


Scaling Considerations


Last updated by | Charlene Wang | Jan 3, 2023 at 12:30 AM PST


Scaling Considerations

Move to Azure Resource Manager REST APIs for Azure SQL Database



[Move to Azure Resource Manager REST APIs for Azure SQL Database](#) | [Azure updates](#) | [Microsoft Azure](#) 

[Azure Resource Manager](#)  is our cloud infrastructure stack that fully replaces the classic Azure Service Manager deployment model. As a result, support for Service Manager REST APIs for Azure SQL Database will be retired on December 1, 2019. As with all changes of this type, we're providing 12 months' notice so you have adequate time to update your services that use Service Manager APIs for SQL Database to Resource Manager APIs.

To continue using Azure SQL Database, please move to Resource Manager by November 30, 2019. We encourage all customers who are still using Service Management APIs to make the switch soon to take advantage of the additional benefits of Resource Manager, which include resource grouping and tags, a streamlined deployment and management process, and fine-grained access control using role-based access control. For more information on the differences between Azure Resource Manager and Service Management, please see the [TechNet Blog](#) .



For more information on Service Management and Resource Manager APIs for Azure SQL Database, please see our [Azure SQL Database REST APIs](#) .

Scaling Operation Slow or Stuck





Scaling operations can take anywhere from a few minutes to several hours depending on the service tiers involved and sometimes the size of the database. For information on factors that affect how long a scaling operation runs and rules of thumb for estimating how long a scaling operation should take, see the following. • [Scaling a single database](#)  • [Scaling an elastic pool](#) 

Additional considerations when changing service tier or rescaling compute size for single DB

[Scale single database resources - Azure SQL Database](#) | [Microsoft Learn](#) 


- If you're upgrading to a higher service tier or compute size, the database max size doesn't increase unless you explicitly specify a larger size (maxsize).
- To downgrade a database, the database used space must be smaller than the maximum allowed size of the target service tier and compute size.
- When downgrading from Premium to the Standard tier, an extra storage cost applies if both (1) the max size of the database is supported in the target compute size, and (2) the max size exceeds the included storage amount of the target compute size. For example, if a P1 database with a max size of 500 GB is downsized to S3, then an extra storage cost applies since S3 supports a max size of 1 TB and its included storage amount is only 250 GB. So, the extra storage amount is 500 GB – 250 GB = 250 GB. For pricing of extra storage, see [Azure SQL Database pricing](#) . If the actual amount of space used is less than the included storage amount, then this extra cost can be avoided by reducing the database max size to the included amount.
- When upgrading a database with [geo-replication](#)  enabled, upgrade its secondary databases to the desired service tier and compute size before upgrading the primary database (general guidance for best

performance). When upgrading to a different edition, it's a requirement that the secondary database is upgraded first.

- When downgrading a database with [geo-replication](#)  enabled, downgrade its primary databases to the desired service tier and compute size before downgrading the secondary database (general guidance for best performance). When downgrading to a different edition, it's a requirement that the primary database is downgraded first.
- The restore service offerings are different for the various service tiers. If you're downgrading to the Basic tier, there's a lower backup retention period. See [Azure SQL Database Backups](#) .
- The new properties for the database aren't applied until the changes are complete.
- When data copying is required to scale a database (see [Latency](#) ) when changing the service tier, high resource utilization concurrent to the scaling operation may cause longer scaling times. With [Accelerated Database Recovery \(ADR\)](#) , rollback of long running transactions is not a significant source of delay, but high concurrent resource usage may leave less compute, storage, and network bandwidth resources for scaling, particularly for smaller compute sizes.

Additional considerations when changing service tier or rescaling compute size for elastic pool

[Scale elastic pool resources - Azure SQL Database | Microsoft Learn](#) 

- When downsizing vCores or eDTUs for an elastic pool, the pool used space must be smaller than the maximum allowed size of the target service tier and pool eDTUs.
- When rescaling eDTUs for an elastic pool, an extra storage cost applies if (1) the storage max size of the pool is supported by the target pool, and (2) the storage max size exceeds the included storage amount of the target pool. For example, if a 100 eDTU Standard pool with a max size of 100 GB is downsized to a 50 eDTU Standard pool, then an extra storage cost applies since target pool supports a max size of 100 GB and its included storage amount is only 50 GB. So, the extra storage amount is 100 GB – 50 GB = 50 GB. For pricing of extra storage, see [SQL Database pricing](#) . If the actual amount of space used is less than the included storage amount, then this extra cost can be avoided by reducing the database max size to the included amount.

How good have you found this content?

