Oracle® GoldenGate

Using Oracle GoldenGate Maximum Availability Hub on Oracle Cloud Marketplace

Release 21.16
F93948-04
November 2024

Oracle GoldenGate Using Oracle GoldenGate Maximum Availability Hub on Oracle Cloud Marketplace, Release 21.16

F93948-04

Copyright © 2024, 2024, Oracle and/or its affiliates.

Primary Author: Jenny Chan

Contributing Authors: Katherine Wardhana

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software, software documentation, data (as defined in the Federal Acquisition Regulation), or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs (including any operating system, integrated software, any programs embedded, installed, or activated on delivered hardware, and modifications of such programs) and Oracle computer documentation or other Oracle data delivered to or accessed by U.S. Government end users are "commercial computer software," "commercial computer software documentation," or "limited rights data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, reproduction, duplication, release, display, disclosure, modification, preparation of derivative works, and/or adaptation of i) Oracle programs (including any operating system, integrated software, any programs embedded, installed, or activated on delivered hardware, and modifications of such programs), ii) Oracle computer documentation and/or iii) other Oracle data, is subject to the rights and limitations specified in the license contained in the applicable contract. The terms governing the U.S. Government's use of Oracle cloud services are defined by the applicable contract for such services. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle®, Java, MySQL, and NetSuite are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Inside are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Epyc, and the AMD logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

Contents

4 Use

5 Upgrade

6 Get help

1

Get started

Learn what you need to get started with Oracle GoldenGate Maximum Availability Hub on Oracle Cloud Marketplace.

Topics in this section:

- About Oracle GoldenGate Maximum Availability Hub on Oracle Cloud Marketplace
- About Oracle Cloud Marketplace

About Oracle GoldenGate Maximum Availability Hub on Oracle Cloud Marketplace

Oracle GoldenGate Maximum Availability Hub was designed to save you time in setting up and configuring your Oracle GoldenGate high availability solution.

It provides high availability by configuring a 2-node cluster server for fast and simple failover, and disaster recovery by leveraging Oracle Advanced Cluster File System (ACFS) replication to another identical GoldenGate hub server on a separate 2-node cluster server.

How it works

When you deploy Oracle GoldenGate Maximum Availability Hub on Oracle Cloud Marketplace, a terraform script creates the necessary networking resources, two 2-node RAC clusters, installs the grid infrastructure and XAG, and then the required resources to set up ACFS Replication between the two clusters.

Once the stack is deployed, the startup script runs and configures ACFS Replication, after which you can run ACFS role reversal. See Managing Planned Outages and Managing unplanned outages.

Overview of Oracle GoldenGate and supporting technologies

Explore the technologies required to replicate data between databases.

Oracle GoldenGate

Oracle GoldenGate provides real-time, log-based change data capture and delivery between homogenous and heterogeneous systems. It lets you construct a cost-effective and low-impact real-time data integration and continuous availability solution.

Oracle GoldenGate replicates data from committed transactions with transaction integrity and minimal overhead on your existing infrastructure. The architecture supports multiple data replication topologies, such as one-to-many, many-to-many, cascading, and bidirectional. Its wide variety of use cases includes real-time business intelligence; query offloading; zero-downtime upgrades and migrations; and active-active databases for data distribution, data synchronization, and high availability.

Oracle GoldenGate Microservices Architecture provides REST-enabled services. The REST-enabled services provide remote configuration, administration, and monitoring through HTML5 web pages, command line interfaces, and APIs.

Recommended Oracle GoldenGate 21c (and higher releases) introduces unified build support, so that a single software installation supports capturing and applying replicated data to multiple major Oracle Database versions (11g Release 2 to 21c). This is possible because an Oracle GoldenGate installation includes the required Oracle Database client libraries without requiring a separate database `ORACLE_HOME` installation.

Oracle Grid Infrastructure Agents

Oracle Grid Infrastructure Agents (XAG) are Oracle Grid Infrastructure components that provide the high availability (HA) framework to application resources and resource types managed through the agent management interface, AGCTL. This framework provides a complete, ready-to-use solution that contains pre-defined Oracle Grid Infrastructure resource configurations and agents to integrate applications for complete application HA.

The Oracle Grid Infrastructure Agents provide pre-defined Oracle Clusterware resources for Oracle GoldenGate, Siebel, Oracle PeopleSoft, JD Edwards, and Oracle WebLogic Server, as well as Apache and MySQL applications. Using the agent for Oracle GoldenGate simplifies the creation of dependencies on the source and target databases, the application VIP, and the Advanced Cluster File System (ACFS) mount point. The agent command line utility (AGCTL) is used to start and stop Oracle GoldenGate, and can also be used to relocate Oracle GoldenGate between the nodes in the cluster.

Oracle Advanced Cluster File System (ACFS)

Oracle ACFS can be used to store Oracle GoldenGate files.

Oracle Advanced Cluster File System (Oracle ACFS) is a multi-platform, scalable file system, and storage management technology that extends Oracle Automatic Storage Management (Oracle ASM) functionality to support all customer files.

Oracle ACFS leverages Oracle Clusterware for cluster membership state transitions and resource-based high availability. Oracle ACFS is bundled into the Oracle Grid Infrastructure (GI) allowing for integrated optimized management of databases, resources, volumes, and file systems.

Concepts

Familiarize yourself with common concepts and abbreviations you're sure to encounter when working with Oracle GoldenGate Maximum Availability Hub on Oracle Cloud Marketplace.

Concept or abbreviation	Definition
ACFS	Advanced cluster file system
ASM	Automatic storage management
ASMLIB	Oracle Database automatic storage management support library.
CRS	Cluster ready services
DNS	Domain name service
GI	Grid infrastructure
NGINX	Pronounced "engine x," an HTTP web server Oracle GoldenGate uses for reverse proxy.
OCI	Oracle Cloud Infrastructure

Concept or abbreviation	Definition
RAC	Real application clusters
SCAN	Single client access name
VCN	Virtual cloud network
VIP	Virtual IP
VM	Virtual machine
VNIC	Vritual network interface card
XAG	Oracle Grid Infrastructure standalone agent

About Oracle Cloud Marketplace

On Oracle Cloud Marketplace, you can find value-added applications and services that complement your existing Oracle Cloud solutions.

Resources

The Oracle GoldenGate Maximum Availability Hub image on Oracle Cloud Marketplace contains the latest Oracle GoldenGate 21c release that is available at the time of provisioning an instance.

The primary and secondary Oracle GoldenGate software is installed on the compute node under the `/u01/app/oracle/goldengate/gg21c` directory.

Supported Compute Shapes

Oracle GoldenGate runs on different OCI Compute Sizes. The following table provides details on the supported OCI Compute Sizes:

Compute Shape	OCPU	Memory (GB)
VM.Standard2.1	1	15
VM.Standard2.4	4	60
VM.Standard2.8	8	160
VM.Standard2.16	16	240
VM.Standard2.24	24	320
VM.Standard3.Flex	32	512 GB
VM.Standard.E4.Flex	64	1024 GB
VM.Standard.E5.Flex	64	1024 GB

For pricing details on compute nodes, please refer to OCI Pricing. For more information about OCPU, memory, and network bandwidth for extended memory VM shapes, see Extended Memory VM Instances.

Block Storage

The following table provides details on the default block storage configuration used by Oracle GoldenGate on Oracle Cloud Marketplace.

Volume	Default Size	Configurable
Boot	200GB	No

Volume	Default Size	Configurable
Swap	256GB	Yes
Deployments	1024GB	Yes

For pricing details on block storage, please refer to Oracle Storage Cloud Pricing.

Cross Cloud Functionality

Oracle GoldenGate is designed for large scale, cloud based architectures and Oracle GoldenGate on Oracle Cloud Marketplace is a key to many cloud-based solutions. By using Oracle GoldenGate from the Oracle Cloud Marketplace, you can replicate data from on premise to the Oracle Cloud, between data points within the Oracle Cloud, or even between third party clouds.

Network Recommendations

The network recommendations listed here primarlily apply for Oracle to Oracle replication, but the general rules apply to all platforms supported by Oracle GoldenGate. This applies to Oracle GoldenGate on-premise, in 3rd party clouds, in the Oracle cloud, and OCI-GoldenGate.

- **For Capture**: If you run Oracle GoldenGate remotely (on a separate server from the database that Oracle GoldenGate is capturing from), then the round trip ping time must be less than 80ms. For Integrated Extract, only the changes to tables that are being captured are sent to the Extract process itself.
- **For Replicat**: If you run Oracle GoldenGate remotely (on a separate server from the database that Oracle GoldenGate is applying changes to) then the round trip ping time must be less than 5ms.
- **For Oracle GoldenGate to Oracle GoldenGate communication**: The Oracle GoldenGate trail files grows at about 30-40% of the generated redo log volume (if you are capturing 100% of the data). That means that Oracle GoldenGate sends about 30-40% of the generated redo log volume across the network. If the network is not able to scale to this volume, then you can enable compression on the trail file data being sent across the network. This compression can typically achieve 8:1 compression ratios or better. You can also modify the TCP window socket size and buffers as well.

The following table provides a matrix on cross cloud support for replication by using Oracle GoldenGate on the Oracle Cloud Marketplace:

Table 1-1 GoldenGate Cross-Cloud Support

Technology	Remote Capture	Remote Apply
Oracle Cloud	Yes	Yes
Amazon Web Services (AWS)	Yes	Yes
Microsoft Azure	Yes	Yes

2
Provision

Learn how you can find and deploy Oracle GoldenGate Maximum Availability Hub on Oracle Cloud Marketplace.

Topics in this section:

Before you begin

What you need

Here are the prerequisites required to deploy Oracle GoldenGate Maximum Availability Hub:

- Oracle Cloud Account
- Access to an assigned Oracle Cloud Tenant
- Policies to create compute node resources within the Oracle Cloud Tenant
- Local SSH/RSA Key

Create an SSH/RSA Key

To work with the Oracle Cloud Infrastructure once the Oracle GoldenGate Compute Node is built, you have to provide a SSH Public Key during the interview process that will allow you to log in to the node once built.

In order to build your SSH keys, perform the following steps:

1. Open a Terminal window and start the key generation program by typing the following command:

   ```
   $ ssh-keygen
   ```

2. Enter the path to store this file. By default, this gets saved in your home directory under a hidden folder called `.ssh`. Change this default location, if required.

   ```
   Enter file in which to save the key (/Users/johndoe/.ssh/id_rsa): <Return>
   ```

3. Enter a passphrase for using your key.

   ```
   Enter passphrase (empty for no passphrase): <passphrase>
   ```

4. Re-enter the passphrase to confirm it.

```
Enter same passphrase again: <passphrase>
```

5. Check the results.
The key fingerprint (a colon separated series of 2 digit hexadecimal values) is displayed. Check if the path to the key is correct. In the above example, the path is `/Users/johndoe/.ssh/id_rsa.pub`. You have now created a public or private key pair.

> **Note:**
>
> For generating key pair on Windows platform, refer to Creating a Key Pair section in *Oracle Cloud Infrastructure Documentation*.

Required policies

Review the following information before you proceed:

- Creating a dynamic group
- Tags and Tag Namespace Concepts
- Creating a policy

Add the following required policies before you deploy the Oracle GoldenGate Maximum Availability Hub stack. You may need assistance from your Service administrator to add these policies to your compartment.

- Allow group <*ggowner*> to manage *instance-family* in compartment <*Compartment Name*>
- Allow group <*ggowner*> to manage *orm-family* in compartment <*Compartment Name*>
- Allow group <*ggowner*> to manage *volume-family* in compartment <*Compartment Name*>
- Allow group <*ggowner*> to use *virtual-network-family* in compartment <*Compartment Name*>
- Allow group <*ggowner*> to manage *public-ips* in compartment <*Compartment Name*>
- Allow group <*ggowner*> to use *tag-namespaces* in tenancy
- Allow group <*ggowner*> to inspect *compartments* in tenancy

Where <*ggowner*> is an example for a group and <*Compartment Name*> is an example of a compartment. The following are permission names: *instance-family*, *orm-family*, *volume-family*, *virtual-network-family*, and *public-ips*.

> **Note:**
>
> - The Networks compartment is an assumption that the customers follow the practice of having a separate network group manage the network resources for all users in the tenancy. If the tenancy instead allows you to create network resources of your own, then the policy would be: `Allow group <marketplace-permissions> to manage virtual-network-family in compartment <Marketplace-Test>`.

Use one of the following examples to assign privileges required for VIP reassignment

- Create a dynamic group, `OracleIdentityCloudService/VIP-Reassignment`, with the following rule for any compartment that requires access:

  ```
  Any {Instance.compartment.id  = '<Compartment OCID>'}
  ```

  For each compartment listed, add the following required policy for the dynamic group to use APIs to reassign the VIP to another instance in failover events:

  ```
  Allow dynamic-group 'OracleIdentityCloudService'/'VIP-Reassignment' to
  { PRIVATE_IP_READ, PRIVATE_IP_UPDATE, VNIC_ASSIGN, VNIC_UNASSIGN,
  VNIC_ATTACHMENT_READ, INSTANCE_INSPECT } in compartment
  <child_compartment_name>
  ```

- Instances created by the Oracle GoldenGate Maximum Availability Hub stack are tagged with the tag namespace, `GG_DEV`, and tag key, `ogg-high-availability`.

  Create the tag namespace, `GG_DEV` in the compartment in which you deploy Oracle GoldenGate Maximum Availability Hub. Create the tag key definition `ogg-high-availability` in the `GG_DEV` namespace. Create a dyamic group, `OracleIdentityCloudService/VIP-Reassignment-Tag`, with the following matching rule to group all instances tagged with the given namespace and tag key:

  ```
  tag.GG_DEV.ogg-high-availability.value
  ```

  Add the following required policy for the dynamic group that assigns privileges to all instances with this namespace and tag, enabling them to reassign the VIP address to other instances. For example:

  ```
  Allow dynamic-group 'OracleIdentityCloudService'/'VIP-Reassignment-Tag' to
  { PRIVATE_IP_READ, PRIVATE_IP_UPDATE, VNIC_ASSIGN, VNIC_UNASSIGN,
  VNIC_ATTACHMENT_READ, INSTANCE_INSPECT } in compartment
  <child_compartment_name>
  ```

Set up the source and target databases for replication

Before you can start replicating data, you should prepare the source or target database to support Oracle GoldenGate. For more information about steps to prepare your Oracle database, see Preparing the Database for Oracle GoldenGate in the *Using Oracle GoldenGate for Oracle Database* Guide.

Create a custom Virtual Cloud Network (VCN)

You can use an existing VCN or create one when you deploy the Oracle GoldenGate Maximum Availability Hub stack, but ensure that the VCN includes the following network configurations.

> **Note:**
>
> Whether you create a custom VCN or use an existing one, ensure that you're in the same compartment as the instances, or a child compartment of the same parent that hosts the instances. Instances cannot be in an unrelated parent compartment from the VCN compartment.

Before you begin

Take note of the following:

- When you create your VCN, you must create both a client subnet and a cluster subnet. The client subnet can be either public, which allows public access to instances created in the subnet, or private, which prohibits public IP address for instances created in the subnet. The cluster subnet is used only for internal communication between clusters, and must be private.
- If your client subnet is public, you must create and use an Internet Gateway. If your client subnet is private, then you must create and use a NAT Gateway.
- Two sets of security lists and route table rules are required, one set for the client subnet and one set for the cluster subnet. You can use the default security list and route table created when you create the subnet, and create a second security list and route table for the other subnet, or create two new security lists and route tables for each subnet, ensuring that the required ingress, egress, and route table rules are included as documented below.

To create a custom VCN:

1. Log in to the Oracle Cloud console with your Oracle Cloud account, if you're not already logged in.
2. Create the VCN:
   a. Open the Oracle Cloud navigation menu, click **Networking**, and then click **Virtual cloud networks**.
   b. On the Virtual Cloud Networks in Compartment page, click **Create VCN**.
   c. In the Create Virtual Cloud Network panel, complete the following fields:
      i. For **Name**, enter a name for the VCN, such as `VCN01`.
      ii. Select a **compartment** in which to create the VCN.
      iii. For **IPv4 CIDR Blocks**, enter an IPv4 CIDR block such as, `10.10.0.0/16`, and then press Enter on your keyboard.
   d. Click **Create VCN**.
3. Create Gateways:
   - Create an Internet Gateway, if the client subnet's access type is public:

a. On the Virtual Cloud Network details page, under **Resources**, click **Internet Gateways**.

b. Click **Create Internet Gateway**.

c. In the Create Internet Gateway panel, enter a name for the Internet Gateway, such as `igwy01`, and then click **Create Internet Gateway**.

- Create a NAT Gateway for the cluster subnet, or if the client subnet's access type is private:

a. Use the breadcrumb to return to the VCN details page.

b. On the Virtual Cloud Network details page, under **Resources**, click **NAT Gateways**, and then click **Create NAT Gateway**.

c. In the Create NAT Gateway panel, enter a name for the NAT Gateway, such as `ngwy01`, and then click **Create NAT Gateway**.

4. Create Route Tables and add Route Rules:

a. Create a Route Table for the client subnet:

i. On the Virtual Cloud Network details page, under **Resources**, click **Route Tables**, and then click **Create Route Table**.

ii. For In the Create Route Table panel, enter a name for the Route Table, such as `client_rt01`, and then click **Create**.

iii. Select the newly created route table.

iv. On the Route Table Details page, click **Add Route Rules**.

v. In the Add Route Rules panel, complete the fields as follows:

i. For **Target Type**, select:

- **Internet Gateway**, if your client subnet is public.
- **NAT Gateway**, if your client subnet is private.

ii. For Destination CIDR Block, enter `0.0.0.0/0`

iii. For Target, select **Internet Gateway** from the dropdown.

vi. Click **Add Route Rules**.

b. Create a Route Table for the cluster subnet:

i. On the Virtual Cloud Network details page, under **Resources**, click **Route Tables**, and then click **Create Route Table**.

ii. For In the Create Route Table panel, enter a name for the Route Table, such as `cluster_rt01`, and then click **Create**.

iii. Select the newly created route table.

iv. On the Route Table Details page, click **Add Route Rules**.

v. In the Add Route Rules panel, complete the fields as follows:

i. For **Target Type**, select **NAT Gateway**.

ii. For Destination CIDR Block, enter `0.0.0.0/0`

iii. For Target, select **Internet Gateway** from the dropdown.

vi. Click **Add Route Rules**.

5. Create Security Lists:

a. Use the breadcrumb to return to the VCN details page.
b. On the Virtual Cloud Network details page, under **Resources**, click **Security Lists**.
c. Create a Security List for the client subnet:
    i. Click **Create Security List**.
    ii. In the Create Security List panel, complete the fields as follows:
        i. For **Name**, enter `client_sl01`.
        ii. Under **Allow Rules for Ingress**, click **+ Another Ingress Rule**.
        iii. For Ingress Rule 1,
            i. For **Source Type**, select **CIDR**.
            ii. For **Source CIDR**, enter `10.10.0.0/24`.
            iii. For **IP Protocol**, select **ICMP** from the dropdown.
            iv. For **Type**, enter `8`.
            v. For Description, enter `Required for ACFS replication`.
            vi. Click **+ Another Ingress Rule**
        iv. For Ingress Rule 2,
            i. For **Source Type**, select **CIDR**.
            ii. For **Source CIDR**, enter the client subnet CIDR. For example, `10.10.0.0/24`.
            iii. For **Source Port Range**, enter `All`
            iv. For **Destination Port Range**, enter `All`.
            v. For **IP Protocol**, select **TCP** from the dropdown.
            vi. For Description, enter `Required for GI communication`.
            vii. Click **+ Another Ingress Rule**
        v. For Ingress Rule 3,
            i. For **Source Type**, select **CIDR**.
            ii. For **Source CIDR**,
                - If the client subnet is public, enter `0.0.0.0/0`.
                - If the client subnet is private, enter `10.10.0.0/24`
            iii. For **Source Port Range**, enter `All`
            iv. For **Destination Port Range**, enter `22`.
            v. For **IP Protocol**, select **TCP** from the dropdown.
            vi. For Description, enter `Required for SSH`.
        vi. For Ingress Rule 4,
            i. For **Source Type**, select **CIDR**.
            ii. For **Source CIDR**,
                - If the client subnet is public, enter `0.0.0.0/0`.
                - If the client subnet is private, enter `10.10.0.0/24`

iii. For **Source Port Range**, enter `443`
iv. For **Destination Port Range**, enter `443`.
v. For **IP Protocol**, select **TCP** from the dropdown.
vi. For Description, enter `Required for web access to GoldenGate`.
vii. Under **Allow Rules for Egress**, click **+ Another Egress Rule**.
viii. For Egress Rule 1,
i. For **Destination Type**, select **CIDR**.
ii. For **Destination CIDR**, enter `0.0.0.0/0`.
iii. For **IP Protocol**, select **All Protocols**.
iii. Click **Create Security List**.

d. Create a Security List for the cluster subnet.
i. Click **Create Security List**.
ii. In the Create Security List panel, complete the fields as follows:
i. For **Name**, enter `cluster_sl01`.
ii. Under **Allow Rules for Ingress**, click **+ Another Ingress Rule**.
iii. For Ingress Rule 1,
i. For **Source Type**, select **CIDR**.
ii. For **Source CIDR**, enter `10.10.1.0/24`.
iii. For **IP Protocol**, select **ICMP** from the dropdown.
iv. For **Type**, enter `All`.
v. For **Code**, enter `All`.
vi. Click **+ Another Ingress Rule**
iv. For Ingress Rule 2,
i. For **Source Type**, select **CIDR**.
ii. For **Source CIDR**, enter `10.10.1.0/24`.
iii. For **Source Port Range**, enter `All`.
iv. For **Destination Port Range**, enter `All`.
v. For **IP Protocol**, select **TCP** from the dropdown.
vi. Click **+ Another Ingress Rule**
v. For Ingress Rule 3,
i. For **Source Type**, select **CIDR**.
ii. For **Source CIDR**, enter `10.10.1.0/24`.
iii. For **Source Port Range**, enter `All`.
iv. For **Destination Port Range**, enter `All`.
v. For **IP Protocol**, select **UDP** from the dropdown.
vi. Click **+ Another Ingress Rule**
vi. Under **Allow Rules for Egress**, click **+ Another Egress Rule**.

vii. For Egress Rule 1,

i. For **Destination Type**, select **CIDR**.

ii. For **Destination CIDR**, enter `0.0.0.0/0`.

iii. For **IP Protocol**, select **All Protocols**.

viii. Click **Create Security List**.

6. Create the client subnet:
   a. Use the breadcrumb to return to the VCN details page.
   b. On your Virtual Cloud Network details page, under **Resources**, click **Subnets**.
   c. In the Subnets list, click **Create Subnet**.
   d. In the Create Subnet panel, complete the following fields:
      i. For **Name**, enter a name for the subnet, such as `clientsubnet001`.
      ii. For **Create in Compartment**, select the compartment in which to create the subnet.
      iii. For **Subnet Type**, select **Regional**.
      iv. For **IPv4 CIDR Blocks**, enter `10.10.0.0/24`.
      v. For Route Table in Compartment, select the client Route Table created in step 4a (**client_rt01**).
      vi. (Optional) For **Subnet Access**, select one of the following:
         - **Publict Subnet**, to allow public IP addresses for instances created in this subnet.
         - **Private Subnet**, to prohibit public IP addresses for instances created in this subnet.
      vii. For Security Lists, select the client Security List created in step 5c (**client_sl01**).
   e. Click **Create Subnet**.
7. Create the cluster subnet:
   a. On your Virtual Cloud Network details page, click **Create Subnet**.
   b. In the Create Subnet panel, complete the following fields:
      i. For **Name**, enter a name for the subnet, such as `clustersubnet001`.
      ii. For **Create in Compartment**, select the compartment in which to create the subnet.
      iii. For **IPv4 CIDR Blocks**, enter an IPv4 CIDR block such as, `10.10.1.0/24`.
      iv. For Route Table in Compartment, select the client Route Table created in step 4b (**cluster_rt01**).
      v. For **Subnet Access**, select **Private Subnet**.
      vi. For Security Lists, select the client Security List created in step 5d (**client_sl01**).
   c. Click **Create Subnet**.
8. Create a private view:
   a. Use the Oracle Cloud console search bar to search for `private view`.
   b. In the search results, under **Services**, select **Private views** (DNS Management).

c. On the Private views page, click **Create private view**.

d. In the Create private view panel, enter `goldengate_dns_view`, and then click **Create**.

9. Create a zone:

a. Use the Oracle Cloud console search bar to search for `zones`.

b. In the search results, under **Services**, select **Zones** (DNS Management).

c. On the Zones page, click **Private zones**, and then click **Create zone**.

d. In the Create private zone panel, enter `goldengate.com`, and then click **Create**.

e. Ensure that the DNS private view selected is the private view created in step 8, and then click **Create**.

10. Update the associated DNS resolver:

a. Use the breadcrumb to return to the **Networking** page, and then select **Virtual cloud networks** from the Networking menu.

b. On the Virtual Cloud Networks page, select your VCN.

c. On the Virtual Cloud Network details page, in the **VCN information** card, locate **DNS Resolver**, and click the VCN name.

d. On the Private resolver details page, click **Manage private views**.

e. In the Manage private views panel, select the DNS private view created in step 8 from the dropdown, and then click **Save changes**.

Find Oracle GoldenGate Maximum Availability Hub on Oracle Cloud Marketplace

To locate Oracle GoldenGate Maximum Availability Hub on Oracle Cloud Marketplace:

1. Log in to Oracle Cloud Marketplace.
2. In the Search field, enter `Oracle GoldenGate`.
3. From the search results, select **Oracle GoldenGate Maximum Availability Hub on Oracle Cloud Marketplace**.

Deploy Oracle GoldenGate Maximum Availability Hub on Oracle Cloud Marketplace

After you find Oracle GoldenGate Maximum Availability Hub on Oracle Cloud Marketplace, you can deploy it using the provided Stack Listing. This TerraForm Stack prompts you for specific information and then builds the Oracle Cloud Infrastructure Compute Nodes with the desired hardware settings, configures the desired network settings, and starts the Grid Infrastructure and Oracle GoldenGate installations.

Before you begin

- Ensure that you add all required policies before you proceed.
- You have the option to create a new Virtual Cloud Network (VCN) or using an existing one. If using an existing VCN, ensure that the network configuration meets the requirements as documented in Create a custom Virtual Cloud Network (VCN).

> **Note:**
>
> Customizations to the stack settings are not kept intact when you upgrade the stack. Carefully consider whether customization is necessary, and ensure that you take note of custom settings so that you can reapply them after upgrading.

To deploy Oracle GoldenGate Maximum Availability Hub on Oracle Cloud Marketplace using the Stack Listing:

1. On the Oracle GoldenGate Maximum Availability Hub Application page, select **Get App**.
2. Select **Commercial Market** or **Government Market**, and then click **Sign In**.
3. Enter the tenancy name for **Cloud Account Name**, and then click **Next**.
4. Sign in to the Identity provider.
5. On the Oracle GoldenGate application page, provide the following information, and then click **Launch Stack**:
   - **Select Version** - It provides a list of available versions for the listing.
   - **Select Compartment** - Specifies the compartment where the compute node will be built. It is generally the location that you have access to build the compute node.
   - **Terms of Use** - Review the Oracle standard Terms and Restrictions, and then select the checkbox.
6. Fill in the required **Stack** information:
   - **Name** - Name of the Stack. It has a default name and provides a date time stamp. You can edit this detail, if required.
   - **Description** - Description of the Stack that you are creating.
   - **Create In Compartment** – It defaults to the compartment you have selected on the Oracle GoldenGate application page.
   - **Tags** (optional) – Tags are a convenient way to assign a tracking mechanism but are not mandatory. You can assign a tag of your choice for easy tracking. You have to assign a tag for some environments for cost analysis purposes.
   - Click **Next**.
7. Fill in the required details to configure variables. This information is required to build the cluster compute nodes with for Oracle GoldenGate Maximum Availability Hub.
   a. For **Name for New Resources**, enter:
      i. **Cluster Prefix**: The prefix to identify the cluster.

> **Note:**
>
> This prefix must be unique within the Compartment and meet the following requirements:
>
> - At least 1, but no more than 14 characters in length.
> - Contains only single-byte alphanumeric characters (upper or lowercase) or hyphens (-).
> - Must begin with a letter.
> - Can't start or end with a hyphen.

ii. **Hostname Prefix**: The prefix to identify an instance in the cluster. You can change the default prefix, but it must be between 2 and 8 alphanumeric characters (including hyphens) in length.

> **Note:**
>
> Two clusters are created. Two nodes are created in each cluster. Add details how to identify each cluster/node.

b. For **VCN Settings**, complete the following fields:

i. **Create New Network**: Select this check box if you wish to create a new network resource.

- If you select this check box, the **Create New Network** wizard appears allowing you to add and edit the new network information.
- If you do not select this check box, the Create New Network wizard does not appear and the compute node is created with the existing network options in the VCN.

ii. **VCN Network Compartment**: Compartment in which to create new or to use existing network resources.

iii. **New VCN DNS Name**: DNS Name to assign to new VCN.

iv. **New VCN CIDR**: A default CIDR is generated, but you can change it, if needed.

v. **Client Subnet Compartment**: Compartment in which to create new or to use existing client subnet.

vi. **Client Subnet DNS Name** (optional): Name assigned to the client subnet. You can leave this field blank if you want to create a subnet with DNS disabled.

vii. **Client Subnet CIDR**: A default CIDR is generated, but you can change it, if needed.

viii. **Cluster Subnet Compartment**: Compartment in which to create or to use existing cluster subnet.

ix. **Cluster Subnet DNS Name** (optional): Name assigned to cluster subnet. You can leave this field blank if you want to create a subnet with DNS disabled.

x. **Cluster Subnet CIDR**: A default CIDR is generated, but you can change it, if needed.

c. For **Client Subnet Settings**, complete the following fields:

i. Select **Private Subnet** to create or use the VCN's private subnet.

> **Note:**
>
> If you are using a private IP address to access the compute node, you have to set up an IPSec VPN or FastConnect connection. Refer to OCI documentation for more details.

ii. If Private Subnet is not selected, then **Assign Public IP** is selected by default, and the VCN's public subnet will be used.

iii. Select the **Client Subnet Compartment**.

iv. Select the **Client Subnet** from the dropdown.

d. Select the **Cluster Subnet Setting** from the dropdown. This private subnet is used only for internal cluster communication, and must be different from the Client subnet selection.

e. For **Instance Settings**, complete the following fields:

i. Select the primary cluster's availability domain from the **Primary Availability Domain** dropdown.

ii. Select the standby cluster's availability domain from the **Standby Availability Domain** dropdown.

iii. Select a **Compute Shape**. Supported shapes are:

- VM.Standard2.1
- VM.Standard2.4
- VM.Standard2.8
- VM.Standard2.16
- VM.Standard2.24
- VM.Standard3.Flex
- VM.Standard.E4.Flex
- VM.Standard.E5.Flex

> **Note:**
>
> If you select a Flex shape, then you must enter the **Number of OCPUs**, and **Amount of Memory (GB)**.

iv. For **Custom Volume Sizes**- Select this option to customize the size of the new block storage volumes that are built for the compute node.
Block Storage (Custom Volume Sizes) -

i. Swap Volume Size: Default value is 256GB

ii. Deployment Volume Size: Default value is 1024GB (includes Trail files, CacheManager, and Config files)

iii. Deployment Volume VPUs: Default value is 20 VPUs

f. For **Create Oracle GoldenGate Deployment**, complete the following fields:

i. **Deployment Name**: Name to assign to the Oracle GoldenGate Deployment.

> **Note:**
>
> Deployment name must:
>
> - Be no greater than 32 characters in length
> - Start with a lower or uppercase character
> - Contain only alphanumeric characters and underscores

ii. **Deployment - Autonomous Database**: Select if deployment connects to an Autonomous Database. If selected, select the **Compartment** in which the Autonomous Database resides, and then select the **Autonomous Database** instance to which the deployment will connect.

g. For **Shell Access**, paste the **SSH Public Key** to allow access as the `opc` user.

h. Click **Next**.

8. On the **Review** page, review the information you provided, select **Run apply** on the created stack, and then click **Create**.

You're brought to the **Stacks Job Details** page. The stack takes a few minutes to create. You can monitor the creation of the compute nodes in the Log section of the page. As the primary and standby clusters and their nodes are created, the following information is outputted to the log for each cluster:

- cluster_name
- domain
  - dns
  - ip_address
  - ip_type
- nodes
  - node1 hostname
  - node1 ip_address
  - node1 ip_type
  - node2 hostname
  - node2 ip_address
  - node2 ip_type

After the status changes to SUCCEEDED, you can view the compute nodes created under **Instances**.

To review all the resources created as a result of this job, see Review stack resources.

At this stage, the Marketplace Stack deployment includes only network resources, the compute instances and storage (four Virtual Machines (VMs) two for the primary cluster and two for the standby cluster), and allocation of block volume. Next, a startup script runs and installs the Grid Infrastructure (GI) and Oracle GoldenGate. SSH to the primary cluster's node1 as the `opc` user to monitor `/tmp/startupScript.log` and know when the deployment is online. See Monitor installation and startup for more information.

Review stack resources

After the stack creation completes, take a moment to review the resources that were created.

Review VCN details

Use the following steps to review the network resources created for Oracle GoldenGate Maximum Availability Hub.

To review network resources created:

1. Use the Oracle Cloud console navigation menu to navigate to **Networking**, and then **Virtual Cloud Networks**.
2. In the Virtual Cloud Networks list, select **`<cluster-prefix> VCN`**.
3. On the VCN details page, the Subnets section lists the private and public subnet that were created.
4. In the VCN Information section of the VCN detail page, click **DNS resolver**.
5. On the Private resolver details page, the Associated private views section lists created private view. Click **goldengate_dns_view**.
6. In the Private zones list, click **goldengate.com**.
7. In the Records section of the goldengate.com zone page, both the primary and standby dns records were added. Take note of the RDATA value for the primary cluster.

Review Compute Instances

Use the following steps to review the Compute Instances (nodes) created for Oracle GoldenGate Maximum Availability Hub.

To review the Compute Instances created:

1. Use the Oracle Cloud console navigation menu to navigate to **Compute**, and then **Instances**.
2. In the list of Instances, you can select a primary or standby node to view its details.

> **Note:**
>
> A total of four nodes were created, two for the primary cluster, and two for the standby cluster. The naming conventions are as follows:
>
> - Primary cluster nodes:
>   - `<cluster-prefix>1-<node-prefix>1`
>   - `<cluster-prefix>1-<node-prefix>2`
> - Standby cluster nodes:
>   - `<cluster-prefix>2-<node-prefix>1`
>   - `<cluster-prefix>2-<node-prefix>2`

3. Each node has public and private network interfaces. On the Instance's details page, click **Attached VNICs** to review this information.

> **Note:**
>
> The IP address that was shown in the DNS record for the primary cluster is the same as shown in the list of IPv4 Addresses of the primary VNIC for `<cluster-prefix>1-<node-prefix>1`.

4. Each node also has attached block volumes. On the Instance's details page, click **Attached Block Volumes** to view:
   - `cluster-gidisk`: Storage volume for the Grid Infrastructure. This is shared between the two nodes in the cluster.
   - `cluster-oggdisk`: Storage volume for Oracle GoldenGate. This is shared between the two nodes in the cluster.

> **Note:**
>
> Automatic storage management cluster file system (ACFS) replication of this storage volume occurs between the primary and standby clusters.

   - `<instance-name>-swap`: This is dedicated to the node and not shared.

Review Block Volumes

Use the following steps to review Block Volumes created for Oracle GoldenGate Maximum Availability Hub.

To review Block Volumes created:

> **Note:**
>
> You can also view Attached Block Volumes for each Compute Instance (node) from Compute Instance's details page.

1. Use the Oracle Cloud console navigation menu to navigate to **Storage**, and then **Block Volumes**.
2. On the Block Volumes page, all block volumes created as part of the stack job is listed here. Select a block volume to view its details:
   - `<cluster-prefix>1-oggdisk`
   - `<cluster-prefix>2-oggdisk`
   - `<cluster-prefix>1-gidisk`
   - `<cluster-prefix>2-gidisk`
   - `<cluster-prefix>1-<node-prefix>1-swap`
   - `<cluster-prefix>1-<node-prefix>2-swap`
   - `<cluster-prefix>2-<node-prefix>1-swap`

- `<cluster-prefix>2-<node-prefix>2-swap`

Next, monitor installation and startup.

Monitor installation and startup

It can take up to 20 minutes for the Grid Infrastructure (GI) installation to complete. SSH into node1 of the primary cluster to monitor this process.

To SSH into node1 of the primary cluster and monitor the GI installation:

1. Get the Public IP address of node1 of the primary cluster:
   a. In the Oracle Cloud console, in the navigation menu, click **Compute**, and then **Instances**.
   b. In the Instances list, locate your node1 of the primary cluster, and then copy its **Public IP**.

   > **Tip:**
   >
   > The naming convention of your primary cluster's node1 is `<cluster-prefix>1-<node-prefix>1`.

2. Use a terminal application or Cloud Shell to SSH into node1 of the primary cluster as the `opc` user.
3. Open `/tmp/startupScript.log` in a supported text editor, for example: `tail -f /tmp/startupScript.log`.
4. Monitor the messages outputted to this file until you see:

   ```
   ################################################################################
   #####
   Deployment is successful, all resources are online and ACFS replication is
   healthy
   ################################################################################
   #####
   ```

   This indicates that the GI installation completed, and Oracle GoldenGate is ready for you to use.

Next, configure the source and target databases.

3
Prepare

Learn how to prepare your maximum availability solution.

Topics in this section:

- Task 1: Configure the Source and Target Databases for Oracle GoldenGate
- Task 2: Configure the Oracle GoldenGate Environment

Task 1: Configure the Source and Target Databases for Oracle GoldenGate

The source and target Oracle GoldenGate databases should be configured using the following recommendations.

Perform the following steps to complete this task:

- Step 1.1 - Database Configuration
- Step 1.2 - Create the Database Replication Administrator User
- Step 1.3 - Create the Database Services

Step 1.1 - Database Configuration

The source and target Oracle GoldenGate databases should be configured using the following recommendations:

Configuration	Scope	Example

Enable Archivelog Mode	Source and Target	`SQL> ARCHIVE LOG LIST`
		`Database log mode Archive Mode`
		`Automatic archival Enabled`
		`Archive destination USE_DB_RECOVERY_FILE_DEST`
		`Oldest online log sequence 110`
		`Next log sequence to archive 113`
		`Current log sequence 113`

Enable `FORCE LOGGING`	Source and Target	`ALTER DATABASE FORCE LOGGING;`

Configuration	Scope	Example
`ENABLE_GOLDENGATE_REPLICATION`	Source, Target, and Standbys	`ALTER SYSTEM SET ENABLE_GOLDENGATE_REPLICATION=TRUE SCOPE=BOTH SID='*';`
Supplemental Logging	Source Required on Target for cases when replication reverses	`ALTER DATABASE ADD SUPPLEMENTAL LOG DATA;`
Add schema or table level logging for replicated objects	Source Required on Target for cases when replication reverses	`ADD SCHEMATRANDATA` or `ADD TRANDATA`
`STREAMS_POOL_SIZE`	Source Required on Target for cases when replication reverses	The value of `STREAMS_POOL_SIZE` should be set to the following value: `STREAMS_POOL_SIZE` = (((#Extracts + #Integrated Replicats) * 1GB) * 1.25) For example, in a database with 2 Extracts and 2 integrated Replicats: `STREAMS_POOL_SIZE` = 4GB * 1.25 = 5GB `ALTER SYSTEM SET STREAMS_POOL_SIZE=5G SCOPE=BOTH SID='*';`

For the steps on preparing the database for Oracle GoldenGate, see Preparing the Database for Oracle GoldenGate.

Step 1.2 - Create the Database Replication Administrator User

The source and target databases need a GoldenGate administrator user created, with appropriate privileges assigned as follows:

- For the multitenant container database (CDB):
  - Source database, GoldenGate Extract must be configured to connect to a user in the root container database, using a c##
  - Target database, a separate GoldenGate administrator user is needed for each pluggable database (PDB).
  - For further details on creating a GoldenGate administrator in an Oracle Multitenant Database, see Configuring Oracle GoldenGate in a Multitenant Container Database.
- For non-CDB databases, see Establishing Oracle GoldenGate Credentials

As the `oracle` OS user on the source database system, execute the following SQL instructions to create the database user for Oracle GoldenGate and assign the required privileges:

```
[opc@exadb1_node1 ~]$ sudo su - oracle
[oracle@exadb1_node1 ~]$ source dbName.env
[oracle@exadb1_node1 ~]$ sqlplus / as sysdba

# Source CDB
```

```
SQL>
alter session set container=cdb$root;
create user c##ggadmin identified by "ggadmin_password" container=all default
tablespace USERS temporary tablespace temp;
alter user c##ggadmin quota unlimited on users;
grant set container to c##ggadmin container=all;
grant alter system to c##ggadmin container=all;
grant create session to c##ggadmin container=all;
grant alter any table to c##ggadmin container=all;
grant resource to c##ggadmin container=all;
exec
dbms_goldengate_auth.grant_admin_privilege('c##ggadmin',container=>'all');

# Source PDB
SQL>
alter session set container=pdbName;
create user ggadmin identified by "ggadmin_password" container=current;
grant create session to ggadmin container=current;
grant alter any table to ggadmin container=current;
grant resource to ggadmin container=current;
exec dbms_goldengate_auth.grant_admin_privilege('ggadmin');
```

As the `oracle` OS user on the target system, execute the following SQL instructions to create the database user for Oracle GoldenGate and assign the required privileges:

```
[opc@exadb2_node1 ~]$ sudo su - oracle
[oracle@exadb2_node1 ~]$ source dbName.env
[oracle@exadb2_node1 ~]$ sqlplus / as sysdba

# Target PDB
SQL>
alter session set container=pdbName;
create user ggadmin identified by "ggadmin_password" container=current;
grant alter system to ggadmin container=current;
grant create session to ggadmin container=current;
grant alter any table to ggadmin container=current;
grant resource to ggadmin container=current;
grant dv_goldengate_admin, dv_goldengate_redo_access to ggadmin
container=current;
exec dbms_goldengate_auth.grant_admin_privilege('ggadmin');
```

Step 1.3 - Create the Database Services

If the source and target databases are running the recommended configuration on an Oracle RAC cluster with Oracle Data Guard, a role-based service must be created that allows the Extract or Replicat processes to connect to the correct Data Guard primary database instance.

When using a source multitenant database, a separate service is required for the root container database (CDB) and the pluggable database (PDB) that contains the schema being replicated. For a target multitenant database, a single service is required for the PDB.

As the `oracle` OS user on the primary database system, use dbaascli to find the CDB and PDB name, as shown here:

```
[opc@exadb1_node1 ~]$ sudo su - oracle
[oracle@exadb1_node1 ~]$ source dbName.env
```

```
[oracle@exadb1_node1 ~]$ dbaascli database getDetails
 --dbname dbName |egrep 'dbName|pdbName'

  "dbName" : "dbName",
      "pdbName" : "pdbName",
```

As the `oracle` OS user on the primary and standby database systems, create and start the CDB database service using the following command:

```
[opc@exadb1_node1 ~]$ sudo su - oracle
[oracle@exadb1_node1 ~]$ source dbName.env
[oracle@exadb1_node1 ~]$ srvctl add service -db $ORACLE_UNQNAME
 -service dbName.goldengate.com -preferred ORACLE_SID1
 -available ORACLE_SID2 -role PRIMARY
```

As the `oracle` OS user on the primary and standby database systems, create and start the PDB database service using the following command:

```
[oracle@exadb1_node1 ~]$ srvctl add service -db $ORACLE_UNQNAME
 -service dbName.pdbName.goldengate.com -preferred ORACLE_SID1
 -available ORACLE_SID2 -pdb pdbName -role PRIMARY
```

As the `oracle` OS user on the primary and standby database systems, start and verify that the services are running, as shown here:

```
[oracle@exadb1_node1 ~]$ srvctl start service -db $ORACLE_UNQNAME -role
[oracle@exadb1_node1 ~]$ srvctl status service -d $ORACLE_UNQNAME |grep
goldengate

Service dbName.goldengate.com  is running on instance(s) SID1
Service dbName.pdbName.goldengate.com is running on instance(s) SID1
```

> **Note:**
>
> Repeat step 1.3 in the source and target database system.

Task 2: Configure the Oracle GoldenGate Environment

Perform the following steps to complete this task:

- Step 2.1 - Access the GoldenGate deployment
- Step 2.2 - Change the default Oracle GoldenGate administrator password
- Step 2.3 - Create the Database Credentials
- Step 2.4 - Create Autostart profiles
- Step 2.5 - Configure Oracle GoldenGate processes

Step 2.1: Access the deployment

To access the deployment:

- If the client subnet is public, you can access the nodes using the VIP address and SSH tunnel. You must create an SSH tunnel with the VIP DNS Name on both the primary and standby clusters using the following command on all nodes:

  ```
  ssh -N -L <local_port>:<vip-dns-name>:443 -p 22 <gghub-node>
  ```

  The `local_port` can be any available port on the local system. The `gghub-node` is the name of the cluster to which the tunnel is configured.
  Oracle recommends that you manage GoldenGate using the primary and standby VIP DNS names as follows:

  - **Primary VIP DNS Name:** `prim.<cluster-prefix>.goldengate.com`
  - **Standby VIP DNS Name:** `stby.<cluster-prefix>.goldengate.com`

  You can then access the deployment with your web browser using `https://localhost:<local_port>`
- If the client subnet is private,
  1. You must create a Bastion to the client subnet with the following values:
     - **Target VCN**: Select your VCN
     - **Target subnet**: Client subnet name
     - **CIDR block allowlist**: Enter or select the IP addresses or address ranges to allow connections to target resources using SSH sessions.

     Learn more about creating a Bastion.
  2. Create an SSH Port Forwarding session using the primary VIP address and standby VIP address with port 443. Learn more about creating Port Forwarding sessions.
  3. From the session's Actions menu (three dots), select **View SSH command** and click **Copy**.
  4. Run the SSH command, ensuring that you replace `<privateKey>` with the local private key path and `443` for `<localPort>`. Learn more about connecting to a Port Forwarding session
  5. You can then access the deployment with your web browser using `https://localhost`.

Step 2.2: Change the default password

Log in to Oracle GoldenGate 21*c* Service Manager as the Administrator for the deployment, to change the password for the Security Role user.

1. In your web browser, use the localhost address and local port used when you set up the SSH tunnel using the VIP DNS name to the instances. For example, `https://localhost:<local-port>`.
2. Log in as `oggadmin`, and use the password found in the `/mnt/acfs_gg/deployments/ogg_credentials.json` file.
3. After you log in, open the navigation menu, and then select **Administrator**.

4. On the Users page, click **Edit** (pencil icon) for the **oggadmin** user.
5. Update the password, and then click **Submit**.

> **Note:**
>
> The password must be between 8 and 30 alphanumeric upper and lowercase characters, and contain at least one of the following special characters: dash (-), exclamation point (!), percent (%), amperstand (&), asterisk (*), comma (,), pound or hastag (#), and underscore (_).

Upon resetting the `oggadmin` password, your session ends. Log back in to Service Manager using the new password. You must now update the password for `oggadmin` for each of the other deployments.

1. On the Service Manager Home page, select the port number for the Administration Server for a deployment.
2. Log in to the Administration Service as `oggadmin`, and enter the password found in `the /mnt/acfs_gg/deployments/ogg_credentials.json`.
3. On the Administration Service Home page, open the navigation menu, and then select **Administrator**.
4. On the Users page, click **Edit** (pencil icon) for the **oggadmin** user.
5. Update the password, and then cilck **Submit**. The same password rules as above apply.

Next, change the AGCTL password.

1. On any node of primary cluster, as the `opc` user, run `sudo su -` to switch to `root` user.
2. Run the following commands, and enter the new password when prompted:

```
/u01/app/grid/xag/bin/agctl stop goldengate <deployment_name>
/u01/app/grid/xag/bin/agctl modify goldengate <deployment_name> --
adminuser oggadmin
/u01/app/grid/xag/bin/agctl start goldengate <deployment_name>
```

3. On any node of Standby Cluster, as `opc` user, run `sudo su -` to switch to `root` user.
4. Run the following command:

```
/u01/app/grid/xag/bin/agctl modify goldengate <deployment_name> --
adminuser oggadmin
```

Step 2.3: Create database credentials

Use the Oracle GoldenGate Microservices UI to create the database credentials using the above TNS alias names. As the `oggadmin` user, add the database credentials:

1. Log in into the Oracle GoldenGate Administration Service: `https://localhost:<localPort>/<instance_name>/adminsrvr`.
2. In the left navigation menu, click **Configuration**.
3. In the Database tab, click **Add Credentials** (plus icon).
4. Add the required information for each source and target CDB and PDB:

Region	Container	Domain	Alias	User ID
Region 1	CDB	GoldenGate	Reg1_CDB	c##ggadmin@<tns_alias>
Region 1	PDB	GoldenGate	Reg1_PDB	ggadmin@<tns_alias>
Region 2	CDB	GoldenGate	Reg2_CDB	c##ggadmin@<tns_alias>
Region 2	PDB	GoldenGate	Reg2_PDB	ggadmin@<tns_alias>

Step 2.4: Create Autostart profiles

Create a profile that automatically starts the Extract and Replicat processes when the Oracle GoldenGate Administration Server starts, and restarts if any Extract or Replicat processes are abandoned. In GoldenGate Microservices, profiles manage auto start and restart.

Using the Oracle GoldenGate Administration Server UI, create a profile to assign to each Oracle GoldenGate process:

1. Log in to the **Administration Service** on the Source GoldenGate.
2. Click on **Profile** under **Administration Service**.
3. Click the **plus (+)** sign next to Profiles on the Managed Process Settings home page.
4. Enter the details as follows:
   - Profile Name: Start_Default
   - Description: Default auto-start/resteart profile
   - Default Profile: Yes
   - Auto Start: Yes
   - Auto Start Options
     - Startup Delay: 1 min
     - Auto Restart: Yes
   - Auto Restart Options
     - Max Retries: 5
     - Retry Delay: 30 sec
     - Retries Window: 30 min
     - Restart on Failure only: Yes
     - Disable Task After Retries Exhausted: Yes
5. Click **Submit**
6. Repeat these steps on the Target GoldenGate.

Step 2.5: Configure Oracle GoldenGate processes

When creating Extract, Distribution Paths, and Replicat processes with Oracle GoldenGate Microservices Architecture, all files that need to be shared between the GGHub nodes are already shared with the deployment files stored on a shared file system.

Below, are the essential configuration details recommended for running Oracle GoldenGate Microservices on GGhub for Extract, Distribution Paths, and Replicat processes.

Perform the following sub-steps to complete this step:

- Step 2.5.1 - Extract configuration
- Step 2.5.2 - Replicat configuration
- Step 2.5.3 - Distribution Path configuration

Step 2.5.1: Extract configuration

When creating an Extract using the Oracle GoldenGate Administration Service interface, leave the Trail Subdirectory parameter blank so that the trail files are automatically created in the deployment directories stored on the shared file system. The default location for trail files is the `/<deployment directory>/var/lib/data` directory.

> **Note:**
>
> To capture from a multitenant database, you must use an Extract configured at the root level using a c## account. To apply data into a multitenant database, a separate Replicat is needed for each PDB because a Replicat connects at the PDB level and doesn't have access to objects outside of that PDB.

For GoldenGate Extract processes using Data Guard configurations that are using redo transport Maximum Performance or Maximum Availability modes, the following parameter must be added to the Extract process parameter file **on the primary system** to avoid losing transactions and resulting in logical data inconsistencies:

```
TRANLOGOPTIONS HANDLEDLFAILOVER
```

This parameter prevents Extract from extracting transaction data from redo that has not yet been applied to the Data Guard standby database. This is crucial to preventing Oracle GoldenGate from replicating data to a target database that does not exist in the source standby database.

If this parameter is not specified, after a data loss failover of the source database it is possible to have data in the target database that is not present in the source database, leading to logical data inconsistencies.

By default, after 60 seconds, a warning message will be written to the Extract report file when the Extract is stalled due to not being able to query the standby database applied SCN information. For example:

```
WARNING OGG-02721  Extract has been waiting for the standby database for 60
seconds.
```

The amount of time before the warning message is written to Extract report file can be adjusted using the Extract parameter "TRANLOGOPTIONS HANDLEDLFAILOVER STANDBY_WARNING".

If the Extract is still not able to query the standby database applied SCN information after 30 minutes (default), the Extract process will abend, logging the following message in the Extract report file:

```
ERROR   OGG-02722  Extract abended waiting for 1,800 seconds for the
    standby database to be accessible or caught up with the primary database.
```

If the standby database becomes available before the default 30 timeout expires, Extract continues mining data from the source database and reports the following message to the report file:

```
INFO    OGG-02723  Extract resumed from stalled state and started
    processing LCRs.
```

The timeout value of 30 minutes can be adjusted using the Extract parameter "TRANLOGOPTIONS HANDLEDLFAILOVER STANDBY_ABEND <value>", where value is the number of seconds the standby is unavailable before abending.

If the standby database will be unavailable for a prolonged duration, such as during a planned maintenance outage, and you wish Extract to continue extracting data from the primary database, remove the "TRANLOGOPTIONS HANDLEDLFAILOVER" parameter from the Extract parameter file and restart Extract (see example below in Figures 4 to 6). Remember to set the parameter after the standby becomes available.

> **Note:**
>
> If extracting from a primary database continues while the standby is unavailable, a data loss failover could result after the standby becomes available, and not all the primary redo was applied before a failover. The GoldenGate target database will contain data that does not exist in the source database.

If the Extract process has been assigned an auto restart profile, after a Data Guard role transition, the Extract process will automatically restart. Extract will continue to mine redo data from the new primary database, ignoring the current state of the new standby database, until a default 5-minute timeout period expires. After this time, if the standby is not available Extract will abend with the following errors:

```
INFO    OGG-25053  Timeout waiting for 300 seconds for standby database
 reinstatement. Now enforcing HANDLEDLFAILOVER.
ERROR   OGG-06219  Unable to extract data from the Logmining server
OGG$CAP_XXXXX.
ERROR   OGG-02078  Extract encountered a fatal error in a processing thread
and is
 abending.
```

Extract will continue to automatically restart, based on the GoldenGate Microservices auto restart profile, and failing due to reaching the `HANDLEDLFAILOVER` timeout, until the number retries is reached or the new standby database becomes available.

During the timeout period following a database role transition, the `HANDLEDLFAILOVER` parameter is automatically suspended, so data will be replicated to the Oracle GoldenGate replica database without consideration of the source standby database not being kept up to

date. The timeout period for the standby database to start up before Extract abends can be adjusted using the Extract parameter `TRANLOGOPTIONS DLFAILOVER_TIMEOUT`.

It is recommended that you leave `DLFAILOVER_TIMEOUT` at the default of 5 minutes, to allow the old primary to convert to a standby. If the new standby database will be unavailable for an extended period of time or completely gone, then in order for Extract to start and remain running, you must remove the `HANDLEDLFAILOVER` parameter from the Extract parameter file. After removing the parameter, Extract no longer waits until redo has been applied to the standby database before extracting the data.

During the time it takes for the standby database to come back online and apply all the redo from the primary database, there will be data divergence between it and the Oracle GoldenGate replica database. This will be resolved once the standby database is up to date. At which point, add the `HANDLEDLFAILOVER` parameter back into the integrated Extract process parameter file, and then stop and restart the Extract.

When Oracle Data Guard Fast-Start Failover is disabled, such that the broker can automatically fail over to a standby database in the event of loss of the primary database, you must specify an additional integrated Extract parameter shown below.

```
TRANLOGOPTIONS FAILOVERTARGETDESTID n
```

This parameter identifies which standby database the Oracle GoldenGate Extract process must remain behind, with regards to not extracting redo data that has not yet been applied to the standby database.

If Oracle Data Guard Fast-Start Failover is disabled, and you don't specify the additional integrated Extract parameter `FAILOVERTARGETDESTID`, the extract will abend with the following errors:

```
ERROR OGG-06219 Unable to extract data from the Logmining server
OGG$CAP_XXXXX.
ERROR OGG-02078 Extract encountered a fatal error in a processing thread and
is
 abending.
```

To determine the correct value for `FAILOVERTARGETDESTID`, use the `LOG_ARCHIVE_DEST_N` parameter from the GoldenGate source database which is used for sending redo to the source standby database. For example, if `LOG_ARCHIVE_DEST_2` points to the standby database, then use a value of 2.

As the `oracle` user on the primary database system, execute the following command:

```
[opc@exapri-node1 ~]$ sudo su - oracle
[oracle@exapri-node1 ~]$ source <db_name>.env
[oracle@exapri-node1 ~]$ sqlplus / as sysdba

SQL> show parameters log_archive_dest

NAME                 TYPE        VALUE
-------------------- -----------
-----------------------------------------------------
log_archive_dest_1   string      location=USE_DB_RECOVERY_FILE_DEST,
                                 valid_for=(ALL_LOGFILES, ALL_ROLES)

log_archive_dest_2   string      service="<db_name>", SYNC AFFIRM delay=0
```

```
                                        optional compression=disable max_failure=0
reopen=300

                                        db_unique_name="<db_name>" net_timeout=30,
                                        valid_for=(online_logfile,all_roles)
```

In this example, the Extract parameter would be set to the following:

```
TRANLOGOPTIONS FAILOVERTARGETDESTID 2
```

Create the Extract:

1. Log in to the Oracle GoldenGate **Administration Server**
2. Click in **Overview** under **Administration Service**
3. Click the **plus** button to **Add Extract**
4. Select **Integrated Extract**
5. Add the required information as follows:
   - Process Name: EXT_1
   - Description: Extract for Region 1 CDB
   - Intent: Unidirection
   - Begin: Now
   - Trail Name: aa
   - Credential Domain: GoldenGate
   - Credential Alias: Reg1_CDB
   - Register to PDBs: PDB Name
6. Click **Next**
7. If using CDB Root Capture from PDB, add the `SOURCECATALOG` parameter with the PDB Name
8. Click **Create and Run**

Step 2.5.2 - Replicat configuration

Oracle generally recommends using integrated parallel Replicat which offers better apply performance for most workloads when the GGHUB is in the same region as the target Oracle GoldenGate database.

The best apply performance can be achieved when the network latency between the GGHUB and the target database is as low as possible. The following configuration is recommended for the remote Replicat running on the Oracle GGHUB.

- `APPLY_PARALLELISM` – Disables automatic parallelism, instead of using `MAX_APPLY_PARALLELISM` and `MIN_APPLY_PARALLELISM`, and allows the highest amount of concurrency to the target database. It is recommended to set this as high as possible based on available CPU of the hub and the target database server.
- `MAP_PARALLELISM` – Should be set with a value of 2 to 5. With a larger number of appliers, increasing the Mappers increases the ability to hand work to the appliers.
- `BATCHSQL` – applies DML using array processing which reduces the amount network overheads with a higher latency network. Be aware that if there are many data conflicts,

`BATCHSQL` results in reduced performance, as rollback of the batch operations followed by a re-read from trail file to apply in non-batch mode.

Step 2.5.2.1 - Create the Checkpoint Table

The checkpoint table is a required component for Oracle GoldenGate Replicat processes. After connecting to the database from the Credentials page of the Administration Service, you can create the checkpoint table.

Create the checkpoint table in the target deployment:

1. Log in to the Oracle GoldenGate **Administration Server**
2. Click in **Configuration** under **Administration Service**.
3. Click on **Database** and **Connect** to the target database or PDB:
4. Click the plus (+) sign next to Checkpoint. The Add Checkpoint page is displayed.
5. Enter the details as follows:
   - Checkpoint Table: ggadmin.chkp_table
6. Click **Submit**

Refer to Oracle GoldenGate with Oracle Database Guide for more information on the checkpoint table.

Step 2.5.2.2 - Add a Replicat

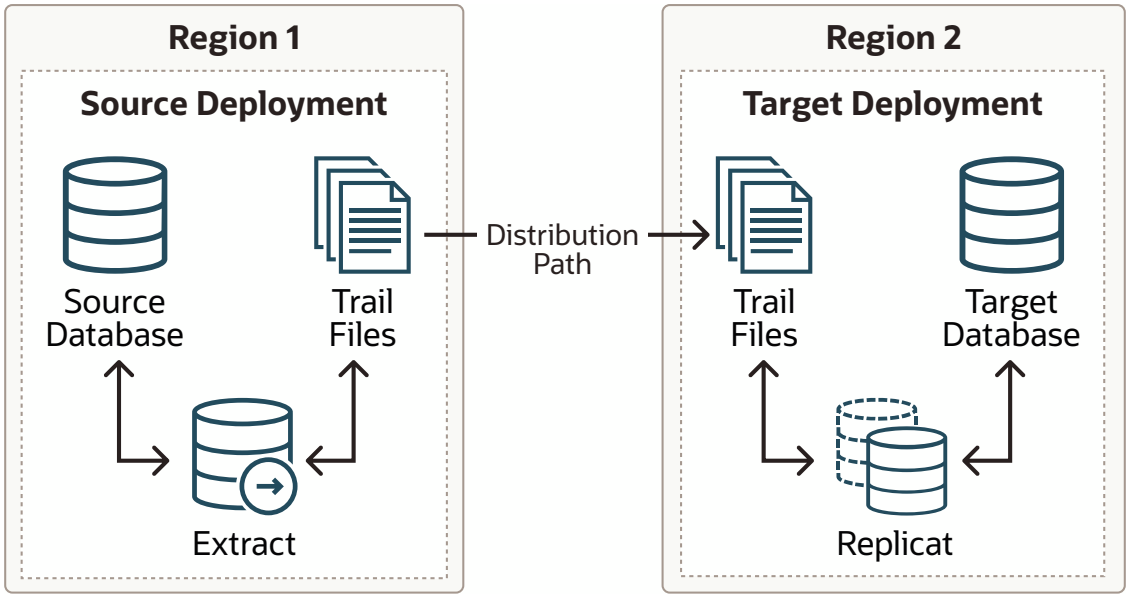
After you've set up your database connections and verified them, you can add a Replicat for the deployment by following these steps:

1. Log in to the Oracle GoldenGate **Administration Server**
2. Click theplus (+) sign next to **Replicats** on the Administration Service home page. The Add Replicat page is displayed.
3. Select a Replicat type and click Next.
4. Enter the details as follows:
   - Process Name: REP_1
   - Description: Replicat for Region 2 PDB
   - Intent: Unidirectional
   - Credential Domain: GoldenGate
   - Credential Alias: Reg2_PDB
   - Source: Trail
   - Trail Name: aa
   - Begin: Position in Log
   - Checkpoint Table: "GGADMIN"."CHKP_TABLE"
5. Click **Next**
6. From the **Action Menu**, click **Start**.

Step 2.5.3 - Distribution Path configuration

Distribution paths are only necessary when trail files need to be sent to an additional Oracle GoldenGate Hub in a different, or even the same, region as described in the following figure.

Figure 3-1 Oracle GoldenGate Distribution Path



When using Oracle GoldenGate Distribution paths with the NGINX Reverse Proxy, additional steps must be carried out to ensure the path client and server certificates are configured.

More instructions about creating distribution paths are available in *Using Oracle GoldenGate Microservices Architecture*. A step-by-step example is in the following video, "Connect an on-premises Oracle GoldenGate to OCI GoldenGate using NGINX," to correctly configure the certificates.

Here are the steps performed in this sub-step:

- Step 2.5.3.1 - Download the Target Server's Root Certificate, and then upload it to the source Oracle GoldenGate
- Step 2.5.3.2 - Create a user in the Target Deployment for the Source Oracle GoldenGate to use
- Step 2.5.3.3 - Create a Credential in the Source Oracle GoldenGate
- Step 2.5.3.4 - Create a Distribution Path on the Source Oracle GoldenGate to the Target Deployment
- Step 2.5.3.5 - Verify the Connection in the Target Deployment Console Receiver Service

Step 2.5.3.1 - Download the Target Server's Root Certificate, and then upload it to the source Oracle GoldenGate

Download the target deployment server's root certificate and add the CA certificate to the source deployment Service Manager.

1. Log in to the **Administration Service** on the Target GoldenGate.
2. Follow "Step 2 - Download the target server's root certificate" in the video "Connect an on-premises Oracle GoldenGate to OCI GoldenGate using NGINX."

Step 2.5.3.2 - Create a user in the Target Deployment for the Source Oracle GoldenGate to use

Create a user in the target deployment for the distribution path to connect to:

1. Log in to the **Administration Service** on the Target GoldenGate.
2. Click on Administrator under Administration Service.
3. Click the plus (+) sign next to Users.
4. Enter the details as follows:
   - Username: ggnet
   - Role: Operator
   - Type: Password
5. Click **Submit**

Step 2.5.3.3 - Create a Credential in the Source Oracle GoldenGate

Create a credential in the source deployment connecting the target deployment with the user created in the previous step. For example, a domain of OP2C and an alias of WSSNET.

1. Log in to the **Administration Service** on the Source Oracle GoldenGate.
2. Click in **Configuration** under **Administration Service.**
3. Click the **plus (+)** sign next to Credentials on the Database home page.
4. Enter the details as follows:
   - Credential Domain: OP2C
   - Credential Alias: wssnet
   - User ID: ggnet
5. Click **Submit**

Step 2.5.3.4 - Create a Distribution Path on the Source Oracle GoldenGate to the Target Deployment

A path is created to send trail files from the Distribution Server to the Receiver Server. You can create a path from the Distribution Service. To add a path for the source deployment:

1. Log in to the **Distribution Service** on the Source Oracle Goldengate.
2. Click the plus (+) sign next to Path on the Distribution Service home page. The Add Path page is displayed.
3. Enter the details as follows:

Option	Description
Path Name	Select a name for the path.
Source: *Trail Name*	Select the Extract name from the drop-down list, which populates the trail name automatically. If it doesn't, enter the trail name you provided while adding the Extract.
Generated Source URI	Specify localhost for the server's name; this allows the distribution path to be started on any of the Oracle RAC nodes.
Target Authentication Method	Use 'UserID Alias'
Target	Set the **Target** transfer protocol to wss (secure web socket). Set the **Target Host** to the target hostname/VIP that will be used for connecting to the target system along with the **Port Number** that NGINX was configured with (default is 443).

Option	Description
Domain	Set the **Domain** to the credential domain created above in Step 2.3.3.3, for example, OP2C.
Alias	The **Alias** is set to the credential alias wssnet, also created in Step 2.3.3.3.
Auto Restart Options	Set the distribution path to restart when the Distribution Server starts automatically. This is required, so that manual intervention is not required after a RAC node relocation of the Distribution Server. It is recommended to set the number of **Retries** to 10. Set the **Delay**, which is the time in minutes to pause between restart attempts, to 1.

4. Click **Create Path**.
5. From the Action Menu, click **Start**.

4
Use

Discover different use cases for Oracle GoldenGate Maximum Availability Hub on Oracle Cloud Marketplace.

Topics in this section:

Monitor ACFS replication

Learn to monitor ACFS replication.

Use the following command to monitor ACFS replication:

```
/sbin/acfsutil repl info -c -v /mnt/acfs_gg
```

You can use the command on both the primary and standby databases.

For example, running the command on the primary cluster where ACFS is mounted produces the following output. Ensure that Primary Status is Running and Background Resources are Active for the primary cluster.

```
[grid@oggclt1-node1 ~]$ /sbin/acfsutil repl info -c -v /mnt/acfs_gg
Site: Primary
Primary hostname: prim.oggclt.goldengate.com
Primary path: /mnt/acfs_gg
Primary status: Running
Background Resources: Active

Standby connect string: grid@stby.oggclt.goldengate.com
Standby path: /mnt/acfs_gg
Replication interval: 0 days, 0 hours, 0 minutes, 0 seconds
Sending primary as of: Wed Mar 06 17:12:34 2024
Status: Sending incremental differences
Lag Time: 00:00:34 (Constant mode)
Retries made: 0
Last send started at: Wed Mar 06 17:12:35 2024
Last send completed at: In progress
Next send starts at: On send completion
Replicated tags:
Data transfer compression: On
ssh strict host key checking: On
Debug log level: 3
Replication ID: 0x966f1b11
```

The next example demonstrates the command run on the standby cluster where ACFS is mounted:

```
[grid@oggclt2-node1 ~]$ /sbin/acfsutil repl info -c -v /mnt/acfs_gg
Site: Standby
Primary hostname: prim.oggclt.goldengate.com
Primary path: /mnt/acfs_gg

Standby connect string: grid@stby.oggclt.goldengate.com
Standby path: /mnt/acfs_gg
Replication interval: 0 days, 0 hours, 0 minutes, 0 seconds
Last sync time with primary: Wed Mar 06 17:17:18 2024
Receiving primary as of: Wed Mar 06 17:17:18 2024
Status: Receive Completed
Last receive started at: Wed Mar 06 17:17:19 2024
Last receive completed at: Wed Mar 06 17:17:25 2024
Elapsed time for last receive: 0 days, 0 hours, 0 minutes, 6 seconds
Data transfer compression: On
ssh strict host key checking: On
Debug log level: 3
Replication ID: 0x966f1b11
```

Managing Planned Outages

When there is a requirement to perform planned maintenance on the GoldenGate hub, some of the CRS resources should be stopped and disabled to prevent them from restarting, or from causing undesirable results when incorrectly instigating a file system failover, or stopping GoldenGate from running. Use the following recommendations in the event of a planned outage of the primary or standby hub clusters.

For all planned maintenance events:

- Operating system software or hardware updates and patches
- Oracle Grid Infrastructure interim or diagnostic patches
- Oracle Grid Infrastructure quarterly updates under the Critical Patch Update (CPU) program, or Oracle Grid Infrastructure release upgrades
- GGHub software life cycle, including:
  - Oracle GoldenGate
  - Oracle Grid Infrastructure Agent
  - NGINX

> **Caution:**
>
> Ensure all resources are in an ONLINE state and `ora.oggdata.gg_acfs_vol.acfs` resource state details is `mounted on /mnt/acfs_gg, STABLE` on any primary node cluster before you initiate cluster switchover on the standby cluster. You can run the following command to check the resource status:
>
> ```
> crsctl stat res -t
> ```

High Availability Solutions with Target Outage Time:

Seconds to minutes where GoldenGate replication is temporarily suspended

Step 1: Software update of idle GGHub node

Step 2: GGhub Node Relocate

Step 3: Software update of the remaining inactive GGHub node

GGHub Node Relocate

As the grid user on the primary GGHub system, relocate the Oracle GoldenGate Instance:

```
[grid@gghubad11 ~]$ agctl status goldengate


Goldengate instance 'Marketplace' is running on gghubad12

[grid@gghubad11 ~]$ time agctl relocate goldengate Marketplace


real    0m43.984s
user    0m0.156s
sys     0m0.049s
```

As the grid user on the primary GGHub system, check the status of the Oracle GoldenGate Instance:

```
[grid@gghubad11 ~]$ agctl status goldengate


Goldengate instance 'Marketplace' is running on gghubad11
```

GGHub Role Reversal for DR events or to move GGHub in the same region as the target database

GGHUB role reversal performs an ACFS role reversal so that the standby becomes the new primary. With both primary and standby file systems online, the `acfsutil repl failover` command ensures that all outstanding primary file system changes are transferred and applied to the standby before the role reversal completes.

When should we use GGHUB role reversal:

- To move the GGHUB deployment close to the target database for replication performance.

- To support site outage
- To support site maintenance

As the grid user on the current standby file system GGhub node, execute the script to perform the ACFS role reversal:

```
[grid@gghub_stby1]$ sh /u01/oracle/scripts/acfs_role_reversal.sh -m /mnt/
acfs_gg -d Marketplace
################################################################################
##
ACFS Primary Site: prim.oggcl.goldengate.com
ACFS Standby Site: stby.oggcl.goldengate.com
################################################################################
##
Site:                     Primary
Primary status:           Running
Status:                   Sending incremental differences
Lag Time:                 00:00:24 (Constant mode)
Retries made:             0
Last send started at:     Wed Jun 05 15:46:04 2024
Last send completed at:   In progress
################################################################################
##
Site:                          Standby
Last sync time with primary: Wed Jun 05 15:45:41 2024
Status:                        Receiving incremental differences
Last receive started at:       Wed Jun 05 15:46:05 2024
Last receive completed at:     In progress
################################################################################
##
Wed Jun  5 15:46:05 GMT 2024 - Begin Stop GoldenGate Marketplace
Wed Jun  5 15:46:08 GMT 2024 - End Stop GoldenGate Marketplace
################################################################################
##
Wed Jun  5 15:46:08 GMT 2024 - Begin Role Reversal
Wed Jun  5 15:47:25 GMT 2024 - End Role Reversal
################################################################################
##
ACFS Primary Site: stby.oggcl.goldengate.com
ACFS Standby Site: prim.oggcl.goldengate.com
################################################################################
##
Site:                      Primary
Primary status:            Running
Status:                    Send Completed
Lag Time:                  00:00:00
Retries made:              0
Last send started at:      Wed Jun 05 15:47:12 2024
Last send completed at:    Wed Jun 05 15:47:23 2024
################################################################################
##
Site:                           Standby
Last sync time with primary:    Wed Jun 05 15:47:12 2024
Status:                         Receive Completed
Last receive started at:        Wed Jun 05 15:47:18 2024
Last receive completed at:      Wed Jun 05 15:47:18 2024
```

```
################################################################################
##
Wed Jun  5 15:47:25 GMT 2024 - Begin Start GoldenGate Marketplace
Wed Jun  5 15:49:49 GMT 2024 - End Start GoldenGate Marketplace
################################################################################
##
```

As the grid user on the current new primary file system GGhub node, check the status of the Oracle GoldenGate deployment:

```
[grid@gghub_stby1]$ agctl status goldengate

Goldengate  instance 'Marketplace' is running on gghub_stby1
```

Managing unplanned outages

Expected impact with unplanned outages

When an unplanned outage occurs on either the primary or standby GGHub clusters, there are some instructions to ensure the continuous operation of GoldenGate. Use the following GGHUB failure use cases to guide you in the event of an unplanned outage of the primary and standby GGHUB systems.

Use case #1 – Standby Hub Failure or Primary GGHub cannot communicate with the Standby GGHub

If the primary GGhub cannot communicate with the standby GGhub, the following messages will be output into the primary CRS trace file (crsd_scriptagent_grid.trc) on the active cluster node:

```
2023-06-21 12:06:59.506 :CLSDYNAM:1427187456: [acfs_primary]{1:8532:12141}
[check] Executing action script: /u01/oracle/scripts/acfs_primary.scr[check]
2023-06-21 12:07:05.666 :CLSDYNAM:1427187456: [acfs_primary]{1:8532:12141}
[check] WARNING: STANDBY not accessible (attempt 1 of 3))
2023-06-21 12:07:18.683 :CLSDYNAM:1427187456: [acfs_primary]{1:8532:12141}
[check] WARNING: STANDBY not accessible (attempt 2 of 3))
2023-06-21 12:07:31.751 :CLSDYNAM:1427187456: [acfs_primary]{1:8532:12141}
[check] WARNING: STANDBY not accessible (attempt 3 of 3))
2023-06-21 12:07:31.751 :CLSDYNAM:1427187456: [acfs_primary]{1:8532:12141}
[check] WARNING: Problem with STANDBY file system (error: 222)
```

At this time, the standby file system is no longer receiving the primary file system changes. The primary file system and Oracle GoldenGate will continue to function unimpeded.

Use the following action plan with this scenario.

- Check the standby file system, using the command `acfsutil repl util verifystandby /mnt/acfs_gg -v` to determine why the standby hub is inaccessible.

- After fixing the cause of the communication errors, the standby will automatically catch up applying the outstanding primary file system changes. The warning messages will no longer be reported into the CRS trace file, being replaced with the following message:

```
2023-06-21 12:15:01.720 :CLSDYNAM:1427187456: [acfs_primary]{1:8532:12141}
[check] SUCCESS: STANDBY file system /mnt/acfs_gg is ONLINE
```

Use case #2 – Primary GGHub Failure or Standby GGHub cannot communicate with the Primary GGHub

If the standby GGhub cannot communicate with the primary GGhub, the the following messages will be output into the standby CRS trace file (`crsd_scriptagent_grid.trc`) on the active cluster node:

```
2023-06-21 12:24:03.823 :CLSDYNAM:4156544768: [acfs_standby]{1:10141:2}
[check] Executing action script: /u01/oracle/scripts/acfs_standby.scr[check]
2023-06-21 12:24:06.928 :CLSDYNAM:4156544768: [acfs_standby]{1:10141:2}
[check] WARNING: PRIMARY not accessible (attempt 1 of 3)
2023-06-21 12:24:19.945 :CLSDYNAM:4156544768: [acfs_standby]{1:10141:2}
[check] WARNING: PRIMARY not accessible (attempt 2 of 3)
2023-06-21 12:24:32.962 :CLSDYNAM:4156544768: [acfs_standby]{1:10141:2}
[check] WARNING: PRIMARY not accessible (attempt 3 of 3)
2023-06-21 12:24:32.962 :CLSDYNAM:4156544768: [acfs_standby]{1:10141:2}
[check] WARNING: Problem with PRIMARY file system (error: 222)
```

At this time, it is unlikely that the standby file system is receiving file system changes from the primary file system.

Use the following action plan with this scenario.

- Check the primary file system, using the command `acfsutil repl util verifyprimary /mnt/acfs_gg -v` to determine why the primary hub is inaccessible.
- If the primary file system cluster is down and cannot be restarted, run `/u01/oracle/scripts/acfs_role_reversal_outage.sh` script:

  ```
  [grid@gghub_stby1]$ /u01/oracle/scripts/acfs_role_reversal_outage.sh -
  m /mnt/acfs_gg -d Marketplace   # Specify the correct mount point

  Wed May 22 19:41:18 GMT 2024 - Begin Role Reversal
  Wed May 22 19:41:32 GMT 2024 - End Role Reversal
  Wed May 22 19:41:32 GMT 2024 - Begin Start GG Marketplace
  Wed May 22 19:43:06 GMT 2024 - End Start GG Marketplace
  ```

- When the old primary file system comes back online, if connectivity is resumed between the new primary and old primary, the old primary file system will automatically convert to the standby.
- If the old primary file system comes back online, but connectivity cannot be established between the primary and standby file systems the acfs_primary resource will detect that node had crashed, and because connectivity to the standby cannot be confirmed, GoldenGate will not be started. This avoids a ‘split-brain’ where two file systems think they are both the primary because they cannot commnunicate with each other.

Use case #3 – Double Failure Case: Primary GGHub Failure and Standby GGHub Connectivity Failure

If the primary GGhub crashes and communication cannot be established with the standby file system when it comes back online, the following messages will be output into the primary CRS trace file (`crsd_scriptagent_grid.trc`) on the active cluster node:

```
2023-06-21 17:08:52.621:[acfs_primary]{1:40360:36312} [start] WARNING:
PRIMARY file system /mnt/acfs_gg previously crashed
2023-06-21 17:08:55.678:[acfs_primary]{1:40360:36312} [start] WARNING:
STANDBY not accessible - disabling acfs_primary
```

If an attempt is made to manually restart the primary file system, an additional message will be output into the CRS trace file:

```
2023-06-21 17:25:54.224:[acfs_primary]{1:40360:37687} [start] WARNING:
    PRIMARY /mnt/acfs_gg disabled to prevent split brain
```

Use the following action plan with this scenario.

- Check the standby file system, using the command `acfsutil repl util verifystandby /mnt/acfs_gg -v` to determine why the standby hub is inaccessible.
- If communication with the the standby file system can re-established, restart GoldenGate on the primary hub:

  ```
  [grid@gghub_prim1]$ agctl start goldengate <instance_name> # Specify the
  GoldenGate instance name

  [grid@gghub_prim1]$ agctl status goldengate

  Goldengate  instance '<instance_name>' is running on gghub_prim1
  ```

- If communication with the standby file system cannot be re-established, use the following commands to restart GoldenGate on the primary hub:

  ```
  [grid@gghub_prim1]$ echo "RESTART" > /mnt/acfs_gg/status/acfs_primary

  [grid@gghub_prim1]$ agctl start goldengate <instance_name>     # Specify
  the GoldenGate instance name

  [grid@gghub_prim1]$ agctl status goldengate

  Goldengate  instance '<instance_name>' is running on gghub_prim1
  ```

- When communication with the standby file system is restored, ACFS Replication will continue to replicate primary file system changes.

5
Upgrade

Topics in this section:

Upgrade Oracle GoldenGate Maximum Availability Hub Stack

Learn to upgrade an existing Oracle GoldenGate Marketplace stack using OCI Stacks.

Before you upgrade, ensure that you:

- Shut down all Oracle GoldenGate processes in the VM instance.
- Back up all block storage, in case you need to revert back in the future because of a failed upgrade or update.
- Take note of or backup any files and configuration settings saved on the VMs that are located outside of the GoldenGate block volume so you can reapply them after the stack upgrade.

About Stacks

Stacks are zip files that contain the latest Terraform code base. Stacks enable you to provision a new compute node and attach your existing block storage. This approach ensures that Oracle GoldenGate does not lose any associated data present in the environment.

Download the latest stack

Before you upgrade Oracle GoldenGate Marketplace, you have to download the latest stack. To download the stack:

1. Log in to Oracle Cloud Infrastructure.
2. In the Oracle Cloud navigation menu, select Marketplace, and then All applications.
3. Enter `Oracle GoldenGate Maximum Availability Hub` into the Marketplace search bar.
4. On the Oracle GoldenGate Maximum Availability Hub page, click the **Download** link located beneath the **Launch Stack** button.

Identify the stack's Terraform version

Before you upgrade the stack, identify the Terraform version of the stack to upgrade.

To identify the stack's Terraform version:

1. Log in to Oracle Cloud Infrastructure.
2. In the Oracle Cloud navigation menu, select **Developer Services**, and then under **Resource Manager**, click **Stacks**.

3. On the Stacks page, select the stack that you want to upgrade.
4. On the Stack details page, under Stack Information, locate the Terraform version field and take note of its value.

Upgrade an Oracle Oracle GoldenGate Maximum Availability Hub image

Before you upgrade the stack, verify that `/mnt/acfs_gg/deployments/ogg-credentials.json` on primary cluster node-1 has the latest oggadmin password saved.

To upgrade an existing Oracle GoldenGate Maximum Availability Hub image:

1. Download the latest stack.
2. In the Oracle Cloud navigation menu, select **Developer Services**, and then under Resource Manager, click **Stacks**.
3. On the Stacks page, select the stack to upgrade.
4. On the Stack's details page, click **Edit**, then select **Edit stack**.
5. On the Edit stack page, for **Terraform configuration source**, click **Browse**, and upload the latest stack zip file.
6. Verify the autofilled fields are correct on each page of the Edit stack form, and then click **Save changes**.

> **WARNING:**
>
> Don't modify any autofilled fields in the update stack workflow.

7. Log in to your Oracle GoldenGate instance and stop all running processes. Ensure that you stop all Microservices (Administration, Distribution, Receiver, and Performance Metrics), including Service Manager.
8. Back in Oracle Cloud, from the navigation menu, select **Compute**, and then click **Instances**.
9. On the Instances page, select the instance to upgrade, and then click **Terminate**.

> **Note:**
>
> You can:
>
> - Click the instance and then click **Terminate** on the Instance details page
> - Check the box next to the instance in the list, and then select **Terminate** from the **Actions** menu
> - Open the instance's **Actions** (three dots) menu and then select **Terminate**.

10. In the Terminate instance dialog, select **Permanently delete attached boot volume**, and then click **Terminate instance**.
11. After the instance is terminated, use the Oracle Cloud navigation menu to return to the Stacks page.
12. On the Stacks page, select the stack to upgrade. It must be the same one you edited earlier.

13. On the Stack's details page, click **Apply**.

After the job completes, verify that the compute node is running. Monitor the `/tmp/startupScript.log` on the primary cluster's node-1 to check if the deployment startup succeeded. You can then access the deployment.

Patch and switch Oracle Grid Infrastructure Homes

Complete the following steps for both the primary and standby clusters of your Oracle GoldenGate Maximum Availability Hub stack.

You can use the `switchGridHome` command only to switch between different RUs of the same release.

1. Download the Oracle Database Release Update (RU) that you want to apply from My Oracle Support into the `/tmp` directory of **node1**.
2. As the `root` user on both **node1** and **node2**, create a new Oracle Grid Infrastructure home directory and disable `iptables`. Ensure that you replace `<version-number>` with the appropriate RU version number.

```
$ mkdir -p /u01/app/<version-number>/grid
$ chown grid:oinstall /u01/app/<version-number>/grid
$ systemctl stop iptables
$ systemctl disable iptables
```

> **Note:**
>
> The new Oracle Grid Infrastructure home path must be different from the current Oracle Grid Infrastructure home path.

3. As the `grid` user on **node1** of the cluster, download the Oracle Grid Infrastructure images files and extract the files to the new Oracle Grid Infrastructure home directory. Copy the grid setup response file from the existing Oracle Grid Infrastructure home path to the new home path. Ensure that you replace `<version-number>` with the appropriate RU version number.

```
$ cd /u01/app/<version-number>/grid
$ unzip -q download_location/grid.zip
$ cp Old_GI_Home/gridsetup.rsp  .
```

4. Start the Oracle Grid Infrastructure installer on **node1** with the `-switchGridHome` flag to switch to the patched Oracle Grid Infrastructure home after the installation and the optional `-applyRU` flag to apply Release Updates (RUs) during the installation. Ensure that you replace `<version-number>` with the appropriate RU version number.

```
$ /u01/app/<version-number>/grid/gridSetup.sh -responseFile /u01/app/
<version-number>/grid/gridsetup.rsp -switchGridHome [-applyRU
patch_directory_location]
[-applyOneOffs comma_seperated_list_of_patch_directory_locations]
```

5. As the `root` user, run the following script on **node1** of the cluster:

```
$ /u01/app/<version-number>/grid/root.sh
```

Then, run the script on **node2**.

6. Verify that the patching has completed.

```
$ crsctl query crs activeversion -f
Oracle Clusterware active version on the cluster is [19.0.0.0.0]. The
cluster upgrade state is [NORMAL].
The cluster active patch level is [patch_level].
```

7. Verify that all CRS processes are running from the new Grid Home.

```
$ ps -ef | grep d\\.bin
```

The response should return the following:

```
grid      6322     1  0 19:16 ?        00:00:00 /u01/app/<version-number>/
grid/bin/acfsrepl_dupd.bin /mnt/acfs_gg
root     28946     1  0 19:12 ?        00:00:05 /u01/app/<version-number>/
grid/bin/ohasd.bin reboot
_ORA_BLOCKING_STACK_LOCALE=AMERICAN_AMERICA.AL32UTF8
grid     29169     1  0 19:12 ?        00:00:00 /u01/app/<version-number>/
grid/bin/mdnsd.bin
grid     29170     1  0 19:12 ?        00:00:02 /u01/app/<version-number>/
grid/bin/evmd.bin
grid     29209     1  0 19:12 ?        00:00:00 /u01/app/<version-number>/
grid/bin/gpnpd.bin
grid     29274     1  0 19:12 ?        00:00:02 /u01/app/<version-number>/
grid/bin/gipcd.bin
root     29310     1  0 19:12 ?        00:00:05 /u01/app/<version-number>/
grid/bin/osysmond.bin
grid     29397     1  0 19:12 ?        00:00:05 /u01/app/<version-number>/
grid/bin/ocssd.bin  -S 1
root     29595     1  0 19:12 ?        00:00:02 /u01/app/<version-number>/
grid/bin/octssd.bin reboot
root     29629     1  1 19:12 ?        00:00:11 /u01/app/<version-number>/
grid/bin/crsd.bin reboot
```

8. Enable and restart `iptables` on both nodes of the cluster.

```
$ systemctl enable iptables
$ systemctl restart iptables
```

9. If the patching fails, then perform the following steps to rollback the patch:
   a. As the `root` user, run the prepatch script.

```
# /u01/app/<version-number>/grid/crs/install/rootcrs.sh -prepatch -
dstcrshome Old_GI_Home -rollback
```

b. As the `root` user, run the postpatch script.

```
# /u01/app/<version-number>/grid/crs/install/rootcrs.sh -postpatch -
dstcrshome Old_GI_Home -rollback
```

10. If you have successfully switched to the new Grid home on all nodes and want to switch back to the old Grid home, then perform the following steps:

a. As the `root` user, unlock the old Grid home.

```
# Old_GI_Home/crs/install/rootcrs.sh -unlock -crshome Old_GI_Home
```

b. As the `grid` user, run `switchGridHome` from the old Grid home.

```
$ Old_GI_Home/gridSetup.sh -silent -switchGridHome [-
zeroDowntimeGIPatching] [-skipDriverUpdate]
```

6
Get help

Should you run into any issues with your Oracle GoldenGate Maximum Availability Hub solution, explore ways to troubleshoot or learn where to get help.

Submit a Service Request

You can raise a service request with My Oracle Support if you need help resolving issues when working with Oracle GoldenGate Maximum Availability Hub.

My Oracle Support is a customer portal that offers product services through various support tools and contains a repository of useful information, where you can find solution to your issue. To create a service request on My Oracle Support, you must have:

- A Support Identifier which verifies your eligibility for Support services
- A My Oracle Support account

1. Access My Oracle Support at https://support.oracle.com/.
2. Click the **Service Requests** tab.
3. Click **Create Technical SR**, and then complete the fields as follows:
    a. Enter the **Problem Summary**, with as much detail as possible.
    b. Enter the **Problem Description**, with as much detail as possible.
    c. Enter the **Error Codes**, if applicable.
    d. In the **Where is the Problem** section, click the **Software** tab.
        i. For **Product**, select **Oracle GoldenGate**.
        ii. Complete the rest of the fields as needed.
        iii. For **Problem Type**, select GoldenGate on OCI (Oracle Cloud Infrastructure) Marketplace connecting to Oracle Database or non-Oracle, depending on your set up.
        iv. Enter the **Support Identifier**.
        v. Select **Yes** for **Is your software running on Oracle Cloud Infrastructure**.
        vi. For **Cloud Infrastructure Product**, select the appropriate option.
    e. Select the **Severity**, and then click **Next**.
    f. Complete the contact information fields, and then click **Create SR**.

Known issues

Discover some common issues you may encounter while using Oracle GoldenGate Maximum Availability Hub and how to work around them.

Terraform destroy fails if instances are in stopped state

Description: If nodes created by the Oracle GoldenGate Maximum Availability Hub stack are in a Stopped state, terraform destroy fails.

Workaround: You can perform a terraform destroy on the stack even when instances are stopped.

Normal Oracle GoldenGate 19 and 21 listings don't attach secondary VNIC, while Oracle GoldenGate Maximum Availability Hub requires a secondary VNIC. When trying to destroy the stack, it would attempt to detach the secondary VNIC and in order to do that, the instance should be running. If the instance is down, then it fails to detach the secondary VNIC.

If there are failures for the resource in the terraform destroy logs, you can destroy the resources in the Oracle Cloud console. Once you destroy the resources that were failing, then run the destroy stack again then it should proceed without error. Note that when you destroy the instance, it also destroys all associated resources.