

Azure SQL Database Failover Groups (Auto-DR)

Last updated by | Vitor Tomaz | Aug 5, 2020 at 12:41 PM PDT

Contents

- [Internal Doc Reference :](#)
- [Classification](#)

Microsoft Confidential before the Public Preview Release

Azure SQL Database Failover Groups (fka Auto-DR) is a SQL Database feature designed to automatically manage replication, connectivity and failover of a set of databases. With it the customers gain the ability to automatically recover the databases after catastrophic regional failures or other unplanned events that result in full or partial loss of the SQL DB service's replication in a region. Additionally, they can use the redundant capacity to load balance secondary read-only workloads. Failover Group provides the user experience similar to AlwaysON Availability Groups.

Public preview brings to our customers the following new capabilities:

- Automatic geo-replication of a group of databases or pools within a logical server
- Ability to configure automatic failover of the entire group
- A connection endpoint to the primary databases that doesn't change after failover
- A connection end-point to the secondary databases that doesn't change after failover (for load balancing the read-only workloads)
- Ability to configure multiple failover groups per logical server

If you have customers using geo-replication or using the private preview of Auto-DR, please refer them to this [blog](#). If you work with the first party customers, using failover groups is part of the official [HA guidance document](#). Please note the Portal support will be enabled on May 24 2017.

1.1 Terms and definitions

Failover Group (FG). A set of geo-replicated databases that fail over together.

Primary region: the region hosting the primary databases in the FG.

Secondary region: one of the regions that host the secondary databases in the FG and can be used as a failover location during recovery.

Primary database. The read-write copy of a database in an FG.

Secondary database. A read-only copy of a database in an FG.

Primary server. A server that hosts the primary databases in FG.

Secondary server. A server that hosts the secondary databases in FG.

Failover Group read-write listener. A DNS CNAME record that points to the current primary server URL. It allows the read-write SQL applications to transparently connect the primary database even when the primary changes after failover.

Failover Group read-only listener. A DNS name that is resolved to one of the existing secondary servers URL. It allows the read-only SQL applications to transparently connect the secondary database.

DR drill – a user initiated simulation of regional failure for the purposes of testing the application's readiness and compliance.

Failover - the process of switching all secondary databases in the Failover Group to the primary role.

Failback – the process of switching all the primary databases in the FG the primary back to the primary region after the outage is mitigated and switching the current primary in the DR region to the secondary role.

Grace period without data loss – a configurable time interval since the outage start time that triggers the friendly failover of the databases in the failover group. If the outage is self-mitigated during this interval no failover will be triggered. If succeeds the failover switches to the secondary databases without data loss. If the friendly failover fails the forced failover will be triggered by the second grace period (grace period with data loss).

Grace period with data loss – a configurable time interval since the outage is detected (impact start time in ICM) that triggers the forced failover of the databases in the failover group. If the outage is self-mitigated during this interval no failover will be triggered. If this grace period is set to the same value as the grace period without data loss only the forced failover will be triggered.

Automatic failover – the service initiated failover based on the configured grace periods.

Manual failover – the user or application initiated failover can applied to all or some databases in the group. Manual failover can be initiated even if Automatic policy is set up.

Value proposition

- Failover Group reduce the management complexity and cost associated with the failover of applications that include multiple database such as SaaS applications using Malmö pools. It provides the following incremental value: ·
- Flexible automatic failover policy. This reflects customer feedback that it is critical for many application but needs to accommodate different application SLAs and different resilience to data loss. ·
- Ability to failover without changing SQL connection strings. This reflects customer feedback that it is very difficult to manage dynamically with the current geo-replication. ·
- Easy to provision the large scale regional stamps by declarative configuration. This is critical for the SaaS vendors as it avoids complex deployment workflows for large sets of individual databases. ·
- Enables active-active SaaS deployment pattern by supporting the ability to cross replicate the databases between two elastic pools containing a mix of primary and secondary databases. The application can failover and failback a subset of databases in each pool, which reduces the scale of customer impact. ·
- Enables read-scale solution by supporting a read-only listener that routes connections to the secondary server. This enables using the secondary databases for load balancing of read-only workloads

PowerShell reference

Create Failover Group

This command is executed in the context of the primary server.

Cmdlet New-AzureRmSqlDatabaseFailoverGroup

Parameters

[-ResourceGroupName <String>]

[-FailoverGroupName <String>]

[-ServerName <String>]

[-PartnerResourceGroupName <String>]

[-PartnerServerName <String>]

[-FailoverPolicy <String>]

[-GracePeriodWithoutDataLossMinutes <String>]

[-GracePeriodWithDataLossMinutes <String>]

[-AllowReadOnlyFailoverToPrimary <String>]

[-CopySecurityConfiguration]

[-Force] [-Confirm] [-WhatIf] [<CommonParameters>]

Parameters

PrimaryResourceGroupName <String>

Name of the resource group hosting the primary server of FG. It is not required, if Server object is piped to the cmdlet.

ServerName <String>

Name of the primary server in FG. It is not required, if Server object is piped to the cmdlet.

FailoverGroupName <String>

Name of the Failover Group. The name must be unique.

PartnerResourceGroupName <String>

Name of the resource group hosting the secondary server. If not specified, the resource group of the source server will be used.

PartnerServerName <String>

Name of the secondary server in FG. It is required.

FailoverPolicy <String>

It can be set to "Automatic" or "Manual". It is "Manual" if not specified. If "Automatic" is selected but none of the grace periods is defined the cmdlet will fail.

GracePeriodWithoutDataLossMinutes <String>

A time interval since the outage is detected (impact start time) that triggers the friendly failover of the databases in the group (with full data synchronization). If the outage is self-mitigated during this interval no failover will be triggered. If the friendly failover fails the forced failover will be triggered by the GracePeriodMinutesWithDataLoss.

GracePeriodWithDataLossMinutes <String>

A time interval since the outage is detected (impact start time) that triggers the forced failover of the databases in the group (without data synchronization). If the outage is self-mitigated during this interval no failover will be triggered. If both grace periods are set to the same value only the forced failover will be triggered.

AllowReadOnlyFailoverToPrimary <String>

Specifies if read-only connection can be routed to the primary database in case all read-only replicas failed. It can be set to "Enabled" or "Disabled". By default it is "Disabled" to prevent overloading the primary by the read-only workload.

CopySecurityConfiguration

Specifies that the security configuration be copied from the primary server to the secondary. If specified audit settings, logins, credentials etc. will be copied. By default the security configuration is not copied.

Semantics

This command will create a new FG and register it on both primary and secondary servers. It will also create two listener endpoints with the URL <FailoverGroupName>.database.windows.net and <FailoverGroupName>.secondary.database.windows.net respectively. All read-write TDS client connections are automatically routed to the server that is the primary at the moment. All read-only TDS client connections are routed to the secondary server. If the primary server is affected by an outage the automatic failover will be triggered if specified. The grace period parameters will control the RTO. If read-only endpoint failover is enabled it will be triggered as soon as the outage of the secondary server is detected. If automatic geo-replication is enabled each new member of the group will automatically be synchronized to all secondary servers.

Inputs

The intended input is the server object for the primary server.

Outputs

The command will return the server object.

· Microsoft.WindowsAzure.Commands.SqlDatabase.Model.FailoverGroupContext

Authentication

In order to add a server, users should have write access to the resource group according to RBAC. See Role-based access control for details.

Example

The following examples creates a FG "myFG" for server "myserver" as primary and server "mydrserver" as secondary. Both servers are members of the resource group "myrg" with 20 and 120 minutes of lossless and lossy grace periods respectively. C:> \$ag = New-

```
AzureRMSqlDatabaseFailoverGroup -ResourceGroupName "myrg" -ServerName "myserver" -
SqlAdministratorCredentials $credential -PartnerServerName "mydrserver" - FailoverGroupName "MyFG" -
FailoverPolicy "Automatic" -GracePeriodWithoutDataLossMinutes 20 -GracePeriodWithDataLossMinutes 120
```

Remove Failover Group

Cmdlet Remove-AzureRmSqlDatabaseFailoverGroup

Azure SQL Database Failover Groups

Parameters

[-ResourceGroupName <String>]

[-FailoverGroupName <String>]

[-ServerName <String>]

[-Force] [-Confirm] [-WhatIf] [<CommonParameters>]

Parameters

ResourceGroupName <String>

Name of the resource group hosting the primary server of the FG. It is not required, if Server object is piped to the cmdlet.

ServerName <String>

Name of the primary server of FG. It is not required, if Server object is piped to the cmdlet.

FailoverGroupName <String>

Specifies the listener name to reference the FG that needs to be removed. It is a required parameter.

Semantics

This command deletes all secondary databases included the FG and removes the FG object with the specified name from each participating server. The primary databases will become regular read-write databases. The listener endpoint will be unregistered from DNS.

Inputs

The intended input is the server object.

Outputs

None

Authentication

In order to add a server, users should have write access to the resource group according to RBAC. See Role-based access control for details.

Example

The following examples delete the FG "myFG" from server "myserver".

```
PS C:\> Remove-AzureRmSqlDatabaseFailoverGroup -PrimaryResourceGroupName "myFG" ServerName "myserver" -FailoverGroupName "myFG"
```

Retrieve Failover Group

Cmdlet Get-AzureRmSqlDatabaseFailoverGroup

Parameters

[-ResourceGroupName <String>]

[-ServerName <String>]

[-FailoverGroupName <String>]

[-Force] [-Confirm] [-WhatIf] [<CommonParameters>]

Parameters

ResourceGroupName <String>

Name of the resource group hosting the primary server of FG. It is not required, if Server object is piped to the cmdlet.

ServerName <String>

Name of the server that is currently the primary. It is not required, if Server object is piped to the cmdlet.

FailoverGroupName <String>

Specifies the listener name to reference the FG that needs to be retrieved. If not specified a list of all FGs is returned.

Semantics

This command returns the FG for the defined on the server.

Inputs

The intended input is the server object.

Outputs

The command will return the object of type

- Microsoft.WindowsAzure.Commands.SqlDatabase.Model.FailoverGroupContext

Authentication

In order to retrieve the server properties, users should have read access to the resource group according to RBAC. See Role-based access control for details.

Example

The following example retrieves all FGs created on server "myserver" in resource group "myrg". C:> \$myFG = Get-AzureRmSqlDatabaseFailoverGroup -ResourceGroupName "myrg" -ServerName "myserver"

that controls failover

Modify Failover Group

This command is executed on the primary server.

Cmdlet Set-AzureRmSqlDatabaseFailoverGroup

Parameters

[-ResourceGroupName <String>]

[-FailoverGroupName <String>]

[-ServerName <String>]

[-PartnerResourceGroupName <String>]

[-PartnerServerName <String>]

[-FailoverPolicy <String>]

[-GracePeriodWithoutDataLossMinutes <String>]

[-GracePeriodWithDataLossMinutes <String>]

[-AllowReadOnlyFailoverToPrimary <String>]

[-Force] [-Confirm] [-WhatIf] [<CommonParameters>]

Parameters

PrimaryResourceGroupName <String>

Name of the resource group hosting the primary server of FG. It is not required, if Server object is piped to the cmdlet.

ServerName <String>

Name of the primary server in FG. It is not required, if Server object is piped to the cmdlet.

FailoverGroupName <String>

Name of the Failover Group. The name must be unique.

PartnerResourceGroupName <String>

Name of the resource group hosting the secondary server. If not specified, the resource group of the source server will be used.

PartnerServerName <String>

Name of the secondary server in FG. It is required.

FailoverPolicy <String>

It can be set to "Automatic" or "Manual". It is "Manual" if not specified. If "Automatic" is selected but none of the grace periods is defined the cmdlet will fail.

GracePeriodWithoutDataLossMinutes <String>

A time interval since the outage is detected (impact start time) that triggers the friendly failover of the databases in the group (with full data synchronization). If the outage is self-mitigated during this interval no failover will be triggered. If the friendly failover fails the forced failover will be triggered by the GracePeriodMinutesWithDataLoss.

Commented [MW2]: This is a Boolean parameter

GracePeriodWithDataLossMinutes <String>

A time interval since the outage is detected (impact start time) that triggers the forced failover of the databases in the group (without data synchronization). If the outage is self-mitigated during this interval no failover will be triggered. If both grace periods are set to the same value only the forced failover will be triggered.

AllowReadOnlyFailoverToPrimary <Bool>

Specifies if read-only connection can be routed to the primary database in case all read-only replicas failed. By default the value is 0 (not allowed) to prevent overloading the primary by the read-only workload.

Semantics

This command modifies the configuration of the failover group. Adding or removing servers and databases requires using the specialized cmdlets.

Inputs

The intended input is the server object for the primary server.

Outputs

The command will return the server object.

· Microsoft.WindowsAzure.Commands.SqlDatabase.Model.FailoverGroupContext

Authentication

In order to add a server, users should have write access to the resource group according to RBAC. See Role-based access control for details.

Example

The following examples changes the failover policy for FG "myFG" to "UserControlled" and sets the read-only failover setting to true (allow). C:> \$server = Get-AzureRMSqlDatabaseServer -ResourceGroupName "myrg" - ServerName "myserver" C:> \$server | Set-AzureRMSqlDatabaseFailoverGroup -ResourceGroupName "myrg" - SqlAdministratorCredentials \$credential -FailoverGroupName "MyFG" -FailoverPolicy "Manual" - AllowReadOnlyFailoverToPrimary \$true

Trigger failover

This command is executed on the secondary server.

Cmdlet Switch-AzureRMSqlDatabaseFailoverGroup

Parameters

[-ResourceGroupName <String>]

[-ServerName <String>]

[-FailoverGroupName <String>]

[-AllowDataLoss]

[-Force] [-Confirm] [-WhatIf] [<CommonParameters>]

Parameters

ResourceGroupName <String>

Name of the resource group hosting the secondary server in FG. It is not required, if Server object is piped to the cmdlet.

ServerName <String>

Name of the secondary server in FG. It is not required, if Server object is piped to the cmdlet.

FailoverGroupName <String>

Specifies a listener name to reference the FG. It is a required parameter.

AllowDataLoss

Allows the cmdlet to promote secondary to primary without syncing replication first. This will enable failover when current primary is unavailable. For planned failover this parameter should not be set.

Semantics

This command must be executed on a secondary server of a specific FG. The FG is identified by the listener name. The command switches all secondary databases to the primary role. All active TDS sessions will be disconnected. All new TDS sessions will be automatically re-routed to the secondary server, which now becomes primary server. When the original primary server is back online it will automatically become the secondary server and all formerly primary databases in it will switch to the secondary role.

Inputs

The intended input is server object.

Outputs

The output type is the Server object representing the new primary server.

· Microsoft.WindowsAzure.Commands.SqlDatabase.Model.FailoverGroupContext

Authentication

In order to failover, users should have write access to the resource group according to RBAC. See Role-based access control for details.

Example

The following command fails over the primary server "myserver" to its replica role of the replica and re-routes the server URL to it. After failover it checks the new location. C:> \$ag = Get-

```
AzureRmSqlDatabaseFailoverGroup -ResourceGroupName "myrg" -ServerName "myserver" -
FailoverGroupName "myFG" C:> $ag | Switch-AzureRmSqlDatabaseFailoverGroup -AllowDataLoss
```

Add database to replication group

This command is executed on the primary server.

```
Cmdlet Add-AzureRmSqlDatabaseToFailoverGroup
```

Parameters

[-ResourceGroupName <String>]

[-FailoverGroupName <String>]

[-ServerName <String>]

[-Database <String[]>]

[-All]

[-ElasticPool <String[]>]

[-Force] [-Confirm] [-WhatIf] [<CommonParameters>]

Parameters

ResourceGroupName <String>

Name of the resource group hosting the secondary server in FG. It is not required, if Server object is piped to the cmdlet.

FailoverGroupName <String>

Specifies a listener name to reference the FG. It is a required parameter.

ServerName <String>

Name of the secondary server in FG. It is not required, if Server object is piped to the cmdlet.

Database <String[]>

Specifies an array of read/write user databases on the primary server. These databases will be added to the FG immediately and geo-replicated to all secondary servers in the FG. The databases cannot belong to another

Failover Group. This parameter is optional if -ElasticPool or -All is specified.

ElasticPool <String[]>

Specifies an array of elastic pools on the primary server. All read/write databases in the pool(s) will be added to the FG immediately and geo-replicated to all secondary servers in the FG. These databases cannot belong to another Failover Group. This parameter is optional if -Database is specified. NOTE: (1) the same pool can belong to multiple FGs. (2) All partner servers must contain pool(s) with the same name if -ElasticPool is specified. (3) If a database happens to have a geo-secondary on a server included in the specified FG the command will recognize it and add both primary and secondary database(s) to it.

Semantics

This command must be executed on the primary server. The command adds the specified databases to the referenced FG. The command will also automatically start geo-replication of the specified database(s).

Inputs

The intended input is server object.

Outputs

The output type is the Server object representing the new primary server.

· Microsoft.WindowsAzure.Commands.SqlDatabase.Model.FailoverGroupContext

Authentication

In order to failover, users should have write access to the resource group according to RBAC. See Role-based access control for details.

Example

The following command adds all databases in elastic pool "mypool" on primary server "myserver" to FG "myFG". The geo-secondaries will be automatically created on all secondary servers in the pools with the same name.

```
C:\> $server = Get-AzureRmSqlDatabaseServer -ResourceGroupName "myrg" -ServerName "myserver" C:\> $server | Add-AzureRmSqlDatabaseToFailoverGroup -FailoverGroupName "myFG" ElasticPool "mypool"
```

Remove Server from Failover Group

This command is executed on the primary server.

Cmdlet Remove-AzureRmSqlserverFromFailoverGroup

Parameters

[-ResourceGroupName <String>]

[-FailoverGroupName <String>]

[-ServerName <String>]

[-PartnerResourceGroupName <String>]

[-PartnerServerName <String>]

[-Force] [-Confirm] [-WhatIf] [<CommonParameters>]

Parameters

PrimaryResourceGroupName <String>

Name of the resource group hosting the primary server of FG. It is not required, if Server object is piped to the cmdlet.

ServerName <String>

Name of the primary server in FG. It is not required, if Server object is piped to the cmdlet.

FailoverGroupName <String>

Name of the Failover Group.

PartnerResourceGroupName <String>

Name of the resource group hosting the secondary server to be removed from the FG. If not specified, the resource group of the source server will be used.

PartnerServerName <String>

Name of the secondary server to be removed from the FG. It is required.

Semantics

This command will unregister the specified secondary server from the specified FG. All secondary read-only databases on that server will be dropped.

NOTE: if all or some of the secondary databases are in an elastic pool on the specified server they will be dropped but the pool's configuration will not be impacted.

Inputs

The intended input is the server object for the primary server.

Outputs

The command will return the server object.

· Microsoft.WindowsAzure.Commands.SqlDatabase.Model.FailoverGroupContext

Authentication

In order to remove a server, users should have write access to the resource group according to RBAC. See Rolebased access control for details.

Example

The following examples removes server "myserver2" from the FG "myFG". Both servers are members of the resource group "myrg".

```
C:> $server = Get-AzureRMSqlDatabaseServer -ResourceGroupName "myrg" -
ServerName "myserver" C:> $ag = $server | Remove-AzureRMSqlServerFromFailoverGroup
SqlAdministratorCredentials $credential -FailoverGroupName "MyFG" -PartnerServer "myserver2"
```

Remove database from Failover Group

This command is executed on the primary server.

Azure SQL Database Failover Groups

Cmdlet Remove-AzureRmSqlDatabaseFromFailoverGroup

Parameters

[-ResourceGroupName <String>]

[-FailoverGroupName <String>]

[-ServerName <String>]

[-Database <String[]>]

[-ElasticPool <String[]>]

[-Force] [-Confirm] [-WhatIf] [<CommonParameters>]

Parameters

ResourceGroupName <String>

Name of the resource group hosting the secondary server in FG. It is not required, if Server object is piped to the cmdlet.

FailoverGroupName <String>

Specifies a listener name to reference the FG. It is a required parameter.

ServerName <String>

Name of the secondary server in FG. It is not required, if Server object is piped to the cmdlet.

Database <String[]>

Specifies an array of read/write user databases on the primary server to be removed from the FG. This parameter is optional if -ElasticPool is specified.

NOTE: the string "*" can be used to specify all. In that case all secondary databases on the in the FG will be removed from it.

ElasticPool <String[]>

Specifies an array of elastic pools on the primary server. All secondaries of the read/write databases in the pool(s) that are members of the specified FG will be dropped immediately. This parameter is optional if -Database is specified. NOTE: the pool(s) configurations on the primary and secondary servers will not be impacted.

Semantics

This command must be executed on the primary server. The command drops the corresponding secondary databases on all secondary servers and unregisters the read-write databases from the FG.

Inputs

The intended input is the server object for the primary server.

Outputs

The command will return the FG object.

· Microsoft.WindowsAzure.Commands.SqlDatabase.Model.FailoverGroupContext

Authentication

In order to remove databases, users should have write access to the resource group according to RBAC. See Rolebased access control for details.

Example

The following command removes all databases from to FG "myFG". The geo-secondaries will be immediately dropped.

```
C:\> $server = Get-AzureRMSqlDatabaseServer -ResourceGroupName "myrg" -ServerName "myserver"
C:\> $ag = $server | Remove-AzureRMSqlDatabaseFromFailoverGroup -FailoverGroupName "myFG" -Database "*"

```

Error codes

6.1 Paired servers cannot reside in the same region
 ErrorNumber 40907 ErrorSeverity EX_USER ErrorFormat Servers involved in a FG cannot reside in the same region
 ErrorCause Servers specified in a FG need to reside in different regions to provide isolation

ErrorCode const int GATEWAY_SERVER_FAILOVER_CONFIGURATION_INVALID_REGION = 7

6.2 FG already exists for the server pair
 ErrorNumber 40908 ErrorSeverity EX_USER ErrorFormat The FG with this name already exists on server '%.*ls'

ErrorCause There is an existing FG for the two servers

ErrorCode const int GATEWAY_SERVER_FAILOVER_CONFIGURATION_ALREADY_EXISTS = 8;

6.3 FG does not exist for the server pair
 ErrorNumber 40909 ErrorSeverity EX_USER ErrorFormat A FG does not exist for servers '%.*ls' and '%.*ls'

ErrorCause There is no FG for the server pair

ErrorCode const int GATEWAY_SERVER_FAILOVER_CONFIGURATION_DOES_NOT_EXIST = 9;

6.4 Failover cannot be initiated from a primary server
 ErrorNumber 40911 ErrorSeverity EX_USER ErrorFormat Server '%.*ls' is not the secondary in the FG and cannot initiate a failover

ErrorCause Failover cannot be initiated from the primary server in a FG

ErrorCode const int GATEWAY_DISASTER_RECOVERY_CONFIGURATION_SERVER_NOT_SECONDARY = 11;

Internal Doc Reference :

<[https://microsoft.sharepoint.com/teams/bidpwiki/Pages1/Azure%20SQL%20Database%20Failover%20Groups%20\(Auto-DR\).aspx](https://microsoft.sharepoint.com/teams/bidpwiki/Pages1/Azure%20SQL%20Database%20Failover%20Groups%20(Auto-DR).aspx)>

Classification

Root Cause: Azure SQL DB v2\GeoDR/AutoDR

How good have you found this content?

