

GeoDR - Create Geo-replication across subscriptions with private endpoints

Last updated by | Vitor Tomaz | Jun 8, 2022 at 5:33 AM PDT

Contents

- [Introduction](#)
- [Prerequisites](#)
- [Environment](#)
- [Limitations](#)
- [Geo-Replication Configuration](#)
- [Configuring private endpoints for both servers](#)
- [Virtual Network setup](#)
- [Disabling public access](#)
- [Next](#)
- [Troubleshooting](#)
- [References](#)
- [Disclaimer](#)

This is a copy of public blog article [Azure SQL Database - GEO Replication across subscription with private endpoints](#) 

Introduction


Active geo-replication is an Azure SQL Database feature that allows you to create readable secondary databases of individual databases on a server in the same or different data center (region).

We have received few cases where customers would like to have this setup across subscriptions with private endpoints. This article describes how to achieve it and set up Geo-replication between two Azure SQL servers across subscriptions using private endpoints while public access is disallowed.

Prerequisites

To start with this setup, kindly make sure the below are available in your environment:


- Two subscriptions for primary and secondary environments,
 1. Primary Environment: Azure SQL Server, Azure SQL database, and Virtual Network.
 2. Secondary Environment: Azure SQL Server, Azure SQL database, and Virtual Network.
Note: Use paired region for this setup, and you can have more information about paired regions by accessing this link.
- Public access should be enabled during the GEO replication configuration.

- Both your virtual network's subnet should not overlap IP addresses. You can refer to [this blog](#)  for more information.


Environment

For this article , the primary and secondary environments will be as below:

Primary Environment

Subscription ID: Primary-Subscription
Server Name: [primaryservertest.database.windows.net](#) 
Database Name: DBprim
Region: West Europe
Virtual Network: VnetPrimary
Subnet: PrimarySubnet - 10.0.0.0/24

Secondary Environment

Subscription ID: Secondary-Subscription
Server Name: [secservertest1.database.windows.net](#) 
Region: North Europe
Virtual Network: VnetSec
Subnet: SecondarySubnet - 10.2.0.0/24

Limitations

- Creating a geo-replica on a logical server in a different Azure tenant is not supported
- Cross-subscription geo-replication operations including setup and failover are only supported through Transact-SQL commands.
- Creating a geo-replica on a logical server in a different Azure tenant is not supported when Azure Active Directory only authentication for Azure SQL is active (enabled) on either primary or secondary logical server.

Geo-Replication Configuration

Follow the below steps to configure GEO replication (***make sure the public access is enabled while executing the below steps***)

1- Create a privileged login/user on both primary and secondary to be used for this setup:

a. Connect to your primary Azure SQL Server and create a login and a user on your master database using the below script:

```
--Primary Master Database
create login GeoReplicationUser with password = 'P@$word123'

create user GeoReplicationUser for login GeoReplicationUser
alter role dbmanager add member GeoReplicationUser
```

Get the created user SID and save it:

```
select sid from sys.sql_logins where name = 'GeoReplicationUser'
```

b. On the primary database create the required user as below:

```
-- primary user database
create user GeoReplicationUser for login GeoReplicationUser
alter role db_owner add member GeoReplicationUser
```

c. Connect to your secondary server and create the same login and user while using the same SID you got from point A:

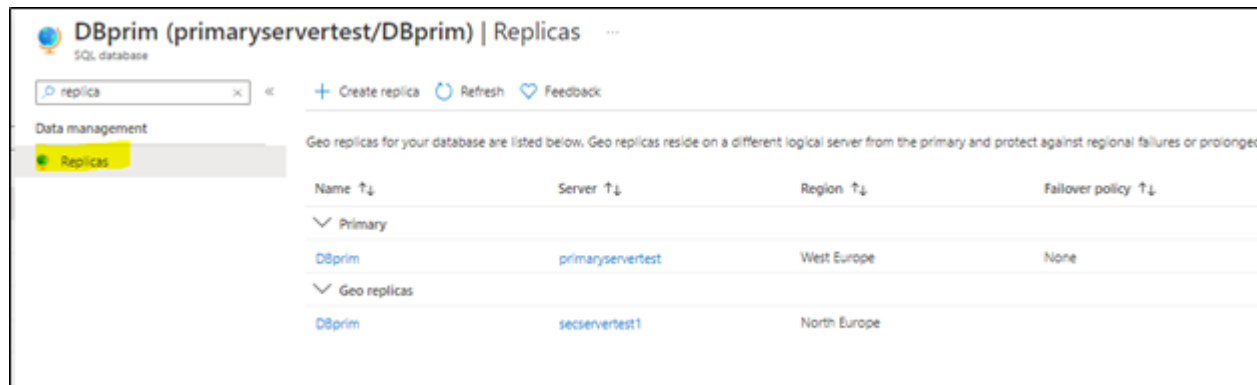
```
--Secondary Master Database
create login GeoReplicationUser with password = 'P@$word123', sid=0x01060000000000640000000000000001C98F52B9
create user GeoReplicationUser for login GeoReplicationUser;
alter role dbmanager add member GeoReplicationUser
```

2- Make sure that both primary and secondary Azure SQL servers firewall rules are configured to allow the connection (such as the IP address of the host running SQL Server Management Studio).

3- Log in with the created user to your primary Azure SQL server to add the secondary server and configure GEO replication, by running the below script on the primary master database:

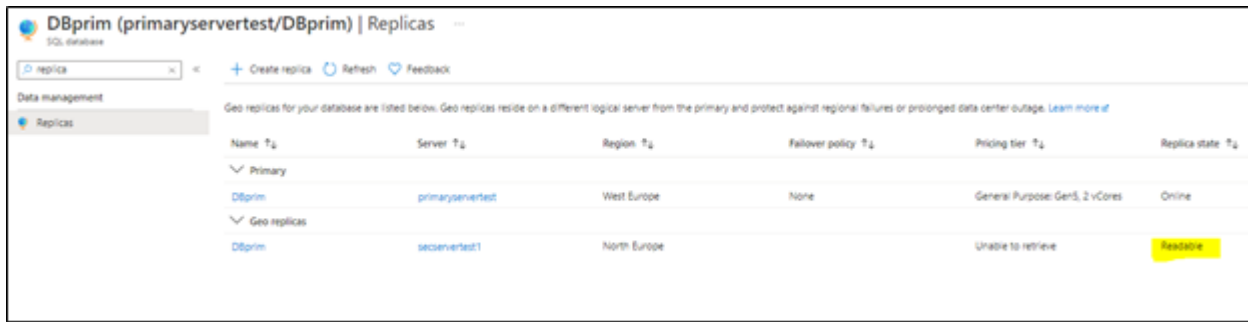
```
-- Primary Master database
alter database DBprim add secondary on server [secservertest1]
```

4- To verify the setup, access your Azure portal, go to your primary Azure SQL database, and access Replicas blade as below:



You will notice that the secondary database has been added and configured.

Note: Before moving to the next step make sure your replica has completed the seeding and is marked as "readable" under replica status (as highlighted below):

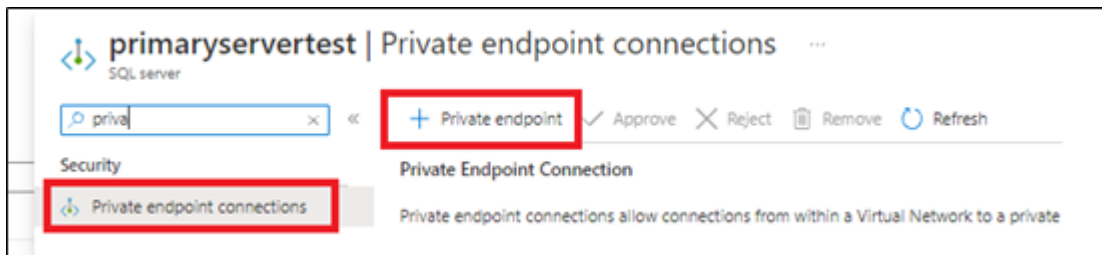


Name	Server	Region	Fallover policy	Pricing tier	Replica state
Primary					
DBprim	primaryservertest	West Europe	None	General Purpose: Gen5, 2 vCores	Online
Geo replicas					
DBprim	secondarytest1	North Europe		Unable to retrieve	Restoring

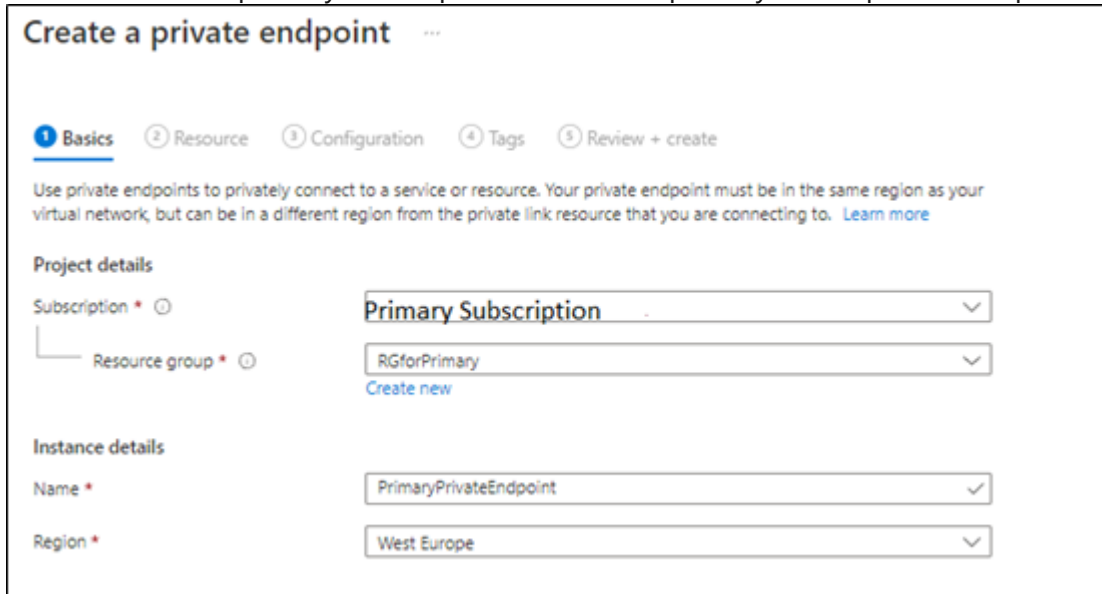
Configuring private endpoints for both servers

Now, we will start preparing the private endpoints setup for both primary and secondary servers.

1- From Azure Portal > Access Primary Server > private endpoints connections blade > add new private endpoints as below:



we will select the primary subscription to host the primary server private endpoints,



Create a private endpoint

1 Basics 2 Resource 3 Configuration 4 Tags 5 Review + create

Use private endpoints to privately connect to a service or resource. Your private endpoint must be in the same region as your virtual network, but can be in a different region from the private link resource that you are connecting to. [Learn more](#)

Project details

Subscription *

Resource group * [Create new](#)

Instance details


Name *


Region *


Create a private endpoint ...


✓ Basics **2 Resource** 3 Configuration 4 Tags 5 Review + create


Private Link offers options to create private endpoints for different Azure resources, like your private link service, a SQL server, or an Azure storage account. Select which resource you would like to connect to using this private endpoint. [Learn more](#)

Connection method  ☒ Connect to an Azure resource in my directory.
☐ Connect to an Azure resource by resource ID or alias.

Subscription *  Primary Subscription

Resource type *  Microsoft.Sql/servers

Resource *  primaryservertest


Target sub-resource *  sqlServer


Next, the primary private endpoint will be linked to the primary virtual network and make sure the private DNS zone is linked to the primary subscription as below:


Create a private endpoint ...

✓ Basics ✓ Resource **3 Configuration** 4 Tags 5 Review + create

Networking
To deploy the private endpoint, select a virtual network subnet. [Learn more](#)

Virtual network *  VnetPrimary

Subnet *  VnetPrimary/PrimarySubnet (10.0.0.0/24)

 If you have a network security group (NSG) enabled for the subnet above, it will be disabled for private endpoints on this subnet only. Other resources on the subnet will still have NSG enforcement.

Private DNS integration
To connect privately with your private endpoint, you need a DNS record. We recommend that you integrate your private endpoint with a private DNS zone. You can also utilize your own DNS servers or create DNS records using the host files on your virtual machines. [Learn more](#)

Integrate with private DNS zone ☒ Yes ☐ No

Configuration name	Subscription	Resource group	Private DNS zone
privatelink-database-windows-net	Primary Subscription	RGforPrimary	privatelink.database.windows.net

2- Create secondary server private endpoint, from Azure Portal > Access Secondary Server > private endpoints connections blade > add a new private endpoint as below:

in the below steps, we will select the secondary server virtual network and subscription,

Create a private endpoint

1 Basics 2 Resource 3 Configuration 4 Tags 5 Review + create

Use private endpoints to privately connect to a service or resource. Your private endpoint must be in the same region as your virtual network, but can be in a different region from the private link resource that you are connecting to. [Learn more](#)

Project details

Subscription *

Resource group *
[Create new](#)

Instance details

Name *

Region *

Create a private endpoint

✓ Basics 2 Resource 3 Configuration 4 Tags 5 Review + create

Private Link offers options to create private endpoints for different Azure resources, like your private link service, a SQL server, or an Azure storage account. Select which resource you would like to connect to using this private endpoint. [Learn more](#)

Connection method ☒ Connect to an Azure resource in my directory.
☐ Connect to an Azure resource by resource ID or alias.

Subscription *

Resource type *

Resource *

Target sub-resource *

In the next step, will link the secondary server private endpoint with the primary private DNS Zone, as both primary and secondary private endpoints should be linked to the same private DNS zone (as below),

Create a private endpoint

✓ Basics ✓ Resource 3 Configuration 4 Tags 5 Review + create

Networking

To deploy the private endpoint, select a virtual network subnet. [Learn more](#)

Virtual network *

Subnet *
ⓘ If you have a network security group (NSG) enabled for the subnet above, it will be disabled for private endpoints on this subnet only. Other resources on the subnet will still have NSG enforcement.

Private DNS integration

To connect privately with your private endpoint, you need a DNS record. We recommend that you integrate your private endpoint with a private DNS zone. You can also utilize your own DNS servers or create DNS records using the host files on your virtual machines. [Learn more](#)

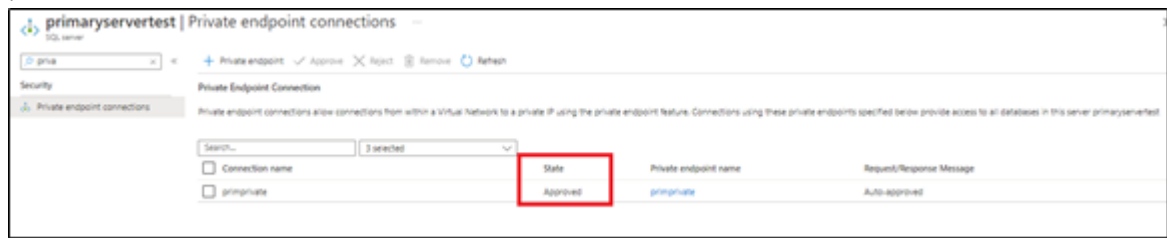
Integrate with private DNS zone ☒ Yes ☐ No

Note: we will select the secondary Vnet

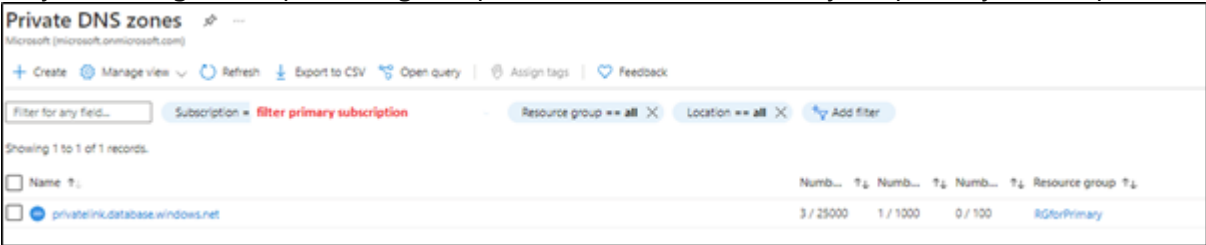
Note: for the private DNS integration we need to make sure we select the primary subscription to link them both

Configuration name	Subscription	Resource group	Private DNS zone
privatelink-database-windows-net	<input type="text" value="Primary Subscription"/>	<input type="text" value="RGforPrimary"/>	privatelink.database.windows.net

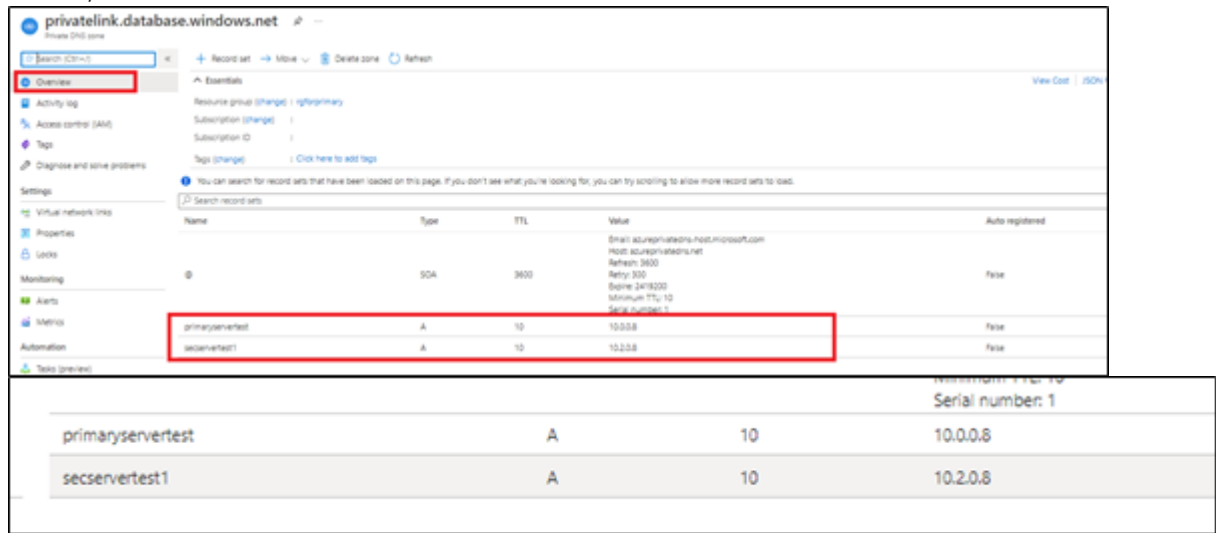
3- Once both private endpoints are created, make sure that they are accepted as mentioned in [this document](#) ☑



4- Access your private DNS zone from Azure portal, and verify that both are linked to the same one. This can be checked by accessing Azure portal > go to private DNS zone > select your primary subscription and check it as



below,



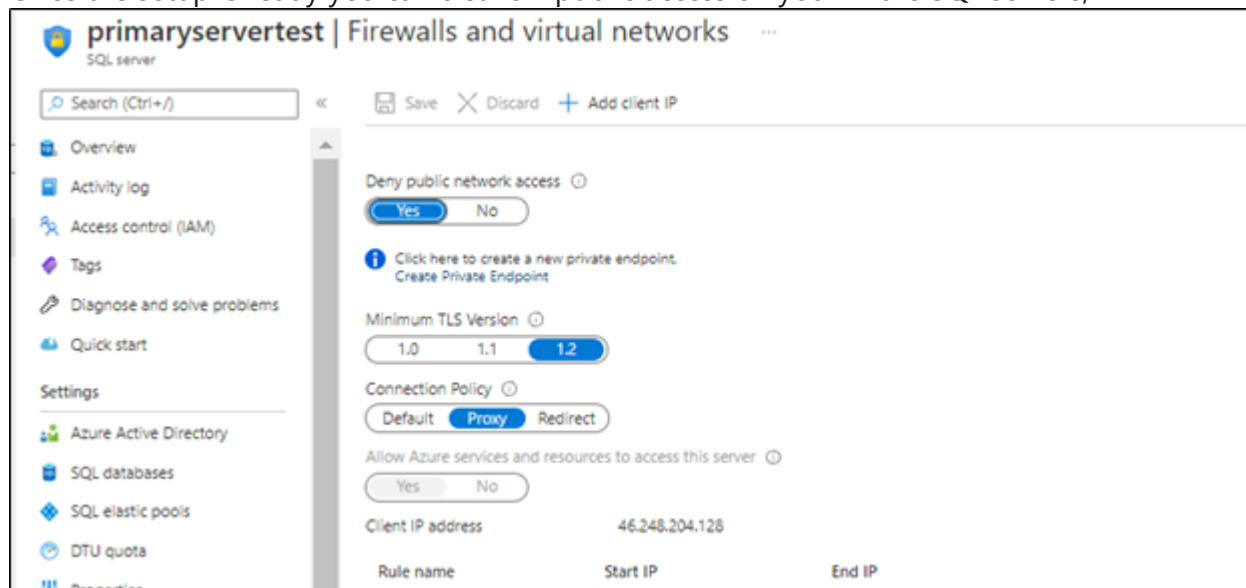
Note: this step has been discussed in detail in [this blog article](#) ☑.

Virtual Network setup

You need to make sure your Azure Virtual networks have Vnet peering between primary and secondary, in order to allow communication once the public access is disabled. For more information, you can access [this document](#) ☑.

Disabling public access

Once the setup is ready you can disallow public access on your Azure SQL servers,



Next

Once the public access is disabled, the Geo-replication will be running under private endpoints between your Azure SQL server across subscriptions.

You can initiate a failover using the below Transact-SQL command:

```
ALTER DATABASE [DatabaseName] FAILOVER
```

Troubleshooting

1- You may encounter below error when adding the secondary using T-SQL

```
alter database DBprim add secondary on server [secservertest1]

Msg 42019, Level 16, State 1, Line 1
ALTER DATABASE SECONDARY/FAILOVER operation failed. Operation timed out.
```

Possible solution:

Set "deny public access" to off while setting up the geo replication via the T-SQL commands. Once the geo-replication is set up, "deny public access" can be turned back on and the secondary will be able to sync and get the data from primary.

Public access only needs to be on for setting up the geo replication.

2- Also, You may encounter below error when adding the secondary using T-SQL

```
alter database DBprim add secondary on server [secservertest1]

Msg 40647, Level 16, State 1, Line 1
Subscription 'xxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxx' does not have the server 'secservertest1'.
```


Possible solution:

Make sure that both private links use the same private DNS zone that was used for the primary. Refer to [blog](#) for more information.

3- Another error, can be due to insufficient permission as follows:

```
Msg 45137, Level 16, State 1, Line 1
Insufficient permission to create a database copy on server 'secservertest1'.
```

Possible solution:

Make sure the user permission as mentioned in [Active geo-replication - Azure SQL Database | Microsoft Docs](#)

4- Another workaround for many issue is to create geo-secondary through ARM template:

[Microsoft.Sql/servers/databases - Bicep & ARM template reference | Microsoft Docs](#)

For example: [How to deploy a sql database with geo replication using azure resource manager templates - Stack Overflow](#)

Sample Template:

```
{
  "$schema": "https://schema.management.azure.com/schemas/2019-04-01/deploymentTemplate.json#",
  "contentVersion": "1.0.0.0",
  "parameters": {
    "servers_secservertest1_name": {
      "defaultValue": "secservertest1",
      "type": "String"
    }
  },
  "variables": {},
  "resources": [
    {
      "type": "Microsoft.Sql/servers/databases",
      "apiVersion": "2021-02-01-preview",
      "name": "[concat(parameters('servers_secservertest1_name'), '/ssdb')]",
      "location": "WestEurope",
      "sku": {
        "name": "Standard",
        "tier": "Standard",
        "capacity": 20
      },
      "kind": "v12.0,user",
      "properties": {
        "collation": "Latin1_General_CI_AS",
        "maxSizeBytes": "268435456000",
        "createMode": "Secondary",
        "catalogCollation": "SQL_Latin1_General_CP1_CI_AS",
        "zoneRedundant": false,
        "readScale": "Disabled",
        "requestedBackupStorageRedundancy": "Geo",
        "maintenanceConfigurationId": "/subscriptions/[redacted]/providers/Microsoft.Maintenance/publicMaintenanceConfigurations/SQL_Default",
        "sourceDatabaseId": "/subscriptions/[redacted]/resourceGroups/[redacted]/providers/Microsoft.Sql/servers/sorucedbserver/databases/ssdb",
        "isLedgerOn": false
      }
    }
  ]
}
```

References

[Active geo-replication - Azure SQL Database | Microsoft Docs](#)

[Using Failover Groups with Private Link for Azure SQL Database - Microsoft Tech Community](#)

Disclaimer

Please note that products and options presented in this article are subject to change. This article reflects the Geo Replication across different subscriptions with private endpoints option available for Azure SQL Database in October, 2021.

How good have you found this content?

