Cross-instance point-in-time restore in Azure SQL Database Managed Instance

techcommunity.microsoft.com/t5/azure-sql/cross-instance-point-in-time-restore-in-azure-sql-database/ba-p/386208

March 24, 2019

By

[JovanPop](#)
Published Mar 23 2019 06:40 PM 6,203 Views

[Skip to footer content](#)

First published on MSDN on Jun 07, 2018
Azure SQL Managed Instance is a fully managed SQL Server instance hosted in Microsoft Azure cloud. Besides built-in management operations, Azure SQL Database Managed Instance enables you to perform some custom management actions such as restoring databases across instances from some point in time in the past using Azure Command line interface. In this post you will see how to perform point-in-time restore of a database from one instance to another using Azure CLI.

Azure SQL Database Managed Instance enables you to create a database as a copy of another database at some point in time in the past. This is known as **point-in-time restore** feature, and up till now you could perform point-in-time restore only within the same instance.

The latest release of Azure SQL Database Managed Instance enables you to perform point-in-time restore of a database from one instance to another. This might be useful if you need to be sure that you could easily restore a database to another instance if there is some issue on the original instance, or if you need a database for testing or auditing purposes on the test instance and you want to use copy of some of the existing database on another server.

Currently, point-in-time restore to another instance can be done only using [Azure CLI](#) version 2.0.34. In order to perform point-in-time restore to another instance you would need to take [the latest installation of Azure CLI (2.0.34 or above)](#). Once you install it, you can open Command prompt and use **az** command to manage your Azure resources.

The following Azure CLI commands enable you to take an automatically taken backups from a database on one instance and restore that backup as a database on another managed instance using [az sql midb restore](#) command:
```
az login
az account set -s "8cb143m6-7591-473d-90f7-b4bcdbd7e197"
az sql midb restore -g mygroup --mi myinstance --name mymanageddb --dest-mi
targetmi --dest-name targetmidb --time "2018-05-20T05:34:22"
```

Once you login to your Azure account using **az login** command and select Azure subscription where your Azure SQL Database Managed Instances are placed using az account set command, you can restore managed database using **az sql midb restore** command. The parameters that you need to provide are:

- -g representing Azure resource group where source and destination managed instances are placed (destination resource group can be changed)

- --mi representing the name of managed instance where source database is placed

- --name that represents a name of the database that you want to restore on the another instance

- --dest-mi representing the name of managed instance

- --dest-name that represents a name of the database that you want to restore on another instance

- --time that represents a point in time in the past of the database that you want to restore

Note that you could restore database to another instance within the same subscription and region. Cross-region restores are still not supported.

You can find additional parameters (such as different destination resource group) on az sql midb restore documentation page . As a result, Azure SQL Managed Instance will look at the automatically taken backups of the source managed instance, go through the history of full, differential, and log backups, and restore the database at the specified point in time.
1 Like
Like

4 Comments

MattMellifont
Regular Visitor

Jul 17 2019 10:28 PM

Hi @JovanPop,

Is cross-region restores on the roadmap for SQL MI?

1 Like
Like

EitanBlumin
Senior Member

Oct 05 2020 01:58 AM

Hello @JovanPop ,

Is this capability also possible in the Azure portal? Or only in the CLI?

0 Likes

Like

Thank you very much for the article. Just want to share when others will get hear searching for the same as me. The above needed a bit modification from my side to make it work in my env.

az sql midb restore --resource-group sourcerResourceGroup --managed-instance SourceMiName --name SourceDBName --dest-name DestDBName --time "2021-03-22T05:34:22" --dest-mi DestMiName --dest-resource-group DestRgroupName

0 Likes

Like

Thank you very much for the article. Just want to share when others will get here searching for the same as me. The above needed a bit modification from my side to make it work in my env.

az sql midb restore --resource-group sourceResourceGroup --managed-instance SourceMiName --name SourceDBName --dest-name DestDBName --time "2021-03-22T05:34:22" --dest-mi DestMiName --dest-resource-group DestResourcegroupName

0 Likes

Like

You must be a registered user to add a comment. If you've already registered, sign in. Otherwise, register and sign in.

Comment