

DR Solution for Azure SQL PaaS DB with Geo Replication

Version 8.2.0

DISCLAIMER

IBM believes that the information in this publication is accurate as of its publication date. The information is subject to change without notice.

COPYRIGHT

©IBM Corporation 2003, 2021. All Rights Reserved.

Printed June 2021.

Use, copy, and distribution of any IBM software described in this publication needs an applicable software license.

No part of this product or document may be reproduced, stored in a retrieval system, or transmitted, in any form by any means, electronic, mechanical, photocopy, recording, or otherwise, without prior written authorization of IBM Corporation and its licensers, if any.

Revision History

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Document Version | Revision Date | Sections Updated | Pages Updated | Supported Product Version |
| 8.2.0 | June 2021 | All | All | 8.2.x |

Contents

[Introduction 7](#_Toc75946571)

[Scope 7](#_Toc75946572)

[References 7](#_Toc75946573)

[Abbreviations 7](#_Toc75946574)

[Technology Overview 7](#_Toc75946575)

[Azure SQL 7](#_Toc75946576)

[Azure SQL Database 8](#_Toc75946577)

[Active Geo-Replication 9](#_Toc75946578)

[Active geo-replication terminology and capabilities 10](#_Toc75946579)

[Automatic Asynchronous Replication 10](#_Toc75946580)

[Readable Secondary Databases 10](#_Toc75946581)

[Planned failover 10](#_Toc75946582)

[Unplanned failover 10](#_Toc75946583)

[Multiple readable secondaries 10](#_Toc75946584)

[Geo-replication of databases in an elastic pool 11](#_Toc75946585)

[User-controlled failover and failback 11](#_Toc75946586)

[Solution Design 11](#_Toc75946587)

[Architecture Diagram 12](#_Toc75946588)

[IBM Resiliency Orchestration 13](#_Toc75946589)

[IBM Resiliency Orchestration (HA) 13](#_Toc75946590)

[Site Controller 13](#_Toc75946591)

[Site Controller mapping 14](#_Toc75946592)

[Site Controller Configuration for multiple Azure SQL Paas DB 14](#_Toc75946593)

[IBM RO Deployment Architecture at Azure Premises 14](#_Toc75946594)

[Site Controller Deployment 14](#_Toc75946595)

[IBM RO Master server deployment 15](#_Toc75946596)

[Supported Platforms and versions 15](#_Toc75946597)

[Pre-requisites 15](#_Toc75946598)

[Security Limitations 16](#_Toc75946599)

[Known Limitations 16](#_Toc75946600)

[Limitations 16](#_Toc75946601)

[Auto Discovery 17](#_Toc75946602)

[Management service 17](#_Toc75946603)

[Adding Azure Management Service 17](#_Toc75946604)

[Configure Azure SQL PaaS DB Management Service 18](#_Toc75946605)

[Multiple Management Services used for monitoring 19](#_Toc75946606)

[Group Creation using Discovery Simplification Service (DSS) Tool 19](#_Toc75946607)

[Recovery Group 21](#_Toc75946608)

[Subsystem Discovery 21](#_Toc75946609)

[Components 22](#_Toc75946610)

[Monitoring 22](#_Toc75946611)

[RPO/RTO Monitoring 22](#_Toc75946612)

[App RPO 23](#_Toc75946613)

[Data RPO 23](#_Toc75946614)

[Recovery Time Objective 23](#_Toc75946615)

[Replication Monitoring 23](#_Toc75946616)

[Replication Status 23](#_Toc75946617)

[Data lag 24](#_Toc75946618)

[Stop Replication 24](#_Toc75946619)

[Start Replication 24](#_Toc75946620)

[Replication Detail 24](#_Toc75946621)

[Events Monitoring 24](#_Toc75946622)

[Replication Health Events 25](#_Toc75946623)

[Replication Database Events 25](#_Toc75946624)

[Drills & BCOs 26](#_Toc75946625)

[Normal Full Copy (NFC) 26](#_Toc75946626)

[Normal Copy 26](#_Toc75946627)

[Reverse Normal Copy 26](#_Toc75946628)

[Drills 26](#_Toc75946629)

[Switchover 26](#_Toc75946630)

[Switchback 27](#_Toc75946631)

[Failover Test Exercise 27](#_Toc75946632)

[BCO 27](#_Toc75946633)

[Failover 27](#_Toc75946634)

[Fallback 28](#_Toc75946635)

[Fallback Resync 28](#_Toc75946636)

[Validation Rules 28](#_Toc75946637)

[Recovery Automation Library 28](#_Toc75946638)

[Reports 28](#_Toc75946639)

# Introduction

This document details Disaster Recovery solution for Azure SQL DB(Database) PaaS(Platform as a Service) solution using Geo Replication as a replication mechanism.

## Scope

This document covers BCO(Failover) and DR drill (SwitchOver and SwitchBack) exercises using Azure Geo Replication as a replication mechanism. It explains the commands and the features of the Azure SQL PaaS DB with geo replication. It also contains the steps to perform drills and failover.

## References

* Azure SQL Database documentation - <https://docs.microsoft.com/en-us/azure/azure-sql/database/>
* Azure SQL DB PaaS documentation - <https://docs.microsoft.com/en-us/azure/azure-sql/database/sql-database-paas-overview>
* Active Geo Replication documentation - <https://docs.microsoft.com/en-us/azure/azure-sql/database/active-geo-replication-overview>

## Abbreviations

|  |  |
| --- | --- |
| **Abbreviation** | **Description** |
| RTO | Recovery Time Object |
| IBM RO | IBM Resiliency Orchestration |
| PaaS | Platform as a Service |
| IBM RO SC | IBM Resiliency Orchestration Site Controller |

## Technology Overview

This solution is based on Azure SQL DB (database) PaaS (Platform as a Service) offering from Azure and protected using Active Geo Replication feature to provide a DR solution. There is Azure SQL server on one region (considered as production), which hosts SQL DB protected to a target region on a target Azure SQL server using Active Geo Replication.

### Azure SQL

Azure SQL is a family of managed, secure, and intelligent products that use the SQL Server database engine in the Azure cloud.

* **Azure SQL Database**: Support modern cloud applications on an intelligent, managed database service, that includes serverless compute.
* **Azure SQL Managed Instance**: Modernize existing SQL Server applications at scale with an intelligent fully managed instance as a service, with almost 100% feature parity with the SQL Server database engine. Best for most migrations to the cloud.
* **SQL Server on Azure VMs**: Lift-and-shift your SQL Server workloads with ease and maintain 100% SQL Server compatibility and operating system-level access.

Azure SQL is built upon familiar SQL Server engine, so that you can migrate applications with ease and continue to use the tools, languages, and resources you are familiar with. Your skills and experience transfer to the cloud, enabling you to perform more.

### Azure SQL Database

Azure SQL database is a relational database-as-a-service (DBaaS) hosted in Azure that falls into the industry category of Platform-as-a-Service (PaaS).

* Best for modern cloud applications that uses the latest stable SQL Server features and have time constraints in development and marketing.
* A fully managed SQL Server database engine is based on the latest stable Enterprise Edition of SQL Server. SQL Database has two deployment options built on standardized hardware and software that is owned, hosted, and maintained by Microsoft.

With SQL Server, you can use built-in features and functionality that requires extensive configuration (either on-premise or in an Azure virtual machine). While using SQL Database, you pay-as-you-go with options to scale up or out for greater power with no interruption. SQL Database has additional features that are not available in SQL Server, such as built-in high availability, intelligence, and management.

Azure SQL Database offers the following deployment options:

* As a [single database](https://docs.microsoft.com/en-us/azure/azure-sql/database/single-database-overview) with its own set of resources managed via a [logical SQL server](https://docs.microsoft.com/en-us/azure/azure-sql/database/logical-servers). A single database is similar to a [contained database](https://docs.microsoft.com/en-us/sql/relational-databases/databases/contained-databases) in SQL Server. This option is optimized for modern application development of new cloud-born applications. [Hyperscale](https://docs.microsoft.com/en-us/azure/azure-sql/database/service-tier-hyperscale) and [serverless](https://docs.microsoft.com/en-us/azure/azure-sql/database/serverless-tier-overview) options are available.
* An [elastic pool](https://docs.microsoft.com/en-us/azure/azure-sql/database/elastic-pool-overview), which is a collection of databases with a shared set of resources managed via a [logical SQL server](https://docs.microsoft.com/en-us/azure/azure-sql/database/logical-servers). Single databases can be moved into and out of an elastic pool. This option is optimized for modern application development of new cloud-born applications using the multi-tenant SaaS application pattern. Elastic pools provide a cost-effective solution for managing the performance of multiple databases that have variable usage patterns.

Azure SQL Database is a fully managed platform as a service (PaaS) database engine that handles most of the database management functions such as upgrading, patching, backups, and monitoring without user involvement. Azure SQL Database is always running on the latest stable version of the SQL Server database engine and patched OS with 99.99% availability. PaaS capabilities that are built into Azure SQL Database enable you to focus on the domain-specific database administration and optimization activities that are critical for your business.

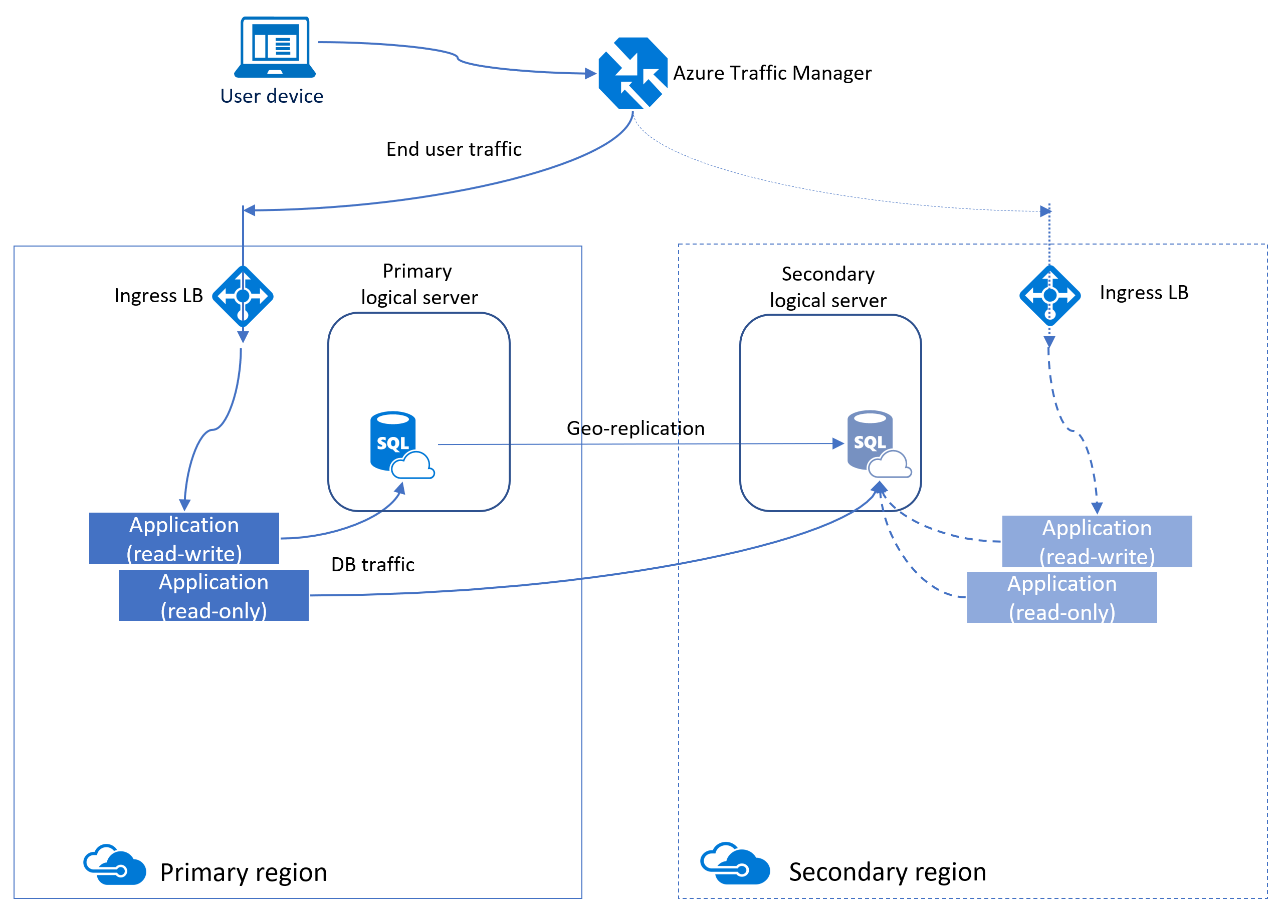
With Azure SQL Database, you can create a highly available and high-performance data storage layer for the applications and solutions in Azure. SQL Database can be the right choice for a variety of modern cloud applications as it enables you to process both relational data and [non-relational structures](https://docs.microsoft.com/en-us/azure/azure-sql/multi-model-features), such as graphs, JSON, spatial, and XML.

# Active Geo-Replication

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 centre (region).

Active geo-replication is designed as a business continuity solution that allows the application to perform quick disaster recovery of individual databases in case of a regional disaster or large-scale outage. If geo-replication is enabled, the application can initiate failover to a secondary database in a different Azure region. Up to four secondaries are supported in the same or different regions, and the secondaries can also be used for read-only access queries. The failover must be initiated manually by the application or the user. After failover, the new primary has a different connection end point.

The following diagram illustrates a typical configuration of a geo-redundant cloud application using Active geo-replication.



If for any reason the primary database fails, or simply needs to be taken offline, you can initiate failover to any of your secondary databases. When failover is activated to one of the secondary databases, all other secondaries are automatically linked to the new primary.

Active geo-replication leverages the [Always On availability group](https://docs.microsoft.com/en-us/sql/database-engine/availability-groups/windows/overview-of-always-on-availability-groups-sql-server) technology of the database engine to asynchronously replicate committed transactions on the primary database to a secondary database using snapshot isolation. Auto-failover groups provide the group semantics on top of active geo-replication, but the same asynchronous replication mechanism is used. While at any given point, the secondary database might be slightly behind the primary database, the secondary data is guaranteed to never have partial transactions. Cross-region redundancy enables applications to quickly recover from a permanent loss of an entire datacentre or parts of a datacentre caused by natural disasters, catastrophic human errors, or malicious acts.

## Active geo-replication terminology and capabilities

### Automatic Asynchronous ****Replication****

You can only create a secondary database by adding to an existing database. The secondary database can be created in any server. Once created, the secondary database is populated with the data copied from the primary database. This process is known as **Seeding**. After secondary database has been created and seeded, updates to the primary database are asynchronously replicated to the secondary database automatically. Asynchronous replication means the transactions are committed on the primary database before they are replicated to the secondary database.

### Readable Secondary Databases

An application can access a secondary database for read-only operations using the same or different security principals used for accessing the primary database. The secondary databases operate in snapshot isolation mode to ensure replication of the updates of the primary (log replay) is not delayed by queries executed on the secondary.

### Planned failover

Planned failover switches the roles of primary and secondary databases after the full synchronization is completed. It is an online operation that does not result in data loss. The time of the operation depends on the size of the transaction log on the primary that needs to be synchronized. Planned failover is designed for following scenarios:

* To perform DR drills in production when the data loss is not acceptable
* To relocate the database to a different region
* To return the database to the primary region after the outage has been mitigated (failback).

### Unplanned failover

Unplanned or forced failover immediately switches the secondary to the primary role without any synchronization with the primary. Any transactions committed to the primary but not replicated to the secondary will be lost. This operation is designed as a recovery method during outages when the primary is not accessible, but the database availability must be quickly restored. When the original primary is back online it will automatically re-connect and become a new secondary. All unsynchronized transactions before the failover will be preserved in the backup file but will not be synchronized with the new primary to avoid conflicts. These transactions must be manually merged with the most recent version of the primary database.

### Multiple readable secondaries

Up to four secondary databases can be created for each primary. If there is only one secondary database, and it fails, the application is exposed to higher risk until a new secondary database is created. If multiple secondary databases exist, the application remains protected even if one of the secondary databases fails. The additional secondaries can also be used to scale out the read-only workloads.

### Geo-replication of databases in an elastic pool

Each secondary database can separately participate in an elastic pool or not be in any elastic pool at all. The pool choice for each secondary database is separate and does not depend upon the configuration of any other secondary database (whether primary or secondary). Each elastic pool is contained within a single region, therefore multiple secondary databases in the same topology can never share an elastic pool.

### User-controlled failover and failback

A secondary database can explicitly be switched to the primary role at any time by the application or the user. During a real outage the "unplanned" option should be used, which immediately promotes a secondary to be the primary. When the failed primary recovers and is available again, the system automatically marks the recovered primary as a secondary and bring it up to date with the new primary. Due to the asynchronous nature of replication, a small amount of data can be lost during unplanned failovers if a primary fails before it replicates the most recent changes to the secondary. When a primary with multiple secondaries fails over, the system automatically reconfigures the replication relationships and links the remaining secondaries to the newly promoted primary without requiring any user intervention. After the outage that caused the failover is mitigated, it may be desirable to return the application to the primary region. To do that, the failover command should be invoked with the "planned" option.

# Solution Design

The solution describes “DR Solution for Azure SQL PaaS DB with Geo Replication”

### Architecture Diagram

With RO and SC hosted on Azure

Diagram

Description automatically generated

**With RO and SC hosted on on-premise infrastructure**

Diagram

Description automatically generated

This solution is designed for PaaS DB protection from one Azure region to another Azure region using Active Geo replication.

The solution provides the capabilities such as:

* Recovering PaaS DB
* Monitoring the SLA’s such as Recovery Point Objective and Recovery Time Objective at the designated recovery site.
* The IBM Resiliency Orchestration (IBM RO) is primarily deployed on the DR site/region representing recovery site/region and the IBM RO component can also be setup as high availability across two sites/regions. IBM RO deployed on Recovery site acts as Master and the component on Primary site acts as Standby state. IBM RO database replication occurs from Master to Standby using TCP Port 3306 which makes the database in sync state.
* As in diagram, each protected Azure SQL database on the primary Azure logical SQL server gets protected to the DR Azure logical SQL server with Active Geo Replication.
* IBM RO (Resiliency Orchestration) communicates with IBM RO SC (Site Controller), which in turn manages the communication to Azure components like the Azure SQL DB, Azure SQL logical server, and Active Geo replication as required.
* The IBM RO components, IBM RO, RO SC, RO HA etc can be either deployed in the respective Azure cloud regions or in the respective on-premise environments.
* Resiliency Orchestration supports Recovery operation with a set of standard workflows for FO, SO and SB.
* In case of Failover (FO), the force failover with data loss action is executed on Active Geo Replication, which brings up the Azure SQL database on the target/secondary region active for read/write access.
* In case of SwitchOver (SO), the failover without data loss action is executed on Active Geo Replication links, which checks for replication to be in sync (100% sync) and moves the production read/write capabilities to the Azure SQL database on the target/secondary region. The Azure SQL database on the primary region will be read only and a reverse replication will be initiated to be ready for a SwitchBack.
* In case of SwitchBack (SB), the failover without data loss action is executed on Active Geo Replication links, which checks for replication to be in sync (100% sync) and moves the production read/write capabilities to the Azure SQL database on the Primary Azure region. The Azure SQL database on the target/secondary region will be read-only and a replication from primary to secondary region will be initiated to be ready for any DR scenarios.

### IBM Resiliency Orchestration

IBM Resiliency Orchestration is an orchestration software hosted on Red Hat Enterprise Linux OS, which will enable the IBM BR task orchestration. IBM RO will be hosted at both PR and DR site.

### IBM Resiliency Orchestration (HA)

IBM RO will be deployed at both PR and DR site in HA with IBM RO server on DR site being the Primary/Master and the IBM RO server on the PR site in Standby.

# Site Controller

Site controller is an IBM RO agent management software server, which can either be hosted on a Microsoft Windows or RHEL platforms (depending on the type of agent that has to be supported) to manage different RO agents to reduce the bandwidth usage between IBM RO agents (installed on protected components managed by IBM RO sever) and IBM RO server.

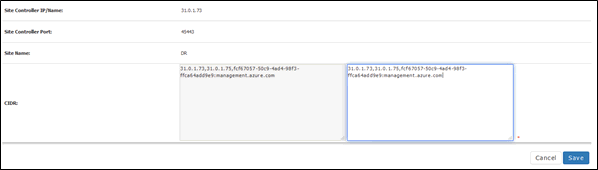
**Note:** Ensure you have internet access for connectivity to Azure.

### Site Controller mapping

Map the FQDN name “management.azure.com” to site controller running on the Azure region and its respective resource (vault) where you want to recovery the application as below.

Format : *<sitecontroller ip> == <tenant id>:management.azure.com*

For example:





**Note :** Site controller mapping status should be connected before discovering Azure Management service.

### Site Controller Configuration for multiple Azure SQL Paas DB

To configure multiple AzureSQLPaasDB management services on **Windows**, for the same host, ensure to prefix the database name to the hostname for unique identification.

In the SiteController mapping, instead of giving hostname alone, database name is prefixed to the hostname as follows:

*DATABASE\_NAME:HOST\_NAME*

**For example:** If the database name is 'SampleDB' and the host is 'abc.xyz.com', then the sitecontroller mapping becomes ***SampleDB:abc.xyz.com***

**Note:** Ensure you have internet access for connectivity to Azure.

### IBM RO Deployment Architecture at Azure Premises

### Site Controller Deployment

The number of site controller deployment depends on Azure VPN networking setup (i.e. Private subnets/Public subnets and Azure defined security groups or Network ACLs for subnets) and the number of supported workloads per site controller.

### IBM RO Master server deployment

* IBM RO server must communicate with deployed site-controllers on Azure and on-premise environment, should have security groups / firewall rules allowed for the required TCP ports.

**Note:** For the supported workloads, design, and Network port details, refer IBM Resiliency Orchestration guide.

# Supported Platforms and versions

For more information on supported platforms and versions, refer to the latest IBM Resiliency Orchestration Interop List document.

# Pre-requisites

The following are the prerequisites:

|  |  |  |
| --- | --- | --- |
| **Component** | **\*Ports - inbound** | **\*Ports – Outbound** |
| RO Master Server | 45443,3389,445,3306,8080,8443 | Any |
| RO Slave Server | 45443,3389,445,3306,8080,8443 | Any |
| RO Site Controller | 42443, 45443 | Any |
| RO Managed Component | 45443 | Any |

\* Configure and allow above ports bi-directional to the existing default cloud ports.

* Azure SQL DB needs to be configured and Active Geo replication should be configured and enabled. Active geo replication configuration and provisioning is not in IBM RO scope.
* All the prerequisite for the Active Geo replication mentioned in below document should be addressed.  
    
  <https://docs.microsoft.com/en-us/azure/azure-sql/database/active-geo-replication-overview>

### Security Limitations

The security limitations for this solution are as follows:

|  |  |  |  |
| --- | --- | --- | --- |
| Category | Requirements | Detail | Comments |
| Azure Cloud | Role with sufficient privileges to manage Active Geo Replication and SQL Server, SQL database and access to relevant subscription, resource group’s and resources to run REST API’s on these resources. | Ensure that the Azure Service principal credentials with the role set to Contributor or greater has been configured at the Azure subscription level.  Azure Service Principal Credential objects:   * Application (Client) ID * Password * Tenant ID   **Note:** Refer Microsoft documentation for more details. | Required for IBM Resiliency Orchestration to communicate with Azure Subscription (Primary and Target Azure Regions) for the managed resources discovery and protection for DR |
| Azure SQL database | Access/Credentials | Any account with view\_database\_state permission can query **sys.dm\_geo\_replication\_link\_status**.  https://docs.microsoft.com/en-us/sql/relational-databases/system-dynamic-management-views/sys-dm-geo-replication-link-status-azure-sql-database?view=azuresqldb-current#permissions  ***Example:***  SELECT  link\_guid  , partner\_server  , last\_replication  , replication\_lag\_sec  FROM sys.dm\_geo\_replication\_link\_status; | For IBM RO to provide the RPO details |

# Known Limitations

### Limitations

* The solution does not support FOTE.
* This solution does not support SQL DB with elastic pool.
* The solution supports only one-to-one Azure SQL DB active Geo Replication use case.
* The solution supports single DB recovery model only.
* This solution has not been tested with RO components on the Azure cloud.

# Auto Discovery

### Management service

Discover the following services for primary and remote sites.

### Adding Azure Management Service

Azure management service must be added to complete discovery of the Azure cluster.

To create an Azure Management Service, follow these steps:

1. Login to IBM Resiliency Orchestration.
2. Click **Discover** -> **Management Service** tab. The existing management services, if any, are listed in the **Management Service List** page.
3. Click **Add Management Service** link. The **New Management** **Service** page is displayed.
4. Select a **Resiliency Orchestration Site** from the **Site** drop-down list.
5. Select **Azure** as the **Management Service** from the drop-downas shown in the image below.

Graphical user interface, application

Description automatically generated

**Note:** The page refreshes with the relevant fields for creating an **Azure Management Service** and is as shown in the image below.

Graphical user interface, text, application

Description automatically generated

1. Select Site, Management Service Type and Site Controller Component from the respective drop-down list.
2. Enter Management Service Name, Client ID, Password, and Tenant ID.
3. Test the credentials and connectivity using the Test Credentials button. (Do not click on the test credentials)
4. Click **Save.**

**Note:**

* To view the Azure credentials, navigate to the Management Service page, and click on the Management Service name.
* To update Azure credentials (password only), on the Management Service page, click on the **Edit** icon corresponding to the Management Service name, update the password field, and then click the **Save** button.

### Configure Azure SQL PaaS DB Management Service

This Management Service is created before creating the group of the type ***AzurePaaSDBWithGeoReplication***. While creating the group, Monitoring Service must be defined in the JSON file.

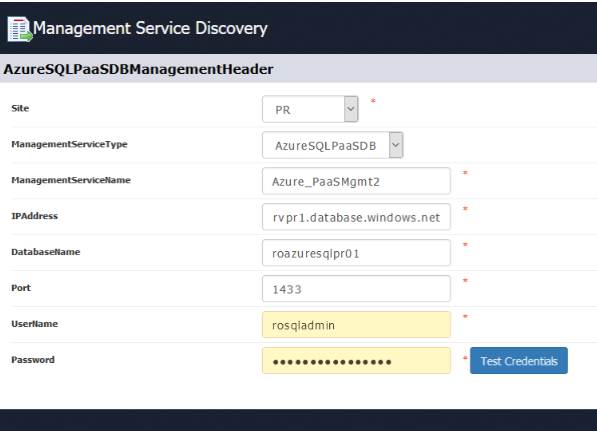
**For Example:** If PaasDBTest is a PaasDB Management Service, then the following entry must be placed under the managementServices section in the JSON:

"name": "PaasDBTest ",

"type": "MonitoringService"

This ensures that the Monitoring Services use PaasDBTest Management Service for making calls to PaasDB.

The AzureSQL PaaS DB Management Service image is displayed below for reference:

****

### Multiple Management Services used for monitoring

If PR and DR database names are different, then, you must create a separate management service of type AzureSQLPaasDB for each database. These database names must be assigned to two different keys namely, *RPOServicePR, RPOServiceDR*, under the additionalProperties section, in Group creation JSON.

**For example:** If PR AzureSQLPaasDB is Azure\_PaaSMgmt1 and DR AzureSQLPaasDB is Azure\_PaaSMgmt2, then the following key/values must to be added in the Additional Properties Section  
*{  
"key": "RPOServicePR",  
"value": "Azure\_PaaSMgmt1"  
},  
{  
"key": "RPOServiceDR",  
"value": "Azure\_PaaSMgmt2"  
}*

**Note:** If these keys are not defined for a group, then RPO computation will always contact PR database to get the RPO information regardless of the group state.

### Group Creation using Discovery Simplification Service (DSS) Tool

**Note:**

* Refer to the DSS user guide/ install guide for setting up the DSS tool and starting the required DSS servicesto use the DSS tool GUI.
* This is an internal tool for Services Professionals, who will have access to this tool during implementation.

In DSS tool CLI path *<DSS tool install path>/discovery-simplification/source/config*, update the file *Solution\_Configuration.csv* file with the required management service information that has been created.

**Format of the lines in the file is as follows:**

*<Solution name>, <management service name>,,,<site>,<site type>*

**Example for AzurePaaSDB solution is -**

*azurepaas,Azure\_Mgmt1,,,AzureMgPRSite,PR*

*Example output of the Solution\_Configuration.csv file:*

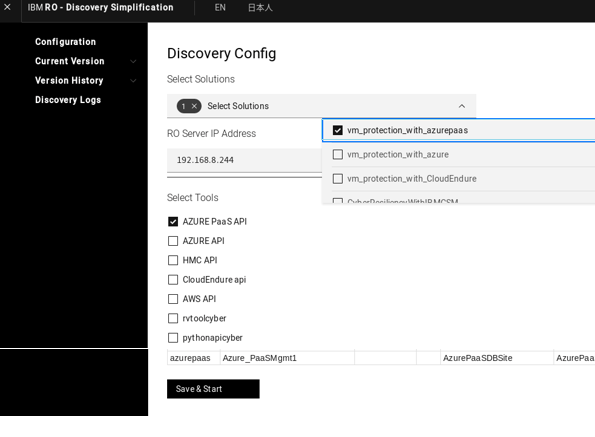
**Text

Description automatically generated**

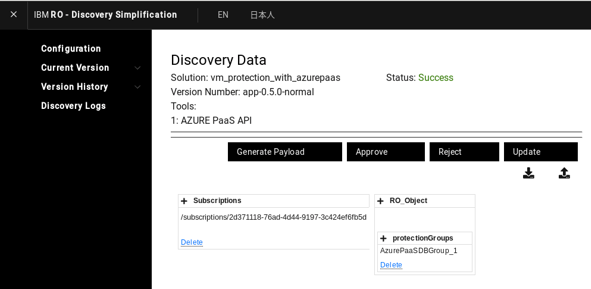
After the file is updated with the management service details, then use the DSS tool GUI to start discovery and ingestion for group creation.

Follow these steps in the DSS tool UI for discovery of the solution environment, validate the discovery information, and group creation in IBM RO.

1. Select the solution for which discovery needs to be performed from the drop down.
2. Update the RO server IP to which the discovery data needs to be ingested for group creation
3. Validate and enable the **Select Tool** checkbox for the relevant solution based on the selection made in step 1.
4. Click **Save & Start** to start the discovery process.



1. Follow these onscreen steps:
   1. Selecting/providing the correct credential information
   2. Selecting the correct resources (like resource group) for which discovery needs to be performed.
2. When the discovery is completed, click **Close** to go to the next step.
3. The next step is to validate the discovery information and continue to the payload generation step.
4. Click on **Generate payload** to generate the payload.
5. Validate the data in **RO\_Object** section and update any information that needs to be updated manually.
   1. The RO\_Object information is same as the JSON info as part of sample JSON provided in the solution.
   2. Resolve any warning that are raised as part of validation of this data by DSS tool.



1. After the validation and required details are updated and ready for group creation, click on the **Approve** option.
2. Approve option displays a screen to enter the RO details and credentials.
   1. Make sure to open the 8443 port on the RO server for data ingestion (based on firewall setup).
3. Based the RO credentials provided the ingestion starts and creates groups in the RO server in the backend.
4. After ingestion is complete, click **OK** to complete the DSS Tool Discovery and group creation operation.

**Limitation:**

1. Group creation works only for one RG at a time, to create multiple RGs at a time use the AG example template from Azure SR solution by just taking the application section before protection groups and modify the discovery JSON for DSS accordingly.
2. Example template is same as VM Protection with Azure SR solution example JSON which is available at <EAMSROOT>/rpd/solutionPlugins/AzureSR with file name ASRGroupConfig.json.

# Recovery Group

### Subsystem Discovery

This solution uses manual discovery, and RG creation is done by using a JSON file with the relevant fields in the JSON populated manually.

### Components

There are no component inputs required for this group creation. The group creation will happen without components and the relationship page will display dummy components with dummy IPs for representation purpose only. The dummy components do not relate any status and always display in green.

Graphical user interface, application

Description automatically generated

Graphical user interface, application

Description automatically generated

Sample JSON file for RG creation can be found in the following path:

*<EAMSROOT>/rpd/solutionPlugins/AzurePaaSDBWithGeoReplication/AzurePaaSDBWithGeoReplication\_GroupCreation\_Sample.json*

**Note:** All this information must be persistently stored. It would be used for the next operations like NFC, Replication Monitoring, DR DRILL/ Recovery operation, EVENTS.

# Monitoring

Monitoring includes the following items.

* RPO/RTO Monitoring
* Replication Monitoring

## RPO/RTO Monitoring

This section corresponds to **Monitor > RPO/RTO** screen of IBM RO.

### App RPO

App RPO is not applicable for this solution.

### Data RPO

To compute RPO, it is mandatory to create AzureSQLPaasDB management service with the PaasDB details.

To create the AzureSQLPaasDB management service, the following details are required to connect to the database using JDBC:

|  |  |
| --- | --- |
| **ManagementServiceName** | Any valid Management Service Name |
| **IPAddress** | Server Name or IPAddress |
| **DatabaseName** | Valid Database Name |
| **Port** | Concerned Port number |
| **UserName** | Database Username |
| **Password** | Database Password |

**Note:**

* The status of the *Management Service* becomes **ACTIVE** when it establishes a successful connection to the database with the above inputs.
* If it fails to establish a successful connection, then the status becomes **INACTIVE.**

When the valid AzureSQLPaaSDB is created, the RPO is fetched from PaaSDB MSSQL database by executing the following query:

SELECT *replication\_lag\_sec* FROM *sys.dm\_geo\_replication\_link\_status*

If the above query does not respond with valid value, RPO cannot be computed.

**Limitation:** Data RPO monitoring is not supported after SO.

### Recovery Time Objective

RTO is the estimated time required for the secondary/DR site to be up and available in case the primary goes down.

**Note:** Data RPO must be reported for each Recovery Group. It should be applicable to the Application Group.

## Replication Monitoring

This section corresponds to **Monitor > Replication** screen of IBM RO.

### Replication Status

Azure Geo Replication with **Catch\_up and seeding** status is in **Active** state, while the **Suspended** status is in **Inactive** state for the RG.

### Data lag

This section corresponds to **Monitor > Group selection > Pending Data**  screen of IBM RO.

The Data lag is measured for the RGs associated with the application. In this solution, it is measured in percentage (%) as follows:

**Computation Logic: 100 - (percentage of replication completed)**

### Stop Replication

Not Available

### Start Replication

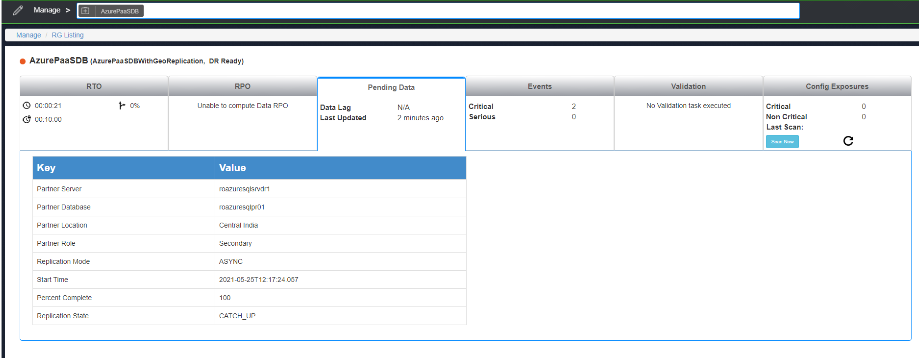
Not Available

### Replication Detail

It displays the Replication Details with the replication progress from the PR site to the DR site for the group. The Replication tab in the Group Dashboard of the desired Group displays the information related to the current replication status of that Group.

Replication Details of the Active Geo Replication is displayed as shown in the image below:

## Events Monitoring



This section corresponds to Events screen of IBM RO.

**Note:** Event raise condition is monitored, and an event is raised every four minutes.

### Replication Health Events

The following table displays the Events, Status, Raise Condition and Polar event

|  |  |  |  |
| --- | --- | --- | --- |
| **Events** | **Status** | **Raise Condition** | **Polar Event** |
| **BCSAzurePaasDB001** | Replication Active state | Raises when replication is either in **CATCH\_UP** status or in **SEEDING** status | * BCSAzurePaasDB002 (Replication is InActive) * BCSAzurePaasDB001 (Replication is Active) |
| **BCSAzurePaasDB002** | Replication InActive state | Raises when replication is in **SUSPENDED** status | * BCSAzurePaasDB001 (Replication is Active) |
| **BCSAzurePaasDB006** | Represents event where replication has stopped for long time | Raises when replication is unknown or logically when returns 404 status code while searching for replication state with Rest API | * BCSAzurePaasDB001 (Replication is Active) * BCSAzurePaasDB002 (Replication is InActive) |

### Replication Database Events

The following table displays the Events, Status, Raise condition and Polar event

|  |  |  |  |
| --- | --- | --- | --- |
| **Events** | **Status** | **Raise condition** | **Polar Event** |
| **BCSAzurePaasDB003** | Actual Primary database status is active | Raises when the actual primary database status is ‘Online’ | * BCSAzurePaasDB004 (Database is not online) * BCSAzurePaasDB005 (Database is unknown or not available) |
| **BCSAzurePaasDB004** | Actual primary database is not ‘Online’ | Raises when the actual primary database is not ‘Online’ | * BCSAzurePaasDB003 (Database is online) * BCSAzurePaasDB005 (Database is unknown or not available) |
| **BCSAzurePaasDB005** | Event represents primary database is unknown or not available | Raises when the actual primary database is unknown or not available | * BCSAzurePaasDB003 (Database is online) * BCSAzurePaasDB004 (Database is not online) |
| **BCSAzurePaasDB007** | Actual DR database status is active | Raises when the actual DR database status is ‘Online’ | * BCSAzurePaasDB008 (Database is not online) * BCSAzurePaasDB009   (Database is unknown or not available) |
| **BCSAzurePaasDB008** | Actual DR database is not ‘Online’ | Raises when the actual DR database is not ‘Online’ | * BCSAzurePaasDB007 (Database is online) * BCSAzurePaasDB009 (Database is unknown or not available) |
| **BCSAzurePaasDB009** | Event represents actual DR database is unknown or not available | Raises when the actual DR database is unknown or not available | * BCSAzurePaasDB007   (Database is online)   * BCSAzurePaasDB008 (Database is not online) |

# Drills & BCOs

Below steps for drills are based on the current topology that is supported and tested in our lab. End user steps may vary and the RALs must be developed accordingly based on the customer sequence followed.

**Important:** Ensure you include pre-checks for all the workflows.

**Note:** For the commands/ APIs, refer to the RAL Guide.

### Normal Full Copy (NFC)

Not available

### Normal Copy

Not available

### Reverse Normal Copy

Not available

## Drills

Switchover/Switchback

Switchover and Switchback is supported.

### Switchover

#### Prerequisites

* Azure Management service
* Resource group name
* DR Azure SQL logical Server name
* Azure SQL Database name
* Subscription ID
* DR Replication Link ID

#### Steps

1. Check DR region Azure SQL logical server role to be Secondary
2. Check Active Geo Replication to be in Sync (percentComplete = 100)
3. Azure SQL SwitchOver
4. Check DR region Azure SQL logical server role to be Primary

### Switchback

#### Prerequisites

* Azure Management service
* Resource group name
* DR Azure SQL logical Server name
* Azure SQL Database name
* Subscription id
* DR Replication Link ID

#### Steps

1. Check PR region Azure SQL logical server role to be Secondary
2. Check Active Geo Replication to be in Sync (percentComplete = 100)
3. Azure SQL SwitchBack
4. Check PR region Azure SQL logical server role to be Primary

## Failover Test Exercise

Not available.

## BCO

### Failover

**Prerequisites:**

* Azure Management service
* Resource group name
* DR Azure SQL logical Server name
* Azure SQL Database name
* Subscription id
* DR Replication Link ID

**Steps:**

1. Failover with data loss
2. Check DR region Azure SQL logical server role to be Primary

### Fallback

Not supported

### Fallback Resync

Not supported

### Validation Rules

Not supported

# Recovery Automation Library

Refer to the latest **IBM Resiliency Orchestration Recovery Automation Library** guide for more information about IBM RALs.

# Reports

You can generate reports on various continuity management and monitoring metrics that includes RPO, RTO, and continuity operations for Groups under IBM Resiliency Orchestration management. All applicable reports are available for this solution. To view all reporting related documentation, please refer to **Reports** section in **IBM Resiliency Orchestration Admin Guide**.