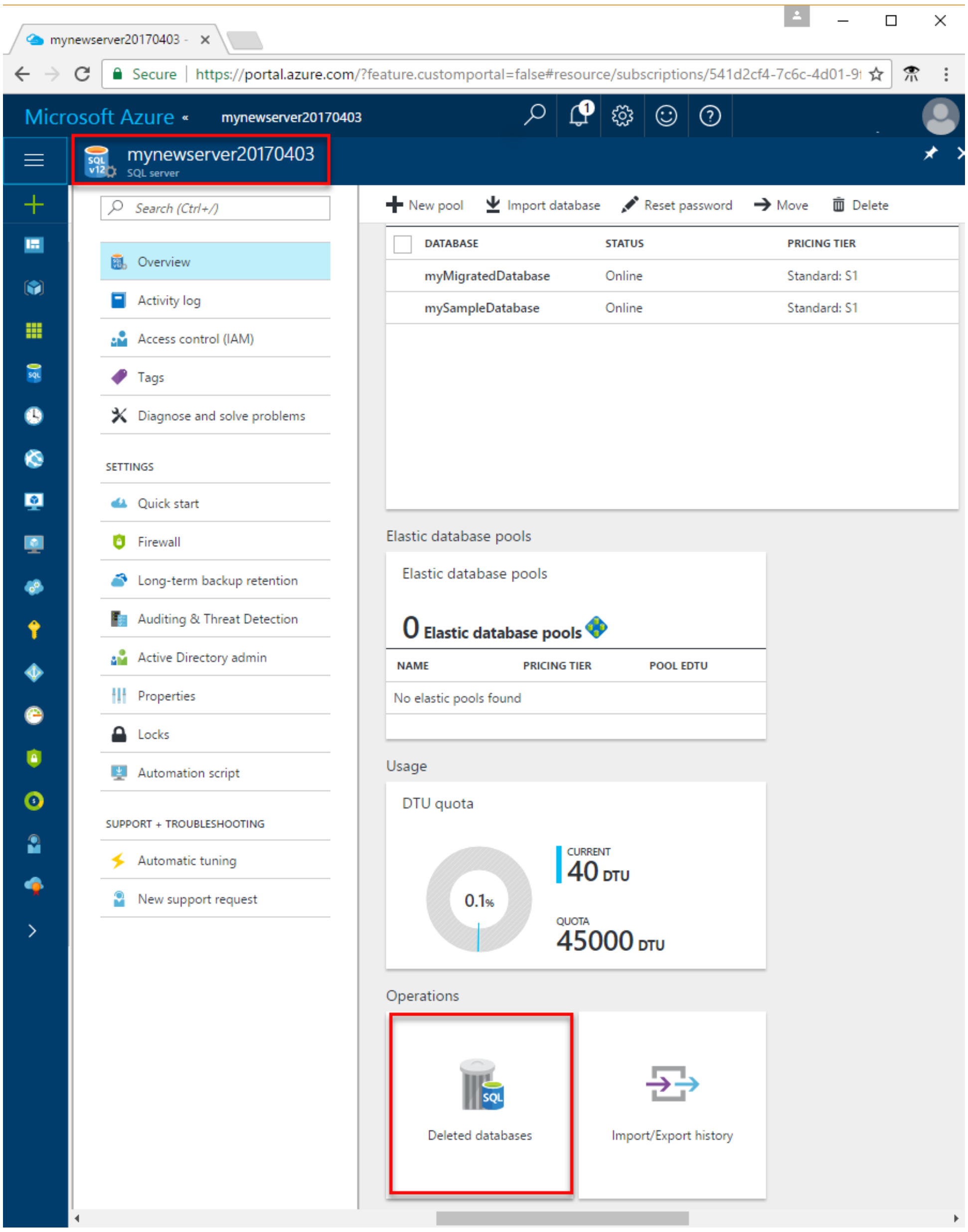
For a sample PowerShell script showing how to restore a deleted database, see **Restore a SQL database using PowerShell**.

Important

If you delete an Azure SQL Database server instance, all its databases are also deleted and cannot be recovered. There is currently no support for restoring a deleted server.

Azure portal

To recover a deleted database during its retention period using the Azure portal, open the page for your server and in the Operations area, click Deleted databases.

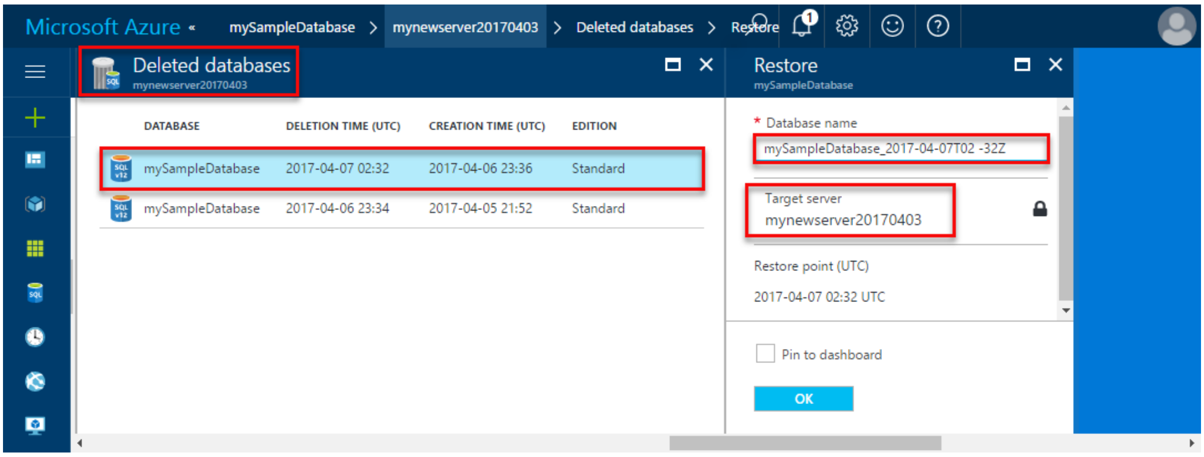


DATABASE	STATUS	PRICING TIER
myMigratedDatabase	Online	Standard: S1
mySampleDatabase	Online	Standard: S1

Elastic database pools

0 Elastic database pools

NAME	PRICING TIER	POOL EDTU
No elastic pools found		


Microsoft Azure
mySampleDatabase > mynewserver20170403 > Deleted databases > Restore

Deleted databases
mynewserver20170403

DATABASE	DELETION TIME (UTC)	CREATION TIME (UTC)	EDITION
mySampleDatabase	2017-04-07 02:32	2017-04-06 23:36	Standard
mySampleDatabase	2017-04-06 23:34	2017-04-05 21:52	Standard

Restore
mySampleDatabase

* Database name
mySampleDatabase_2017-04-07T02 -32Z

Target server
mynewserver20170403

Restore point (UTC)
2017-04-07 02:32 UTC

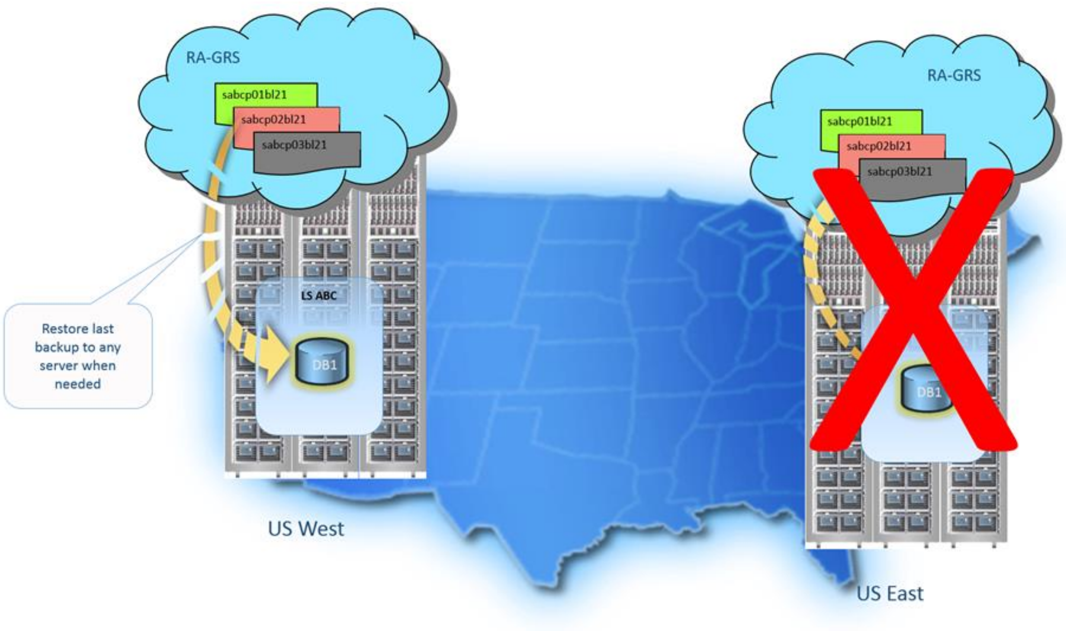
□ Pin to dashboard

OK

Geo-restore

You can restore a SQL database on any server in any Azure region from the most recent geo-replicated full and differential backups. Geo-restore uses a geo-redundant backup as its source and can be used to recover a database even if the database or datacenter is inaccessible due to an outage.

Geo-restore is the default recovery option when your database is unavailable because of an incident in the region where the database is hosted. If a large-scale incident in a region results in unavailability of your database application, you can restore a database from the geo-replicated backups to a server in any other region. There is a delay between when a differential backup is taken and when it is geo-replicated to an Azure blob in a different region. This delay can be up to an hour, so, if a disaster occurs, there can be up to one hour data loss. The following illustration shows restore of the database from the last available backup in another region.



Tip

For a sample PowerShell script showing how to perform a geo-restore, see Restore a SQL database using PowerShell.

Point-in-time restore on a geo-secondary is not currently supported. Point-in-time restore can be done only on a primary database. For detailed information about using geo-restore to recover from an outage, see Recover from an outage.

Important

Recovery from backups is the most basic of the disaster recovery solutions available in SQL Database with the longest Recovery Point Objective (RPO) and Estimate Recovery Time (ERT). For solutions using Basic databases, geo-restore is frequently a reasonable DR solution with an ERT of 12 hours. For solutions using larger Standard or Premium databases that require shorter recovery times, you should consider using active geo-replication. Active geo-replication offers a much lower RPO and ERT as it only requires you initiate a failover to a continuously replicated secondary. For more information on business continuity choices, see Overview of business continuity.

Azure portal

To geo-restore a database during its retention period using the Azure portal, open the SQL Databases page and then click Add. In the Select source text box, select Backup. Specify the backup from which to perform the recovery in the region and on the server of your choice.